

В.А.Острейковский

ИНФОРМАТИКА

•••••

Допущено
Министерством образования
Российской Федерации
в качестве учебного пособия
для студентов учреждений среднего
профессионального образования

5944

БИБЛИОТЕКА
Колледжа предпринимательства
и социального управления



Москва

«Высшая школа» 2001

УДК 007
ББК 32.81
О 76

Рецензенты:

Сургутский нефтяной техникум (директор Л.М. Джебраилов);
профессор В.А. Тупчиев (зав. каф. «Прикладная математика»
Обнинского института атомной энергетики)

Острейковский, В.А.
О 76 Информатика: Учеб. пособие для студ. сред. проф. учеб. заве-
дений. — М.: Высш. шк., 2001.— 319 с.: ил.

ISBN 5-06-003803-3

В учебном пособии согласно требованиям «Государственного образовательного стандарта» рассмотрены основные понятия информатики как естественно-научной дисциплины, дана общая характеристика процесса сбора, обработки, хранения и передачи информации. Проанализировано аппаратное и программное обеспечение ЭВМ. Приведено описание структуры и функций персонального компьютера. Большое внимание уделено раскрытию основных видов информационных технологий и информатизации общества.

Для студентов средних профессиональных учебных заведений и колледжей.

УДК 007
ББК 32.81

ISBN 5-06-003803-3

© ГУП «Издательство «Высшая школа», 2001

Оригинал-макет данного издания является собственностью издательства «Высшая школа», и его репродуцирование (воспроизведение) любым способом без согласия издательства запрещено.

Список основных сокращений

АИС	— автоматизированная информационная система
АЛУ	— арифметико-логическое устройство
АПД	— аппаратура передачи данных
АРМ	— автоматизированное рабочее место
АСНИ	— автоматизированная система научных исследований
АСУ	— автоматизированная система управления
АСУП	— автоматизированная система управления предприятием
АСУТП	— автоматизированная система управления технологическими процессами
7468 АЦПУ	— алфавитно-цифровое печатающее устройство
БЗ	— база знаний
БД	— база данных
БИС	— большая интегральная схема
БнД	— банк данных
БИТ	— [от англ. bi(binary digit) – двоичная цифра] – двоичная единица измерения энтропии и количества информации
ВЗУ	— внешнее запоминающее устройство
ВС	— вычислительная система
ВСт	— вычислительная сеть
ВТ	— вычислительная техника
ВУ	— внешнее устройство
ГМД	— гибкий магнитный диск
ЕС ЭВМ	— единая система ЭВМ
ЗУ	— запоминающее устройство
ИВЦ	— информационно-вычислительный центр
ИИ	— искусственный интеллект
ИР	— информационный ресурс
ИС	— информационная система
ИТ	— информационная технология
К	— контроллер
КБ	— конструкторское бюро
КМ	— канал мультиплексный
КС	— канал селекторный
ЛВС	— локальная вычислительная сеть
МД	— магнитный диск

БИБЛИОТЕКА
Колледжа предпринимательства
и социального управления

МЛ	— магнитная лента
МП	— микропроцессор
НГМД	— накопитель на гибком магнитном диске
НЖМД	— накопитель на жестком магнитном диске
НМД	— накопитель на магнитном диске
НМЛ	— накопитель на магнитной ленте
ОГАС	— общегосударственная автоматизированная система
ОЗУ	— оперативное запоминающее устройство
ОП	— оперативная память
ОС	— операционная система
ПК	— персональный компьютер
ПО	— программное обеспечение
ПП	— постоянная память
ПЗУ	— постоянное запоминающее устройство
ППП	— пакет прикладных программ
ПУ	— печатающее устройство
ПЭВМ	— персональная ЭВМ
САПР	— система автоматизированного проектирования
СБИС	— сверхбольшая интегральная схема
СВМ	— система виртуальных машин
СЗ	— система знаний
СОП	— сверхоперативная память
СМ ЭВМ	— серия мини-ЭВМ
СП	— сопроцессор
СПО	— система поддержки оперативного персонала
СУБД	— система управления базами данных
ТОУ	— технологический объект управления
УВВ	— устройство ввода-вывода
УВв	— устройство ввода
УВыв	— устройство вывода
УУ	— устройство управления
УПД	— устройство подготовки данных
УСО	— устройство связи с объектом
ЦВМ	— цифровая вычислительная машина
ЦП	— центральный процессор
ЭВМ	— электронная вычислительная машина
ЭЛТ	— электронно-лучевая трубка
ЭС	— экспертная система
CD-ROM	— Compact Disk - Read Only Memory
IBM	— International Business Machines Corporation
MS	— Microsoft
WWW(W3)	— World Wide Web

ПРЕДИСЛОВИЕ

Во второй половине XX в. человечество вступило в новый этап своего развития. В этот период начался переход от индустриального общества к информационному. Процесс, обеспечивающий этот переход, получил название информатизации. **ИНФОРМАТИЗАЦИЯ** — это процесс создания, развития и всеобщего применения информационных средств и технологий, обеспечивающих достижение и поддержание уровня информированности всех членов общества, необходимого и достаточного для кардинального улучшения качества труда и условий жизни в обществе. При этом информация становится важнейшим стратегическим ресурсом общества и занимает ключевое место в экономике, образовании и культуре.

Неизбежность информатизации общества обусловлена резким возрастанием роли и значения информации. Информационное общество характеризуется высокоразвитой информационной сферой, которая включает деятельность человека по созданию, переработке, хранению, передаче и накоплению информации.

Научным фундаментом процесса информатизации общества является новая научная дисциплина — *информатика*.

Целями данного учебного пособия являются:

- 1) четкое изложение основных понятий и современных подходов к информатике как естественнонаучной дисциплине;
- 2) изучение математических основ информатики как инструмента для решения прикладных задач;
- 3) освоение первоначальных знаний по структуре и функциям блоков ЭВМ, алгоритмизации и программированию;
- 4) демонстрация возможностей информатики в современных информационных технологиях.

В соответствии с поставленными целями материал изложен в предисловии, введении, трех разделах, девяти главах, заключении и приложении.

Предисловие призвано помочь студенту лучше понять содержание учебного пособия. Здесь приводится краткий анализ научных основ и практических достижений информатики, даются основные принципы, положенные в основу учебного пособия. Рассматриваются методические особенности курса «Информатика».

Введение — вступление к основному тексту учебного пособия. Его цель — познакомить студента с сущностью проблемы и роли информации, информационных технологий и информационных ресурсов в информатизации общества.

Первый раздел — «Теоретическая информатика» — состоит из трех глав. Материал первой главы содержит основные понятия и определения информатики. Подробно раскрывается объект и предметная область

информатики. Вторая глава посвящается математическим методам, составляющим теоретическую основу информатики. Дается количественная оценка информации. Раскрываются системы счисления, формы представления и преобразования информации. Третья глава завершает раздел теоретической информатики. Здесь рассматриваются самые первоначальные сведения об алгоритмах, методах их разработки, программировании, языках программирования, жизненном цикле программного обеспечения.

Второй раздел — «Прикладная информатика» — начинается изложением в четвертой главе общей характеристики процессов сбора, передачи, обработки и хранения информации. В пятой главе приводятся сведения о технических средствах информационных процессов. Дается общая характеристика информационного процесса и способов его технической реализации. Подробно рассказывается о структуре и функциях основных блоков электронных вычислительных машин (ЭВМ) и истории развития электронных вычислительных средств. Шестая глава посвящена изучению персонального компьютера (ПК). Эта одна из центральных глав учебного пособия, раскрывающая состав и назначение узлов и устройств персональной ЭВМ (ПЭВМ). Здесь же приводятся основные сведения о работе ПК, организации и представлении данных, интерфейсе пользователя. Дается понятие о программных и графических оболочках и мультимедиа.

В третьем разделе — «Элементы информационных технологий» — рассмотрены примеры новых информационных технологий, которые позволяют студенту увидеть возможности приложений информатики как науки не только сегодня, но и в ближайшей перспективе. В седьмой главе даны основные сведения о базах и банках данных. Приведены понятия автоматизированных банков данных, схемы функционирования системы управления базами данных, организации поиска данных, администратора базы данных. Здесь же содержится материал по классификации, содержанию и применению пакетов прикладных программ (ППП) как одного из эффективных способов внедрения вычислительной техники в практику, по принципам построения и эксплуатации вычислительных и информационных сетей и автоматизированных рабочих мест (АРМ), приведен обзор проблем систем искусственного интеллекта, экспертных и геоинформационных систем. Из всего многообразия сетей для первоначального знакомства в главе девятой приведена глобальная сеть Интернет.

В заключении рассматриваются перспективы развития информатики и на ее методологической основе современных информационных технологий.

Опыт усвоения основ информатики показывает, что студенты испытывают большие трудности с терминологией по информатике и вычислительной технике. Поэтому в приложении приведен краткий глоссарий,

где представлена терминология, относящаяся к основополагающим понятиям информатики и смежных дисциплин, к проблемам автоматизации ввода, хранения и выдачи информации, организации информационного обеспечения пользователей.

Основными отличиями предлагаемого учебного пособия от существующих учебных пособий по информатике и вычислительной технике являются:

— систематическое изложение теоретической и прикладной информатики в соответствии с требованиями Государственного стандарта среднего профессионального образования России;

— ориентация на конкретные программы обучения студентов по дисциплине «Математика и информатика»;

— изложение материала с двух позиций: «информация — информационные технологии — информационный ресурс» и «модель — алгоритм — программа».

В результате изучения материала студент должен

— знать и уметь использовать:

функциональную организацию ЭВМ;

состав ПК;

создавать простые файлы с помощью ППП;

осуществлять первоначальную загрузку ПК, редактирование текстовых файлов, таблиц и графических изображений;

— иметь представление о:

месте и роли информатики в современном мире;

информатизации общества;

методах и средствах сбора, обработки, хранения, передачи и накопления информации;

технологии решения задач на ЭВМ;

современных информационных технологиях (банках и базах данных, ППП, АРМ, экспертных системах, телекоммуникационных сетях и др.).

Примерная программа дисциплины «Математика и информатика» [15], предназначенная для реализации государственных требований к минимуму содержания и уровню подготовки выпускников специальностей среднего профессионального образования базового уровня, предусматривает углубленное изучение отдельных разделов и тем. Поэтому некоторые темы учебного пособия написаны несколько шире программы, заложенной в образовательный стандарт. Это дает возможность учебному заведению использовать материал учебного пособия для более рационального построения рабочей учебной программы по информатике.

Автор считает своим долгом выразить искреннюю признательность рецензентам — доктору физ.-мат. наук, проф. В.А. Тупчеву и сотрудникам Сургутского нефтяного техникума (директор Л.М. Джебраилов) за ценные замечания, сделанные при рецензировании учебного пособия; а также сотрудникам и коллегам кафедры АСУ Обнинского института

атомной энергетики И.В. Алексеевой, А.В. Васяшину и Е.В. Крылову за большую помощь в отборе материала учебного пособия и дискуссии по его содержанию, О.М. Филипповой — за подготовку рукописи к печати.

Особую благодарность автор выражает ректору Сургутского государственного университета — профессору Г.И. Назину за внимание к работе над рукописью учебного пособия и В.А. Вахрушеву за помощь в издании.

Пожелания и замечания по содержанию книги направлять по адресу: 101430, Москва, ГСП-4, Неглинная ул., 29/14, издательство «Высшая школа».

Автор

ВВЕДЕНИЕ

Информационное общество имеет следующие основные признаки:

1. Большинство работающих в информационном обществе (около 80%) занято в информационной сфере, т.е. сфере производства информации и информационных услуг.

2. Обеспечены техническая, технологическая и правовая возможности доступа любому члену общества практически в любой точке территории и в приемлемое время к нужной ему информации (за исключением военных и государственных секретов, точно оговоренных в соответствующих законодательных актах).

3. Информация становится важнейшим стратегическим ресурсом общества и занимает ключевое место в экономике, образовании и культуре.

Информатизация — необходимое условие научно-технического, социального, экономического и политического прогресса в обществе. Неизбежность информатизации обусловлена следующими причинами:

беспрецедентным усложнением социально-экономических процессов в результате увеличения масштабов и темпов общественного производства, углубления разделения труда и его специализации в научно-технической революции;

необходимостью адекватно реагировать на возникающие проблемы в динамично меняющейся обстановке, присущей постоянно развивающемуся обществу;

повышением степени самоуправления предприятий, территорий, регионов.

Процесс перехода от постиндустриального общества к информационному происходит одновременно в различных странах, он характеризуется также и разными темпами развития. Первыми на этот путь встали в конце 50-х — начале 60-х гг. XX в. США, Япония и страны Западной Европы. В этих государствах начиная с 60 — 70-х гг. проводится политика повсеместной информатизации всех сфер деятельности человека. Были разработаны и приняты на государственном уровне программы информатизации с целью наиболее полного использования информационного ресурса для ускорения экономического, социального и

культурного развития общества. Предполагается, что США завершат переход к информационному обществу к 2020 г., Япония и основные страны Западной Европы — к 2030 — 2040 гг.

В СССР в 1989 г. разработана Концепция информатизации общества. По предварительным оценкам информатизация в России завершится к 2050 г. при условии стабилизации экономической и политической обстановки в стране. По мнению специалистов, любая страна, насколько бы индустриально развитой она ни была, перейдет в разряд стран третьего мира, если опоздает с информатизацией.

Если предшествующие этапы развития человечества длились каждый около трех веков, то ученые прогнозируют, что информационный этап продлится значительно меньше. Его срок существования ограничится, вероятно, сотней лет. Это означает, что основные регионы мира войдут в развитое информационное общество в XXI в. и в этом же веке начнется переход к постинформационному обществу.

Для информационного общества характерно обеспечение требуемой степени информированности всех его членов, возрастание объема и уровня информационных услуг, предоставляемых пользователю. Информационное общество в теоретическом аспекте характеризуется высококоразвитой информационной сферой (инфосферой), которая включает деятельность человека по созданию, переработке, хранению, передаче и накоплению информации.

В соответствии с учением В.И. Вернадского о ноосфере, инфосферу можно рассматривать как третью после техносферы и энергосферы этап преобразования биосферы в ноосферу — сферу разума. Можно утверждать, что у человечества нет другой альтернативы, которая позволила бы не только сохранить человека как биологический вид, но и обеспечить возможность дальнейшего поступательного развития общества.

Информатизация прежде всего охватывает все элементы рыночной инфраструктуры экономики:

- сеть оптовой и розничной торговли;
- товарные, фондовые, валютные биржи, биржи труда, ярмарки;
- сети банков и кредитно-финансовых учреждений;
- сеть независимых посреднических фирм и контор, оказывающих различные услуги, в том числе и информационные, субъектам рынка;
- складское и транспортное хозяйство;
- страховые компании;
- налоговую службу;
- систему подготовки и переквалификации кадров;
- аудиторскую службу и т.д.

Для изучения спроса и предложения (товаров, идей, видов обслуживания) создаются региональные сети самых различных форм (лаборатории изучения спроса, информационно-коммерческих и рекламно-маркетинговых центров, центров аналитических исследований и

научного прогнозирования и т.п.), специальные подразделения при крупных производственных, сбытовых и торговых предприятиях.

Важную роль в экономике играет сфера услуг. Она занимает доминирующее положение как в производстве валового продукта, так и в обеспечении занятости населения в регионе. Информатизация сферы услуг стимулирует развитие всех видов сервиса, особенно в торговле, финансово-кредитной сфере, связи, здравоохранении, организации досуга и отдыха населения, что положительно сказывается на производственной сфере, обеспечивает экономический рост, снижает напряженность на региональном рынке трудовых ресурсов. При этом в процессе информатизации возрастают потребности самой сферы услуг в информационных средствах, технологиях и услугах. Информатизация сферы услуг в экономике способна повысить производительность труда в этой сфере и тем самым обеспечить реальное безинфляционное увеличение доходов и рост благосостояния населения в течение длительного времени.

Информатизация бирж труда позволяет рационально использовать людские ресурсы региона, сбалансировать предложение и спрос на рабочую силу, предложить жителям региона работу в соответствии с их способностями и возможностями.

Приведенные примеры свидетельствуют, что информатизация проводится по различным направлениям экономики и социальной политики общества.

Научным фундаментом процесса информатизации общества является новая научная дисциплина — информатика. В широком смысле информатика — это наука об информационной деятельности, информационных процессах и их организации в человеко-машинных системах. Основными разделами информатики являются исследование и разработка информационных средств и технологий, программных средств и моделирование предметных областей.

РАЗДЕЛ I

ТЕОРЕТИЧЕСКАЯ ИНФОРМАТИКА

ГЛАВА 1

ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ ИНФОРМАТИКИ

В этой главе изложены основные понятия и определения информатики. Детально раскрываются фундаментальные понятия информации, данных, знаний, информатики, информационной технологии, информационного ресурса. Даются классификация и краткое пояснение автоматизированных информационных систем как объекта изучения информатики. Анализируется информационный ресурс в качестве предмета информатики. Все это помогает сразу проникнуть в содержательную часть информатики и облегчает понимание существа теоретической информатики. Краткое изложение истории развития информатики способствует становлению естественнонаучного понимания. Изучение материала первой главы дает возможность ответить на ряд вопросов: что такое информатика? Для чего она нужна? Каково содержание информатики? Какие задачи она решает?

Методические указания

В результате изучения материала первой главы учебного пособия студенты должны: усвоить определения объекта и предмета информатики; четко представлять смысл понятий информация, данные, знания; разобраться в содержании терминов «информационная технология» и «информационный ресурс» и запомнить определение и содержание естественнонаучной дисциплины «информатика», а также значение и необходимость информатики и ЭВМ в жизни общества. Уже в самом начале изучения дисциплины «информатика» студент должен

усвоить понятие канала связи как необходимого элемента между источником информации и ее получателем и роль в этом процессе ЭВМ.

Преподаватель обязан донести до студента первую главную мысль: появление науки «информатика» — это объективный процесс развития общества. И второе, что без внедрения информатизации общество не может развиваться дальше.

1.1. ТЕРМИНОЛОГИЯ ИНФОРМАТИКИ

Термин **информация** имеет множество определений. В «Энциклопедии кибернетики» «информация (лат. *informatio* — разъяснение, изложение, осведомленность) — одно из наиболее общих понятий науки, обозначающее некоторые сведения, совокупность каких-либо данных, знаний и т.п.». В широком смысле «информация» — это отражение реального мира; в узком смысле «информация» — это любые сведения, являющиеся объектом хранения, передачи и преобразования.

С практической точки зрения информация всегда представляется в виде сообщения. Информационное сообщение связано с источником сообщения, «получателем сообщений» и каналом связи (рис. 1.1).

Сообщение от источника к приемнику передается в материально-энергетической форме (электрический, световой, звуковой сигналы и т.д.). Человек воспринимает сообщения посредством органов чувств. Приемники информации в технике воспринимают сообщения с помощью различной измерительной и регистрирующей аппаратуры. В обоих случаях с приемом информации связано изменение во времени какой-либо величины, характеризующей состояние приемника. В этом смысле информационное сообщение можно представить функцией $x(t)$, характеризующей изменение во времени материально-энергетических параметров физической среды, в которой осуществляются информационные процессы.

Функция $x(t)$ принимает любые вещественные значения в диапазоне изменения времени t . Если функция $x(t)$ непрерывна, то имеет место непрерывная или аналоговая информация, источником которой обычно являются различные природные объекты (например, температура, давление и влажность воздуха), объекты технологических производственных процессов (например, нейтронный поток в активной зоне, давление

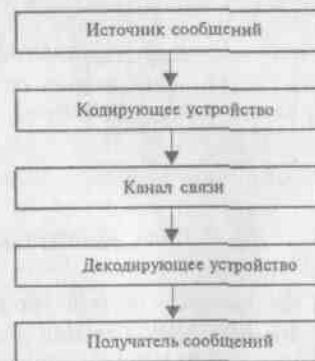


Рис. 1.1. Общая схема передачи информации

и температура теплоносителя в контурах ядерного реактора) и др. Если функция $x(t)$ дискретна, то информационные сообщения, используемые человеком, имеют характер дискретных сообщений (например, сигналы тревоги, передаваемые посредством световых и звуковых сообщений, языковые сообщения, передаваемые в письменном виде или с помощью звуковых сигналов; сообщения, передаваемые с помощью жестов, и др.).

В современном мире информация, как правило, обрабатывается на вычислительных машинах. Поэтому информатика тесно связана с инструментарием — вычислительной машиной.



Рис. 1.2. Классы вычислительных машин

Компьютер — устройство преобразования информации посредством выполнения управляемой программой последовательности операций. Синоним компьютера — вычислительная машина, чаще электронная вычислительная машина (ЭВМ).

В зависимости от вида перерабатываемой информации вычислительные машины (ВМ) делят на два основных класса: аналоговые и цифровые (рис. 1.2). *Аналоговая вычислительная машина (АВМ)* — это машина, оперирующая информацией, представленной в виде непрерывных изменений некоторых физических величин. При этом в качестве физических переменных используются сила тока электрической цепи, угол поворота вала, скорость и ускорение движения тела и т.п. Используя тот факт, что многие явления в природе математически описываются одними и теми же уравнениями, АВМ позволяют с помощью одного физического процесса моделировать различные процессы.

Цифровая вычислительная машина (ЦВМ) — машина, оперирующая информацией, представленной в дискретном виде. В настоящее время разработаны методы численного решения многих видов уравнений, что дало возможность решать на ЦВМ различные уравнения и задачи с помощью набора простых арифметических и логических операций. По-

этому если АВМ обычно предназначаются для решения определенного класса задач, т.е. являются специализированными, то ЦВМ, как правило, является универсальным вычислительным средством. Наибольшее распространение получили электронные вычислительные машины, выполненные с использованием новейших достижений электроники.

Для представления в ЭВМ дискретной информации применяется *алфавитный способ*, основанный на использовании фиксированного конечного набора символов любой природы, называемого *алфавитом*. Примерами алфавитов могут служить алфавиты естественных человеческих языков, совокупность десятичных цифр, любая другая упорядоченность знаков, предназначенная для образования и передачи сообщений. Символы из набора алфавита называются *буквами*, а любая конечная последовательность букв — *словом* в этом алфавите. При этом не требуется, чтобы слово обязательно имело языковое смысловое значение.

Все процессы, происходящие в вычислительной системе, связаны непосредственно с различными физическими *носителями информационных сообщений*, а все узлы и блоки этой системы являются физической сферой, в которой осуществляются информационные процессы. Процесс преобразования информации часто требует представлять буквы одного алфавита средствами (буквами, словами) другого алфавита. Такое представление называется *кодированием*. Процесс обратного преобразования информации относительно ранее выполненного кодирования называется *декодированием*.

Очень широко используется еще один термин: *данные* (лат. *data*). Этот термин принято применять в отношении информации, представленной в виде, позволяющем хранить, передавать или обрабатывать ее с помощью технических средств. Поэтому наряду с терминами *ввод информации, обработка информации, хранение информации, поиск информации* используются термины *ввод данных, обработка данных, хранение данных* и т.п.

Если же отвлечься от самых общих определений, то одна большая группа специалистов под информатикой понимает науку, связанную с: разработкой вычислительных машин и систем, технологий их создания; разработкой математических моделей естествознания и общественных явлений с целью их строгой формализации; обработкой данных, созданием численных и логических методов решения задач, сформулированных на этапе построения математической модели; разработкой алгоритмов решения задач управления, расчета и анализа математических моделей; программированием алгоритмов, созданием программного обеспечения ЭВМ.

Другая, не менее большая группа специалистов считает, что информатика — это то, что связано с применением ЭВМ, их разработкой и созданием программ для них.

Более строгое определение информатики целесообразно дать после детального описания объекта и предмета информатики.

1.2. ОБЪЕКТ И ПРЕДМЕТНАЯ ОБЛАСТЬ ИНФОРМАТИКИ

Объектом информатики выступают автоматизированные, основанные на ЭВМ и телекоммуникационной технике, информационные системы (ИС) различного класса и назначения. Информатика изучает все стороны их разработки, проектирования, создания, анализа и использования на практике. *Информационные технологии (ИТ)* — это машинизированные (инженерные) способы обработки семантической информации — данных и знаний, которые реализуются посредством автоматизированных информационных систем (АИС).

В настоящее время АИС получили широчайшее распространение. Классификация АИС осуществляется по ряду признаков, и в зависимости от решаемой задачи можно выбрать разные признаки классификации. При этом одна и та же АИС может характеризоваться одним или несколькими признаками. В качестве признаков классификации АИС используются: область применения, охватываемая территория, организация информационных процессов, направление деятельности, назначение, структура и др.

Классификация АИС по направлению деятельности показана на рис. 1.3.

В промышленной сфере превалирует отраслевой характер иерархии АИС. По территориальному признаку классификация АИС приведена на рис. 1.4.

В зависимости от организации информационных процессов АИС делятся на два больших класса: управляющие и информационные. В информационных системах управление отсутствует (автоматизированные системы научных исследований — АСНИ, «Библиотека», системы автоматизированного проектирования — САПР, экспертные системы — ЭС и др.).

По сфере применения АИС классифицируются следующим образом: административные, производственные, учебные, медицинские, военные, метеорологические, экологические, криминалистические и др. Назначение и структура построения АИС характеризуются наличием соответствующих подсистем (рис. 1.5). Этот класс АИС является исторически одним из первых на производстве.

АИС различаются также по уровню развития в зависимости от поколений ЭВМ, на которых они базируются. Разнообразие ИТ и АИС постоянно растет.

Следует выделить ИТ и АИС, с помощью которых регулируются микросоциальные процессы. Это ИТ и АИС денежно-кассовых опера-

ций, распределения мест на транспорте, в гостиницах, метеорологической информации и т.п. Они также относятся к объекту информатики.

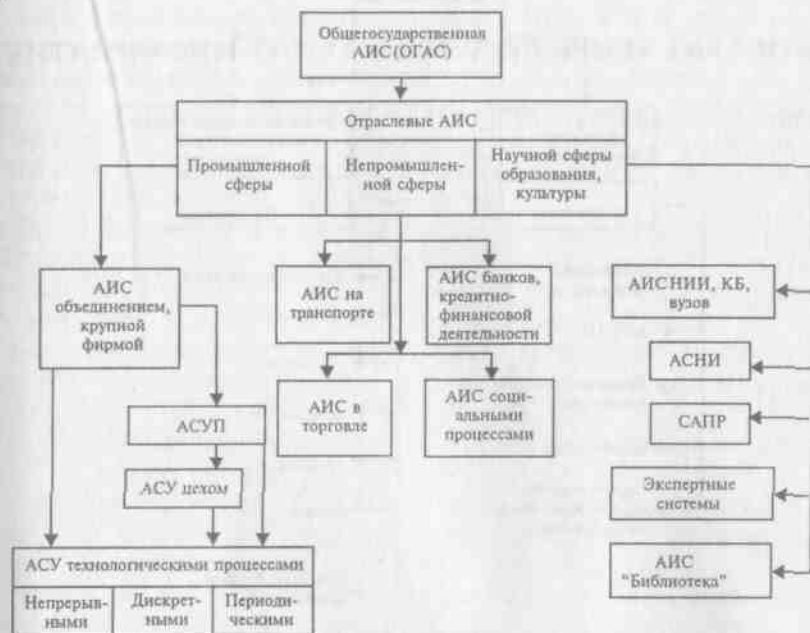


Рис. 1.3. Классификация АИС по направлению деятельности



Рис. 1.4. Классификация АИС по территории

Информационные технологии — это машинизированные способы обработки, хранения, передачи и использования информации в виде знаний. Они включают два основных элемента — машинный и человеческий (социальный), причем последний выступает главным. Подчеркнем, что объектом информатики выступают не сами по себе ЭВМ как программно-технические комплексы, а ИТ, воплощением которых служат компьютеризированные, или человеко-машинные, точнее, социотехнические системы. В отличие от производственных, энергопре-

БИБЛИОТЕКА
Колледжа предпринимательства
и социального управления

образующих, ИТ (объект информатики) относятся к социальным, знание-преобразующим, технологиям.



Рис. 1.5. Классификация АИС на предприятии

Понятие технологии вообще включает комплекс научных и инженерных знаний, воплощенных в приемах труда, наборах материальных, технических, энергетических, трудовых факторов производства, способов их соединения для создания продукта или услуги, отвечающих определенным требованиям, стандартам. В таком понимании термин *технология* неразрывно связан с механизацией производственного или непроизводственного (социального) процесса. Последнее важно подчеркнуть для обозначения точных исторических границ информатики: информатики не могло быть в домашний период обработки и представления знаний, когда не было ИТ. Переработка информации с помощью ЭВМ и выработка новых знаний, соотносенных с целями пользователей, — функциональное назначение ИТ.

Если информация — категория всеобщая, присущая всем историческим периодам, то информатика — категория конкретно-историческая, присущая лишь современному и будущим историческим периодам. Ранее, когда не было ИТ, т.е. не было объекта информатики, информация перерабатывалась не машинными, рутинными способами. На базе ЭВМ рождаются ИС, реализующие специальные технологии сбора, переработки, передачи и применения информации, т.е. рождается

объект информатики, что является основным содержанием современной научно-технической революции.

В будущем возникнут (и уже возникают) сложнейшие проблемы взаимодействия искусственного интеллекта (ИИ) в виде сверхсложных вычислительных сетей пятого поколения с такими элементами, как супер-ЭВМ, актуализируемые базы знаний, персональные компьютеры (ПК), новейшие локальные и глобальные телекоммуникационные системы, и сложившегося естественного социального интеллекта, промышленных роботов массового применения и естественной рабочей силы. Это рождает сверхсложные социальные технологии.

Таким образом, объект информатики, охватывающий все элементы ИТ: технические средства, математическое, алгоритмическое, программное, лингвистическое обеспечения, средства связи во взаимодействии с людьми, резко усложняется. Особенно сложные инженерные проблемы информатики возникают применительно к телематическим системам в виде «безлюдных» производств и технологий, в которых интегрированы организационные человеко-компьютерные системы, робототехнические комплексы и новейшие средства связи. Это уже информдинамические объекты — новые исторические феномены — с заложенными в них творческими системами (*Creative Systems*), в процессе создания и эксплуатации которых мы попадаем в исключительную зависимость от теории — нового комплекса знаний и нового мышления.

Один и тот же объект может изучаться различными науками. *Объект познания* — это некий фрагмент реального мира, а его *предмет* — это выбранная для исследования методами данной науки сторона, грань, аспект объекта.

Информатизация общества в части материально-технической базы, математического и программного обеспечения ИТ изучается различными науками: кибернетикой, системотехникой, теорией информации, а в части формирования функциональных подсистем — различными общественными науками: экономикой, правоведением, психологией. В формировании ИТ участвуют и науки, относящиеся к той или иной автоматизируемой области: медицина, когда речь идет о внедрении ЭВМ в здравоохранение; педагогика (компьютеризация учебного процесса); военные науки (использование ЭВМ в военном деле), экономика и т.д. Каждая из указанных наук рассматривает компьютеризацию со своей стороны, прилагает к ней свои законы и принципы.

А какую же сторону рассматриваемого объекта выбирает информатика, делая ее своим предметом? Она выбирает содержательную, смысловую сторону создания и функционирования информационных систем и технологий, связанную с их сущностью, социальной отдачей, полезностью, местом в общественных системах, историческим значением как фактора радикального прогресса и выхода общества на качественно новые исторические рубежи.

Переход к ИТ как технической базе автоматизированных информационно-управляющих систем обнажил и до крайности обострил проблемы, относящиеся ко всему технологическому циклу сбора, переработки и применения информации в планово-управленческих, познавательных и других процессах, выдвинул на первый план общие проблемы содержания информационных процессов и значения ИТ (рис. 1.6). Говорить о воздействии науки на что-либо вне информационного процесса бессмысленно. Научная идея должна превратиться в информацию, т.е. быть закодированной, переданной по каналу связи, принятой адресатом, чтобы ее можно было применить на практике. Чем больше потенциал знаний, тем важнее задача развития информационно-коммуникативных сфер народного хозяйства. Знания должны определенным образом фиксироваться, трансформироваться, распределяться, приниматься и перерабатываться.



Рис. 1.6. Основные этапы технологического процесса в информационных системах

Информационные технологии и выступили новым средством превращения знаний в информационный ресурс (ИР) общества, его новым движущим фактором, стали средством его эффективного использования. Информационный ресурс стал основным ресурсом человечества, главной ценностью современной цивилизации. Но возникли и сложные проблемы, относящиеся к роли, механизму функционирования, социальным последствиям использования ИР. Для их решения и появилась новая наука — информатика.

Предметом информатики как новой фундаментальной науки выступает ИР — его сущность, законы функционирования, механизмы взаимодействия с другими ресурсами общества и воздействия на социальный прогресс. Переход на уровень ИР в его содержательной трактовке означает переход к изучению внутренних связей и закономерностей социальной динамики, основанной на использовании ИТ.

Информатика как наука о законах получения, передачи и использования ИР в общественной практике подводит теоретический фундамент под использование ЭВМ и автоматизированных систем, которые и

предназначены для усиления информационных процессов в обществе, использования ИР. Речь идет прежде всего о специальных ИР, основанных на компьютерной технике и реализующих ИТ, т.е. инженерную обработку знаний (*Knowledge Engineering*).

Таким образом, предметом информатики является информационный ресурс как симбиоз знания и информации. Он выступает в качестве предмета новой науки и с содержательной, и с формально-математической, и с технической стороны. Необходимо разграничивать предмет информатики как фундаментальной науки, ее объект и инструментарий: основанные на ЭВМ вычислительные системы, программы, сети связи и т.д. Без ЭВМ нет информатики, но нельзя объявлять информатику наукой об ЭВМ. Конечно, практическая необходимость в информатике возникла в связи с использованием ЭВМ. Но, «оттолкнувшись от ЭВМ», информатика во главу угла ставит новые понятия — ИР и его социальную полезность, отдачу. Поэтому по аналогии с термодинамикой информатику можно назвать информдинамикой — наукой о развитии социальных систем под воздействием ИР (семантической информации).

Информатика делится на две части: теоретическую и прикладную информатику.

Теоретическая информатика рассматривает все аспекты разработки автоматизированных информационных систем: их проектирования, создания и использования не только с формально-технической, но и содержательной стороны, а также комплекс экономического, политического и культурного воздействия на социальную динамику. В орбиту анализа теоретической информатики попадают и традиционные системы преобразования информации и распространения знаний: средства и системы массовой информации, система лекционной пропаганды, кино, театры, справочные службы и т.д. Но теоретическая информатика рассматривает их с определенной стороны — с позиций получения и использования ИР, форм и способов воздействия указанных систем на общественный прогресс, возможной их технологизации.

Теоретическая информатика изучает ИР, законы его функционирования и использования как движущей силы социального прогресса, а также общие, фундаментальные проблемы ИТ как исторического феномена, выводящего общество на новую ступень развития.

Прикладная информатика изучает конкретные разновидности ИТ, которые формируются с помощью специальных ИС (управленческих, медицинских, обучающих, военных, криминалистических и др.). Очевидно, что такие ИТ, как, например, управление (АСУП, АСУТП), проектные разработки (САПР) или криминалистика, имея общие черты, в то же время существенно различаются между собой. Разные операции и процедуры, различное оборудование, специализация критериев и показателей, разная степень замкнутости информационных контуров, даже разные информационные носители, т.е. разные информационные среды, — все это стано-

вится объектом изучения конкретных функциональных и отраслевых информатик. Так рождаются ветви прикладной информатики, обслуживающие создание проектирующих систем, экспертных систем, диагностических комплексов, управляющих и других функциональных систем. Возникли также отраслевые ветви информатики, обслуживающие информатизацию разных сфер социальной и экономической практики: промышленность, науку, медицину, связь и т.д. Поэтому наряду с теоретической информатикой развиваются ее конкретные ветви: экономическая информатика, медицинская информатика, военная информатика и др.

1.3. ВИДЫ И СВОЙСТВА ИНФОРМАЦИИ

Остановимся более подробно на раскрытии понятия информация. Рассмотрим следующий перечень:

- генетическая информация;
- геологическая информация;
- синоптическая информация;
- ложная информация (дезинформация);
- полная информация;
- экономическая информация;
- техническая информация и т.д.

Наверное, каждый согласится, что в этом перечне приведены далеко не все виды информации, так же как и с тем, что от приведенного перечня мало проку. Этот перечень не систематизирован. Для того чтобы классификация по видам была полезной, она должна быть основана на некоторой системе. Обычно при классификации объектов одной природы в качестве базы для классификации используется то или иное свойство (может быть набор свойств) объектов. Как правило, свойства объектов можно разделить на два больших класса: внешние и внутренние свойства.

Внутренние свойства — это свойства, органически присущие объекту. Они обычно «скрыты» от изучающего объект и проявляют себя косвенным образом при взаимодействии данного объекта с другими.

Внешние свойства — это свойства, характеризующие поведение объекта при взаимодействии с другими объектами.

Поясним сказанное на примере. Масса является внутренним свойством вещества (материи). Проявляет же она себя во взаимодействии или в ходе некоторого процесса. Отсюда появляются такие понятия физики, как гравитационная масса и инерциальная масса, которые можно было бы назвать внешними свойствами вещества.

Подобное разделение свойств можно привести и для информации. Для любой информации можно указать три объекта взаимодействия: источник информации, приемник информации (ее потребитель) и объект или явление, которые данная информация отражает. Поэтому можно

выделить три группы внешних свойств, важнейшими из которых являются свойства информации с точки зрения потребителя.

Качество информации — обобщенная положительная характеристика информации, отражающая степень ее полезности для пользователя.

Показатель качества — одно из важных положительных свойств информации (с позиции потребителя). Любое отрицательное свойство может быть заменено обратным ему, положительным.

Чаще всего рассматривают показатели качества, которые можно выразить числом, и такие показатели являются количественными характеристиками положительных свойств информации.

Как явствует из приведенных определений, для того, чтобы определить набор важнейших показателей качества, необходимо оценить информацию с точки зрения ее потребителя.

Потребитель на практике сталкивается со следующими ситуациями: часть информации соответствует его запросу, его требованиям и такую информацию называют релевантной; а часть — нет, и ее называют нерелевантной; вся информация релевантна, но ее недостаточно для нужд потребителя; если полученной информации достаточно, то такую информацию естественно называть полной;

полученная информация несвоевременная (например, устарела);

часть информации из признанной потребителем релевантной может оказаться недостоверной, т.е. содержащей скрытые ошибки (если часть ошибок потребитель обнаруживает, то он испорченную информацию попросту относит к нерелевантной);

информация недоступна;

информация подвержена «нежелательному» использованию и изменению со стороны других потребителей;

информация имеет неудобные для потребителя форму или объем.

Обзор приведенных ситуаций позволяет сформулировать следующие определения свойств информации.

Релевантность — способность информации соответствовать нуждам (запросам) потребителя.

Полнота — свойство информации исчерпывающе (для данного потребителя) характеризовать отображаемый объект и / или процесс.

Своевременность — способность информации соответствовать нуждам потребителя в нужный момент времени.

Достоверность — свойство информации не иметь скрытых ошибок.

Доступность — свойство информации, характеризующее возможность ее получения данным потребителем.

Защищенность — свойство, характеризующее невозможность несанкционированного использования или изменения.

Эргономичность — свойство, характеризующее удобство формы или объема информации с точки зрения данного потребителя.

Кроме того, информацию можно классифицировать с точки зрения ее использования на следующие виды: политическую, техническую, биоло-

гическую, химическую и т.д. Это, по существу, классификация информации по потребителю.

Наконец, обобщенно характеризуя качество информации, часто используют следующее определение. Логическая, адекватно отображающая объективные закономерности природы, общества и мышления — это есть *научная информация*. Заметим, что последнее определение характеризует не взаимоотношение «информация — потребитель», а взаимоотношение «информация — отражаемый объект / явление», т.е. это уже следующая группа внешних свойств информации. Здесь наиболее важным является свойство адекватности.

Адекватность — свойство информации однозначно соответствовать отображаемому объекту или явлению. Адекватность оказывается для потребителя внутренним свойством информации, проявляющим себя через релевантность и достоверность.

Среди внутренних свойств информации важнейшими являются объем (количество) информации и ее внутренняя организация, структура. Об объеме и количестве информации речь пойдет в следующих главах, а по способу ее внутренней организации информацию делят на две группы:

1. Данные или простой, логически неупорядоченный набор сведений.
2. Логически упорядоченные, организованные наборы данных.

Упорядоченность данных достигается наложением на данные некоторой структуры (отсюда часто используемый термин — структура данных).

Во второй группе выделяют особым образом организованную информацию — *знания*. Знания в отличие от данных представляют собой информацию не о каком-то единичном и конкретном факте, а о том, как устроены все факты определенного типа.

Наконец, вне поля нашего зрения оказались свойства информации, связанные с процессом ее хранения. Здесь важнейшим свойством является *живучесть* — способность информации сохранять свое качество с течением времени. К этому еще можно добавить свойство *уникальности*. Уникальной называют информацию, хранящуюся в единственном экземпляре.

Таким образом, мы описали основные свойства информации, а соответственно и определили базу для классификации ее по видам.

1.4. ИНФОРМАТИЗАЦИЯ ОБЩЕСТВА

В истории человечества было несколько промышленных и информационных переворотов. Первый промышленный переворот основывался на паровом двигателе, затем на электромоторе и двигателе внутреннего сгорания, т.е. на энергопреобразующих машинах. С ним связаны индустриализация общества, переход к индустриальной цивилизации. Вто-

рой промышленный переворот связан с овладением человечеством атомной энергией и проникновением в космос — современная научно-техническая революция (НТР).

Перевороты в информационной сфере были уже трижды. При первом перевороте — появлении письменности — возник специальный механизм фиксации и распространения информации во времени и пространстве. Появились документированная информация и информационно-накопительные центры в виде библиотек, архивов. Второй информационный переворот в истории связан с книгопечатанием. Эта революция состояла в тиражировании документированной информации с помощью специальной машины — печатного станка. Если появление письменности коррелирует с ростом науки, просвещения и искусства, то типографский станок породил книжно-журнальную и газетную индустрию — основу дальнейшего культурного прогресса человечества, а также индустрию документооборота. Прорыв состоял в производстве «бумажной» информации, возникновении механизма ее размножения и хранения. Но переработка и использование информации по-прежнему зависели от физиологических возможностей человека с его способностью за секунду воспринимать 8 — 9 единиц информации и в лучшем случае произвести 15 логических операций.

Скачкообразное увеличение скорости распространения информации, вызванное появлением электрического телеграфа, телефона, радио, телевидения, а также ростом на 2 — 3 порядка транспортных скоростей в связи с появлением автотранспорта и авиации, конечно, привело к огромной интенсификации информационных потоков и коммуникативных процессов (третий информационный переворот). Но это существенно не отразилось на скорости выработки сигналов обратной связи, которые по-прежнему зависели от способностей «невооруженной» головы человека.

Таким образом, человечество столкнулось с новым, невиданным ранее в истории, противоречием между машинной вооруженностью рук (в середине XX в. 99% всей физической работы выполнялось машинами) и «ремесленной вооруженностью» головы (интеллекта). Информационно-управляющие возможности людей, ограниченные физическими рамками, перестали соответствовать вещественно-энергетическим возможностям производства, ставшими, по существу, безграничными в силу использования машинной техники. Постепенно разрастаясь, это противоречие вело к информационно-организационному кризису. Это противоречие и разрешается путем создания информационных, компьютеризированных технологий в области переработки и использования знаний.

С первым промышленным переворотом связано появление промышленных технологий, со вторым — появление информационных технологий, насыщение которыми народного хозяйства и означает информатизацию общества. Переворот в информационной сфере, ее индуст-

риализация на базе ЭВМ составляет главное содержание современной НТР, в то же время можно считать третьим в истории великим информационным переворотом после изобретения письменности и книгопечатания.

Альтернативы информатизации нет. Это объективный этап социального прогресса во всех областях, прежде всего в экономике, управлении, науке и технологии. Важно понимать крайности в понятии «информатизация». С одной стороны, нельзя отождествлять информатизацию с компьютеризацией, так как в этом случае все дело сведется к созданию технической базы, насаждению ЭВМ в неподготовленную для них социальную среду, нагромождению бесполезных с хозяйственной точки зрения пирамид из электроники. С другой стороны, нельзя к информатизации сводить все социально-экономические преобразования — слишком широкое ее понимание, нивелирующее значение других шагов и направлений социальной перестройки.

Информатизацию общества следует понимать как создание и развитие *информационной среды*: комплекса условий и факторов, обеспечивающих наилучшие условия функционирования ИР с учетом автоматизированных способов их переработки и использования в целях социального прогресса. Можно сказать и иначе: информатизация сводится к формированию ИТ и созданию условий для эффективного их использования в различных общественных системах.

Таким образом, информатизация предполагает более широкий подход к компьютеризации, включающий преобразование всего комплекса средств и условий развертывания информационных процессов: создание соответствующей технической базы, модернизацию организационно-экономических, юридических «человеческих факторов». Иными словами, это целостный процесс формирования новой автоматизированной сферы как необходимого условия эффективного задействования ЭВМ, их сетей, интегрированных АСУ, робототехнических комплексов, банков данных и т.д. Речь идет о программируемом изменении информационной основы функционирования различных общественных систем и подсистем, замене в допустимых пределах бумажной информации человеко-машинными диалоговыми системами, создании новых, несравненно более эффективных моделей деятельности людей.

До начала 70-х гг., когда компьютеризация была поверхностной и очаговой, потребность в информатизации общества остро не ощущалась. Резкое расширение процессов компьютеризации, массовое развитие микропроцессорной техники и особенно наметившийся переход к пятому поколению ЭВМ сделали информатизацию насущно необходимой. Информатизация — это как бы надстроечный процесс, происходящий на базе компьютеризации, т.е. индустриализации непродуцированной сферы народного хозяйства, процесс формирования новой, автома-

тизированной среды зарождения знаний, их переработки, распространения и превращения в силу, в материальный фактор.

Техническая база информатизации — это компьютерные и телекоммуникационные системы и сети, которые должны составить «ядро» экономики, точнее — производственного аппарата будущего общества. Такой аппарат будет включать роботы и роботизированные производства, обрабатывающие центры, гибкие производственные системы, безлюдные участки, цехи, предприятия и, конечно, новые организационно-управленческие комплексы и системы связи.

Стержень всей информатизации — создание новых информационных структур. Основой новых информационных структур являются технические структуры. Создание и развитие технической базы информатизации — конечно, приоритетная задача, без решения которой нельзя двигаться вперед. Вычислительные системы пятого поколения будут иметь следующие функциональные блоки: базу знаний, машины решения задач и логических выводов, интеллектуальный интерфейс. Особенно заметна разница между ЭВМ пятого поколения и машинами предыдущих поколений на уровне интерфейса — способов взаимодействия с человеком. Последние будут приспособлены к человеку в отличие от «машинного» интерфейса современных ЭВМ, заставляющего человека приспосабливаться к машине и говорить не на своем естественном, а на машинном, искусственном языке.

С точки зрения пользователей, системы пятого поколения ЭВМ должны соответствовать следующим принципам:

- простота применения, когда от пользователя не требуется профессиональных знаний;
- моделирование «человеческих» функциональных возможностей, таких, как построение доказательств и принятие решений;
- автоматический выбор релевантных запросов данных из машинных хранилищ большого объема;
- гибкость конфигурации, обеспечивающая приспособленность ЭВМ к выполнению широкого диапазона работ;
- автоматизация программирования;
- высокая надежность.

Важнейшим аспектом информатизации является формирование и развитие индустрии информатики — весьма специфической сферы народного хозяйства. Здесь необходимо учитывать влияние следующих факторов:

1. Наличие и дальнейшее развитие всех звеньев, непосредственно реализующих переработку информации: вычислительные центры (ВЦ) и их сети, фонды алгоритмов и программ, базы данных (БД) и базы знаний (БЗ), исследовательские и проектные институты по созданию ИС, организации по обслуживанию ВЦ и т.д. Все это превращается в особую ин-

фраструктуру, обслуживающую сферу хозяйства, требующую единой технической политики и организационного оформления.

2. Создание производства и целых отраслей по выпуску средств ВТ и автоматизации: ЭВМ, элементная база, средства телекоммуникации, периферийное оборудование, магнитные носители, материалы и т.п.

3. Создание предприятий по выпуску вспомогательного оборудования и материалов, средств оргтехники, средств магнитной записи, оборудования по выработке бумажных носителей и т.п.

4. Создание организаций и предприятий по разработке математического и программного обеспечения, поскольку создание программных средств переводится на промышленную основу.

5. Разработка и создание системы и сети связи и коммуникаций, без которых немислим машинно-информационный процесс.

Рост индустрии информатики резко опережает рост других отраслей народного хозяйства. По числу занятых, объемам продукции она уже в ряде стран выдвинулась на одно из первых мест. Этой исключительно динамичной, технически развитой, наукоемкой сфере народного хозяйства придается стратегическое значение. Для обеспечения ее развития нужна мобилизация огромных ресурсов: финансовых, материально-технических, научных, человеческих. Эта сфера деятельности уже превратилась в область массового приложения труда.

В настоящее время стало ясно следующее: чтобы та или иная страна могла занять достойное место в мире в XXI в. и на равных участвовать в экономическом соревновании с другими странами, она должна перестраивать и приспособлять свои структуры, приоритеты, ценности, институты к требованиям индустриальной ИТ. Экономические позиции той или иной страны в начале — середине XXI в. будут определяться такими новаторскими технологиями, как термоядерный синтез, биотехнология, плазменные процессы, космическая связь и др. Но их развитие, в свою очередь, зависит от информатики. Нельзя представить себе эффективное и надежное функционирование, например, атомных станций или космической связи без высокотехнологического, т.е. компьютерного информационно-организационного, обеспечения.

Страны Запада и Япония уже два десятилетия разрабатывают проблемы информатизации общества. В последние годы там развернуты соответствующие общенациональные программы, поддерживаемые огромными государственными субсидиями. Информатизация считается прорывом в будущее. На это пошли все развитые страны, придав информатизации высшие приоритеты, подчинив этой цели основные ресурсы и усилия. Из объекта теоретического анализа информатизация превратилась в критерий оценки могущества и фактор выживания той или иной страны в борьбе за экономическое и политическое превосходство, стала важнейшим ориентиром для выработки внутренней и внешней стратегии государства.

В настоящее время объем расходов США на информатизацию (создание, производство, монтаж, использование ЭВМ, информационных сетей и систем разного уровня, БД и т.д.) достиг нескольких сот миллиардов долларов в год и превышает объем военных расходов. Американские системы компьютерной оборонной инициативы (КОИ) и особенно стратегической оборонной инициативы (СОИ) базируются на развитых ИТ, которые, в свою очередь, предполагают информатизацию управления, науки, проектных разработок, всех систем выработки, принятия и поддержки решений.

Известно наше отставание от США в количестве и качестве парка ЭВМ, особенно суперЭВМ и ПЭВМ, и в развитии коммуникационных систем. Такое положение опасно, особенно если учесть оборонные аспекты информатизации.

Очень часто в литературе по информатике отмечается только наше отставание от Запада и США, совершенно замалчивается опыт СССР и советских ученых и специалистов в области информатизации. Этот опыт должен быть осмыслен, проанализирован, и из него необходимо сделать правильные выводы для условий сегодняшней действительности.

В конце 60-х годов в СССР под руководством В.М. Глушкова (Институт кибернетики Академии наук Украины), подчеркнем, в те же годы, что и на Западе, был разработан проект создания общегосударственной автоматизированной системы (ОГАС). Согласно концепции ОГАС все ее территориальные и отраслевые звенья должны были создаваться на единой научно-методической основе. Суть этой основы выражалась в системном (комплексном) подходе, обеспечивающем целевую, проблемную, информационную и программно-техническую совместимость всех звеньев управления народным хозяйством. Комплексный подход состоит из возможности построения системы управления как единого целого путем определения главной деятельности системы и последующего развертывания ее в иерархическую структуру целей методом декомпозиции.

Осуществление такого подхода оказалось нереальным по следующим причинам:

- огромная масштабность и трудоемкость;
- отсутствие в 60-х — начале 70-х годов математического аппарата системотехники;

- отсутствие технических и программных средств;

- сложившаяся в те годы организация и культура управления.

Одним словом, автоматизировать сразу все на голом месте, игнорируя естественную эволюцию любой системы — ее постепенное наращивание и усложнение, — оказалось невозможным. В результате многолетних работ на разных территориях страны появилось множество информационно разобщенных ведомственных систем, функционирующих независимо друг от друга. Хотя, казалось бы, что в силу привязки всех от-

раслевых звеньев ОГАС к той или иной территории при формировании своих информационных баз они должны были использовать всестороннюю информацию о «своей» территории. Однако преобладание отраслевых (ведомственных) интересов над территориальными и ограниченность полномочий местных органов власти в вопросах управления территориями исключали появление таких стимулов.

В концепции информатизации нашей страны в конце 80-х годов указанные выше недостатки были в большей части устранены и во главу информатизации был поставлен региональный принцип: создание развитой информационной среды будет происходить регионально по мере готовности регионов к решению соответствующих проблем.

К этим проблемам развития информационной сферы регионов относятся следующие: проблемы индустриализации получения и обработки информации, психологические, правовые, экономические и социальные.

Индустриализация получения и обработки информации означает создание и развитие крупного машинного производства в информационной сфере. Эти проблемы порождены противоречиями между необходимостью современного использования во всех сферах человеческой деятельности больших объемов информации и невозможностью оперативно формировать такие объемы с помощью традиционных информационных средств, технологий и систем связи.

К первоочередным проблемам информатизации следует отнести проблему готовности населения к переходу в информационное общество психологическую проблему. Этот переход в настоящее время затрудняется низким уровнем информационной культуры населения, недостаточной компьютерной грамотностью, а отсюда и низкими информационными потребностями, а также отсутствием желания их развивать. Наблюдается невосприимчивость экономики управления на всех уровнях к результатам НТР и прежде всего в инфосфере. Психологический аспект проблемы определяется совместимостью человека и новой информационной техники и технологии.

Здесь необходимо еще раз подчеркнуть, что информатизация общества предполагает организацию компьютерного ликбеза населения, подготовку и переподготовку кадров — специалистов по ЭВМ и неспециалистов, т.е. пользователей — профессионалов в области программирования и ВТ, компьютеризацию всех звеньев образования от начальной школы до вуза и системы послевузовского образования, создания огромной сети по переквалификации работников, формирование, особенно у молодежи, новой информационной культуры, расширение математического образования, преодоление барьеров на пути к ПЭВМ, машинным языкам и т.д.

Правовые проблемы возникают в связи с превращением информации в основные ресурсы развития общества. Возникают новые виды правонарушений, которые свойственны лишь информационной сфере.

Таким образом, информационное общество — это общество, структуры, техническая база и человеческий потенциал которого приспособлены для оптимального превращения знаний в ИР и переработки последнего с целью перевода пассивных форм (книги, статьи, патенты и т.п.) в активные (модели, алгоритмы, программы, проекты). Но особое значение для активизации информационного потенциала общества имеет создание современных БЗ. Это достигается на путях качественного преобразования традиционных баз данных (БД), рожденных ранними поколениями ЭВМ до появления искусственного интеллекта (ИИ), в БЗ.

В отличие от БД, содержащих сведения о качественных характеристиках конкретных объектов, БЗ содержат концептуальные, понятийные знания, выраженные на естественном языке в терминах предметной области, т.е. знания о стоящих за этими терминами классах объектов, их свойствах и логических связях, которыми может оперировать машина логического вывода как элемент ИИ.

1.5. КРАТКАЯ ИСТОРИЯ РАЗВИТИЯ ИНФОРМАТИКИ

Информатика как наука стала развиваться с середины нашего столетия, что связано с появлением ЭВМ и начинающейся компьютерной революцией. Появление вычислительных машин в 50-е годы создало для информатики необходимую ей аппаратную поддержку, или, иначе говоря, благоприятную среду для ее развития как науки. Всю историю информатики принято разбивать на два больших этапа: предистория и история.

Предистория информатики такая же древняя, как и история развития человеческого общества. В предистории выделяют (весьма приближенно) ряд этапов. Каждый из этих этапов характеризуется по сравнению с предыдущим резким возрастанием возможностей хранения, передачи и обработки информации.

Начальный этап предистории — освоение человеком развитой устной речи. Членораздельная речь, язык стал специфическим социальным средством хранения и передачи информации.

Второй этап — возникновение письменности. Прежде всего резко возросли (по сравнению с предыдущим этапом) возможности по хранению информации. Человек получил искусственную внешнюю память. Организация почтовых служб позволила использовать письменность и как средство для передачи информации. Кроме того, возникновение письменности было необходимым условием для начала развития наук (вспомним Древнюю Грецию, например). С этим же этапом, по всей видимости, связано и возникновение понятия *натуральное число*. Все наро-

ды, обладавшие письменностью, владели понятием числа и пользовались той или иной системой счисления.

Третий этап — книгопечатание. Книгопечатание можно смело назвать первой информационной технологией. Воспроизведение информации было поставлено на поток, на промышленную основу. По сравнению с предыдущим этот этап не столько увеличил возможности по хранению (хотя и здесь был выигрыш: письменный источник — часто единственный экземпляр, печатная книга — целый тираж экземпляров, а следовательно, и малая вероятность потери информации при хранении (вспомним «Слово о полку Игореве»), сколько повысил доступность информации и точность ее воспроизведения.

Четвертый и последний этап предистории связан с успехами точных наук (прежде всего математики и физики) и начинающейся в то время научно-технической революцией. Этот этап характеризуется возникновением таких мощных средств связи, как радио, телефон и телеграф, к которым по завершению этапа добавилось и телевидение. Кроме средств связи появились новые возможности по получению и хранению информации — фотография и кино. К ним также очень важно добавить разработку методов записи информации на магнитные носители (магнитные ленты, диски).

С разработкой первых ЭВМ принято связывать возникновение информатики как науки, начало ее истории. Для такой «привязки» имеется несколько причин. Во-первых, сам термин «информатика» появился на свет благодаря развитию вычислительной техники, и поначалу под ним понималась наука о вычислениях (первые ЭВМ большей частью использовались для проведения числовых расчетов). Во-вторых, выделению информатики в отдельную науку способствовало такое важное свойство современной вычислительной техники, как единая форма представления обрабатываемой и хранимой информации. Вся информация, вне зависимости от ее вида, хранится и обрабатывается на ЭВМ в двоичной форме. Так получилось, что компьютер в одной системе объединил хранение и обработку числовой, текстовой (символьной) и аудиовизуальной (звук, изображение) информации. В этом состояла иницирующая роль вычислительной техники при возникновении и оформлении новой науки.

На сегодняшний день информатика представляет собой комплексную научно-техническую дисциплину. Информатика под своим названием объединяет довольно обширный комплекс наук, каждая из которых занимается изучением одного из аспектов понятия *информация*. Предпринимаются интенсивные усилия ученых по сближению наук, составляющих информатику. Однако процесс сближения этих научных дисциплин идет довольно медленно и создание единой и всеохватывающей науки об информации представляется делом будущего.

Контрольные вопросы

1. Дайте определение понятию «информация».
2. Расскажите общую схему системы передачи информации.
3. В чем разница между АВМ и ЦВМ?
4. Как Вы понимаете смысл дисциплины «информатика»?
5. Что такое АИС?
6. Раскройте понятие «информационная технология».
7. Назовите этапы технологического процесса в информационных системах?
8. Дайте определение информационному ресурсу.
9. Перечислите известные Вам виды информации.
10. Как Вы понимаете свойства информации?
11. Что такое информационное общество?
12. Какие Вы знаете *промышленные и информационные* перевороты в обществе?
13. Раскройте сущность и цели информатизации?
14. Раскройте содержание технической базы информатизации.
15. В чем Вы видите значение вычислительных машин пятого поколения для информатизации общества.
16. Дайте характеристику факторам, влияющим на развитие индустрии информатики.

ГЛАВА 2

МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

Так как понятие «информация» занимает центральное место в информатике, то является естественным начать изучение математических основ информатики с методов и моделей количественной оценки информации. Трудно себе представить материал главы о математических основах информатики без раздела «Системы счисления». Во второй главе рассмотрены основные на сегодня системы счисления, используемые в вычислительной технике: позиционные и смешанные, перевод чисел из одной системы счисления в другую. Приведены доходчивые примеры. Глава заканчивается рассмотрением форм представления и преобразования информации. Раскрывается числовая система ЭВМ, представление чисел, кодирование и декодирование символьной информации в ЭВМ, форматы данных.

Методические указания

При изучении материала второй главы студенты должны понимать «кибернетическое» понятие информации как произвольного текста, т.е. последовательности символов некоторого алфавита, а также уметь определять количество информации в конкретных сообщениях (при заданном способе кодирования). Непременным условием проверки правильности усвоения данного материала является умение студента определять объем памяти ЭВМ, необходимый для хранения данной информации.

Важным моментом понимания студентом работы ЭВМ является усвоение понятия «системы счисления», форм представления и преобразования информации, т.е. каким образом в ЭВМ организованы числовая и символьная информация.

2.1. МЕТОДЫ И МОДЕЛИ ОЦЕНКИ КОЛИЧЕСТВА ИНФОРМАЦИИ

Для теоретической информатики информация играет такую же роль, как и вещество в физике. И подобно тому, как веществу можно приписывать довольно большое количество характеристик: массу, заряд, объем и т.д., так и для информации имеется пусть и не столь большой, но достаточно представительный набор характеристик. Как и для характеристик вещества, так и для характеристик информации имеются единицы измерения, что позволяет некоторой порции информации приписывать числа — *количественные характеристики информации*.

На сегодняшний день наиболее известны следующие способы измерения информации:

- объемный,
- энтропийный,
- алгоритмический.

Объемный является самым простым и грубым способом измерения информации. Соответствующую количественную оценку информации естественно назвать объемом информации.

Объем информации в сообщении — это количество символов в сообщении. Поскольку, например, одно и то же число может быть записано многими разными способами (с использованием разных алфавитов):

«двадцать один»

21

11001

XXI,

то этот способ чувствителен к форме представления (записи) сообщения. В вычислительной технике вся обрабатываемая и хранимая информация вне зависимости от ее природы (число, текст, отображение) представлена в двоичной форме (с использованием алфавита, состоящего всего из двух символов 0 и 1). Такая стандартизация позволила ввести две стандартные единицы измерения: бит и байт. *Байт* — это восемь бит. Более подробно эти единицы измерения будут рассмотрены в разделе «Формы представления и преобразования информации».

В теории информации и кодирования принят энтропийный подход к измерению информации. Этот способ измерения исходит из следующей модели. Получатель информации (сообщения) имеет определенные представления о возможных наступлениях некоторых событий. Эти представления в общем случае недостоверны и выражаются вероятностями, с которыми он ожидает то или иное событие. Общая мера неопределенности (энтропия) характеризуется некоторой математической зависимостью от совокупности этих вероятностей. Количество информации в сообщении определяется тем, насколько уменьшится эта мера после получения сообщения.

Поясним эту идею на простом примере. Пусть имеется колода карт, содержащая 32 различные карты. Возможность выбора одной карты из колоды — 32. Априори (доопытно, до произведения выбора) естественно предположить, что наши шансы выбрать некоторую определенную карту одинаковы для всех карт колоды. Произведя выбор, мы устраняем эту априорную неопределенность. Нашу априорную неопределенность можно было бы охарактеризовать количеством возможных равновероятных выборов. Если теперь определить количество информации как меру устраненной неопределенности, то и полученную в результате выбора информацию можно охарактеризовать числом 32.

Однако в теории информации получила использование другая количественная оценка, а именно — логарифм от описанной выше оценки по основанию 2:

$$H = \log_2 m, \quad (3.1)$$

где m — число возможных равновероятных выборов (при $m = 2$, $H = 1$).

То есть для выбора из колоды имеем следующую оценку количества информации, получаемую в результате выбора:

$$H = \log_2 32 = 5.$$

Полученная оценка имеет интересную интерпретацию. Она характеризует число двоичных вопросов, ответы на которые позволяют выбрать либо «да», либо «нет». Для выбора дамы пик такими вопросами будут:

- | | |
|-----------------------------|----------------|
| 1. Карта красной масти? | Ответ «нет» 0. |
| 2. Трефы? | Ответ «нет» 0. |
| 3. Одна из четырех старших? | Ответ «да» 1. |
| 4. Одна из двух старших? | Ответ «нет» 0. |
| 5. Дама? | Ответ «да» 1. |

Этот выбор можно описать последовательностью из пяти двоичных символов 00101 (0 — нет, 1 — да).

На первый взгляд может показаться, что эта интерпретация не годится в случае, когда количество выборов не равно степени двойки, так как получается нецелое количество вопросов, к примеру, если взять колоду из 36 карт (добавлены шестерки), то можно заметить, что для того, чтобы выяснить у участника «эксперимента», какую карту он выбрал, в ряде случаев понадобится 5 вопросов, как и в предыдущем случае, а в ряде случаев — и 6 вопросов. Усреднение по случаям и дает получаемую по формуле нецелую величину.

К. Шеннону принадлежит обобщение H на случай, когда H зависит не только от m , но и от вероятностей возможных выборов (для сообщения — вероятности выбора символов). Для количества собственной индивидуальной информации он предложил соотношение

$$h_i = \log(1/P_i) = -\log P_i, \quad (3.2)$$

где P_i — вероятность выбора i -го символа алфавита.

Более удобно пользоваться средним значением количества информации, приходящимся на один символ алфавита:

$$H = -\sum_{j=1}^m P_j \log P_j, \quad j = \overline{1, m}. \quad (3.3)$$

При равновероятных выборах все $P_i = 1/m$ и получается прежняя формула $H = \log m$.

В алгоритмической теории информации (раздел теории алгоритмов) предлагается алгоритмический метод оценки информации в сообщении. Этот метод кратко можно охарактеризовать следующими рассуждениями.

Каждый согласится, что слово 0101...01 сложнее слова 00...0, а слово, где 0 и 1 выбираются из эксперимента — бросания монеты (где 0 — герб, 1 — решка), сложнее обоих предыдущих.

Компьютерная программа, производящая слово из одних нулей, крайне проста: печатать один и тот же символ. Для получения 0101...01 нужна чуть более сложная программа, печатающая символ, противоположный только что напечатанному. Случайная, не обладающая ни какими закономерностями последовательность не может быть произведена никакой «короткой» программой. Длина программы, производящей хаотичную последовательность, должна быть близка к длине последней.

Приведенные рассуждения позволяют предположить, что любому сообщению можно приписать количественную характеристику, отражающую сложность (размер) программы, которая позволяет ее произвести.

Так как имеется много разных вычислительных машин и разных языков программирования (разных способов задания алгоритма), то для определенности задаются некоторой конкретной вычислительной машиной, сложность слова (сообщения) определяется как минимальное число внутренних состояний машины, требующиеся для его воспроизведения. Так же в алгоритмической теории информации используются и другие способы задания сложности.

2.2. СИСТЕМЫ СЧИСЛЕНИЯ

Системой счисления называется совокупность приемов наименования и записи чисел. В любой системе счисления для представления чисел выбираются некоторые символы (слова или знаки), называемые *базисными числами*, а все остальные числа получаются в результате каких-либо операций из базисных чисел данной системы исчисления. Символы, используемые для записи чисел, могут быть любыми, только они должны быть разными и значение каждого из них должно быть известно. В современ-

ном мире наиболее распространенным является представление чисел посредством арабских цифр 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 — специальных знаков, используемых для записи чисел. Системы счисления различаются выбором базисных чисел и правилами образования из них остальных чисел. Например, в римской системе счисления базисными являются числа 1, 5, 10, 50, 100, 500, 1000, которые обозначаются знаками I, V, X, L, C, D, M, а другие получаются путем сложения и вычитания базисных: если цифра справа меньше или равна цифре слева, то эти цифры складываются; если цифра слева меньше, чем цифра справа, то левая цифра вычитается из правой. Так, например, число 146 в римской системе счисления имеет вид CXLVI (C — 100, XL — 40, VI — 6), здесь «сорок» получается посредством вычитания из «пятидесяти» числа «десять», «шесть» — посредством сложения «пяти» и «единицы». Системы счисления, в которых любое число получается путем сложения или вычитания базисных чисел, называются *аддитивными*. При таком представлении чисел правила сложения для небольших чисел очевидны и просты, однако если возникает необходимость выполнять операции сложения над большими числами или операции умножения и деления, то римская система счисления оказывается неудобной. В этой ситуации преимущественнее оказываются позиционные системы счисления. Хотя в них, как правило, представление чисел далеко не так просто и очевидно, как в римской системе счисления, систематичность представления, основанная на «позиционном весе» цифр, обеспечивает простоту выполнения операций умножения и деления.

В римской системе счисления каждый числовой знак в записи любого числа имеет одно и то же значение, т.е. значение числового знака не зависит от его расположения в записи числа. Таким образом, римская система счисления не является позиционной системой счисления.

2.2.1. ПОЗИЦИОННЫЕ СИСТЕМЫ СЧИСЛЕНИЯ

Для изображения (или представления) чисел в настоящее время используются в основном позиционные системы счисления. Привычной для всех является десятичная система счисления. В этой системе для записи любых чисел используется только десять разных знаков (цифр): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Эти цифры введены для обозначения первых десяти последовательных чисел, а следующее число 10 и т.д. обозначается уже без использования новых цифр. Однако введением этого обозначения сделан важный шаг в построении системы счисления: значение каждой цифры поставлено в зависимость от того места, где она стоит в изображении числа.

Таким образом, система называется *позиционной*, если значение каждой цифры (ее вес) изменяется в зависимости от ее положения (позиции) в последовательности цифр, изображающих число.

Десятичная позиционная система счисления основана на том, что десять единиц каждого разряда объединяются в одну единицу соседнего старшего разряда. Таким образом, каждый разряд имеет вес, равный степени 10. Например, в записи числа 343.32 цифра 3 повторена три раза, при этом самая левая цифра 3 означает количество сотен (ее вес равен 10^2); цифра 3, стоящая перед точкой, означает количество единиц (ее вес равен 10^0), а самая правая цифра 3 — количество десятых долей единицы (ее вес равен 10^{-1}), так что последовательность цифр 343.32 представляет собой сокращенную запись выражения:

$$3 \times 10^2 + 4 \times 10^1 + 3 \times 10^0 + 3 \times 10^{-1} + 2 \times 10^{-2}.$$

Десятичная запись любого числа X в виде последовательности цифр:

$$a_n a_{n-1} \dots a_1 a_0 a_{-1} \dots a_{-m} \dots \quad (3.4)$$

основана на представлении этого числа в виде полинома:

$$X = a_n \times 10^n + a_{n-1} \times 10^{n-1} + \dots + a_1 \times 10^1 + a_0 \times 10^0 + a_{-1} \times 10^{-1} + \dots + a_{-m} \times 10^{-m} \dots, \quad (3.5)$$

где каждый коэффициент a_i может быть одним из чисел, для обозначения которых введены специальные знаки. Запись числа X в виде (3.4) представляет собой просто перечисление всех коэффициентов этого полинома. Точка, отделяющая целую часть числа от дробной, служит для фиксации конкретных значений каждой позиции в этой последовательности цифр и является началом отсчета.

Число K единиц какого-либо разряда, объединяемых в единицу более старшего разряда, называют *основанием позиционной системы счисления*, а сама система счисления называется *K-ичной*. Например, основанием десятичной системы счисления является число 10; двоичной — число 2; троичной — число 3 и т.д. Для записи произвольного числа в K -ичной системе счисления достаточно иметь K разных цифр a_i , $i = 1, \dots, K$. Например, в троичной системе счисления любое число может быть выражено посредством цифр 0, 1, 2. Эти цифры служат для обозначения некоторых различных целых чисел, называемых базисными.

Числа можно записать как суммы степеней не только числа 10, но и любого другого натурального числа, большего 1. Например, в Древнем Вавилоне использовалась система счисления с основанием 60. Деление часа на 60 минут, а минуты на 60 секунд заимствовано именно из этой системы счисления. А то, что человечество выбрало в качестве основания системы счисления число 10, вероятно, связано с тем, что природа наделила людей десятью пальцами.

Запись произвольного числа X в K -ичной позиционной системе счисления основывается на представлении этого числа в виде полинома:

$$X = a_n K^n + a_{n-1} K^{n-1} + \dots + a_1 K^1 + a_0 K^0 + a_{-1} K^{-1} + \dots + a_{-m} K^{-m} + \dots, \quad (3.6)$$

где каждый коэффициент a_i может быть одним из базисных чисел и изображается одной цифрой. Как и в десятичной системе счисления, число X , представленное в K -ичной системе счисления, можно кратко записать в виде (3.4) путем перечисления всех коэффициентов полинома (3.6) с указанием позиционной точки. Последовательность цифр, стоящая в (3.4), является изображением числа X в K -ичной системе счисления. Базисные числа должны быть выбраны так, чтобы любое число X могло быть представлено в виде полинома (3.6). Обычно в качестве базисных чисел берутся последовательные целые числа от 0 до $K-1$ включительно.

Арифметические действия над числами в любой позиционной системе счисления производятся по тем же правилам, что и в десятичной системе, так как все они основываются на правилах выполнения действий над соответствующими полиномами. При этом нужно только пользоваться теми таблицами сложения и умножения, которые имеют место при данном основании K системы счисления.

Для указания того, в какой системе счисления записано число, условимся при его изображении основание системы счисления указывать в виде нижнего индекса при нем, например, 35.64₈.

2.2.2. ДВОИЧНАЯ СИСТЕМА СЧИСЛЕНИЯ

В современной вычислительной технике, в устройствах автоматики и связи широко используется двоичная система счисления. Это система счисления с наименьшим возможным основанием. В ней для изображения числа используются только две цифры: 0 и 1.

Произвольное число X в двоичной системе представляется в виде полинома:

$$X = a_n \times 2^n + a_{n-1} \times 2^{n-1} + \dots + a_1 \times 2^1 + a_0 \times 2^0 + a_{-1} \times 2^{-1} + \dots + a_{-m} \times 2^{-m} + \dots, \quad (3.7)$$

где каждый коэффициент a_i может быть либо 0, либо 1.

Примеры изображения чисел в двоичной системе счисления:

1 = 1 ₂	5 = 101 ₂	9 = 1001 ₂
2 = 10 ₂	6 = 110 ₂	10 = 1010 ₂
3 = 11 ₂	7 = 111 ₂	0.5 = 0.1 ₂
4 = 100 ₂	8 = 1000 ₂	0.25 = 0.01 ₂

Таблица сложений чисел в двоичной системе имеет вид:

0 + 0 = 0	1 + 0 = 1
0 + 1 = 1	1 + 1 = 10

Таблица умножений в двоичной системе счисления имеет вид:

0 × 0 = 0	1 × 0 = 0
0 × 1 = 0	1 × 1 = 1

Так как в двоичной системе счисления для изображения любых чисел используются только две различные цифры, то при построении ЭВМ можно использовать элементы, которые могут находиться только в двух состояниях (например, высокое или низкое напряжение в цепи тока, наличие или отсутствие электрического импульса и т.п.). Это обстоятельство, а также простота выполнения арифметических операций являются причиной того, что большинство современных ЭВМ используют двоичную систему счисления.

2.2.3. ДРУГИЕ ПОЗИЦИОННЫЕ СИСТЕМЫ СЧИСЛЕНИЯ

Неудобство использования двоичной системы счисления заключается в громоздкости записи чисел. Это неудобство не имеет существенного значения на ЭВМ. Однако если возникает необходимость кодировать информацию «вручную», например, при составлении программы на машинном языке, то предпочтительнее оказывается пользоваться восьмеричной или шестнадцатеричной системой счисления (в силу их свойств, которые будут отмечены позднее).

В восьмеричной системе счисления базисными числами являются 0, 1, 2, 3, 4, 5, 6, 7. Запись любого числа в этой системе основывается на его разложении по степеням числа восемь с коэффициентами, являющимися указанными выше базисными числами.

Например, десятичное число 83.5 в восьмеричной системе будет изображаться в виде 123.4. Действительно, эта запись по определению означает представление числа в виде полинома:

$$1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0 + 4 \times 8^{-1} = 64 + 16 + 3 + 4/8 = 83.5.$$

В шестнадцатеричной системе счисления базисными являются числа от нуля до пятнадцати. Эта система отличается от рассмотренных ранее тем, что в ней общепринятых (арабских) цифр не хватает для обозначения всех базисных чисел, поэтому приходится вводить в употребление новые символы. Обычно для обозначения первых десяти целых чисел от нуля до девяти используют арабские цифры, а для следующих целых чисел от десяти до пятнадцати используются буквенные обозначения a, b, c, d, e, f.

Например, десятичное число 175.5 в шестнадцатеричной системе будет записываться в виде af.8. Действительно:

$$10 \times 16^1 + 15 \times 16^0 + 8 \times 16^{-1} = 160 + 15 + 8/16 = 175.5.$$

2.2.4. СМЕШАННЫЕ СИСТЕМЫ СЧИСЛЕНИЯ

В ряде случаев числа, заданные в системе счисления с основанием P , приходится изображать с помощью цифр другой системы счисления с основанием Q , где $Q < P$. Такая ситуация возникает, например, когда в ЭВМ, способной непосредственно воспринимать только двоичные числа, необходимо изобразить десятичные числа, с которыми привыкли работать. В этих случаях используются смешанные системы счисления, в которых каждый коэффициент P -ичного разложения числа записывается в Q -ичной системе. В такой системе P называется старшим основанием, а Q — младшим основанием, а сама смешанная система называется $(Q - P)$ -ичной. Для того чтобы запись числа в смешанной системе счисления была однозначной, для представления любой P -ичной цифры отводится одно и то же количество Q -ичных разрядов, достаточное для представления любого базисного числа P -ичной системы. Так, в смешанной двоично-десятичной системе счисления для изображения каждой десятичной цифры отводится четыре двоичных разряда. Например, десятичное число $x = 925$ в двоично-десятичной системе запишется в виде 1001 0010 0101. Здесь последовательные четверки (тетрады) двоичных разрядов изображают цифры 9, 2, 5 записи числа в десятичной системе счисления. Следует обратить внимание, что хотя в двоично-десятичной записи числа и используются только цифры 0 и 1, эта запись отличается от двоичного изображения данного числа. Например, приведенный выше двоичный код в двоичной системе счисления изображает число 2341, а не число 925.

Условимся изображать принадлежность числа к $(Q - P)$ -ичной системе счисления с помощью нижнего индекса $(Q - P)$ при данном числе, например: $925_{10} = 100100100101_{2-10}$.

Аналогично рассмотренной выше двоично-десятичной системе можно использовать и другие смешанные системы при различных значениях P и Q .

Свойство смешанных систем широко используется на практике для сокращенной записи чисел, заданных в системе счисления с небольшим основанием. Для этого в исходной записи числа разряды объединяются вправо и влево от точки в группы некоторой длины (добавляя в случае необходимости левее старшей или правее младшей значащих цифр соответствующее количество нулей), и каждая такая группа записывается одной цифрой другой системы, основание которой равно соответствующей степени исходного основания. Например, двоичное изображение 101110.1 числа 46.5 можно записать короче с использованием цифр других систем, причем эта сокращенная запись одновременно является и изображением данного числа в соответствующей системе счисления:

$$\begin{aligned}10\ 11\ 10.10 &= 232.2_4; \\101\ 110.100 &= 56.4_8; \\0010\ 1110.1000 &= 2e.8_{16} \text{ и т.д.}\end{aligned}$$

2.2.5. ПЕРЕВОД ЧИСЕЛ ИЗ ОДНОЙ СИСТЕМЫ СЧИСЛЕНИЯ В ДРУГУЮ

При решении задач с помощью ЭВМ исходные данные обычно задаются в десятичной системе счисления; в этой же системе, как правило, нужно получить и окончательные результаты. Так как в современных ЭВМ данные кодируются в основном в двоичных кодах, то, в частности, возникает необходимость перевода чисел из десятичной в двоичную систему счисления и наоборот.

При рассмотрении правил перевода чисел из одной системы счисления в другую ограничимся только такими системами счисления, у которых базисными числами являются последовательные целые числа от 0 до $P-1$ включительно, где P — основание системы счисления.

Задача перевода заключается в следующем. Пусть известна запись числа x в системе счисления с каким-либо основанием P :

$$p_n p_{n-1} \dots p_1 p_0 p_{-1} p_{-2} \dots,$$

где p_i — цифры P -ичной системы ($0 \leq p_i \leq P-1$). Требуется найти запись этого же числа x в системе счисления с другим основанием Q :

$$q_i q_{i-1} \dots q_1 q_0 q_{-1} q_{-2} \dots,$$

где q_i — искомые цифры Q -ичной системы ($0 \leq q_i \leq Q-1$). При этом можно ограничиться случаем положительных чисел, так как перевод любого числа сводится к переводу его модуля и приписыванию числу нужного знака.

2.3. ФОРМЫ ПРЕДСТАВЛЕНИЯ И ПРЕОБРАЗОВАНИЯ ИНФОРМАЦИИ

При любых видах работы с информацией всегда идет речь о ее представлении в виде определенных символических структур. Наиболее распространены одномерные представления информации, при которых сообщения имеют вид последовательности символов. Так информация представляется в письменных текстах, при передаче по каналам связи, в памяти ЭВМ. Однако широко используется и многомерное представление информации, причем под многомерностью понимается не только расположение элементов информации на плоскости или в пространстве в виде рисунков,

схем, графов, объемных макетов и т.п., но и множественность признаков используемых символов, например цвет, размер, вид шрифта в тексте.

Формирование представления информации называется ее *кодированием*. В более узком смысле под кодированием понимается переход от исходного представления информации, удобного для восприятия человеком, к представлению, удобному для хранения, передачи и обработки. В этом случае обратный переход к исходному представлению называется *декодированием*.

При кодировании информации ставятся следующие цели:

- 1) удобство физической реализации;
- 2) удобство восприятия;
- 3) высокая скорость передачи и обработки;
- 4) экономичность, т.е. уменьшение избыточности сообщения;
- 5) надежность, т.е. защита от случайных искажений;
- 6) сохранность, т.е. защита от нежелательного доступа к информации.

Эти цели часто противоречат друг другу. Стремясь к экономным сообщениям, мы тем самым уменьшаем их надежность и удобство восприятия. Экономные сообщения могут повысить скорость обработки информации (более короткое сообщение будет передано или прочтено быстрее), но могут и уменьшить ее. А защита информации от нежелательного доступа уменьшает объем хранимой информации и замедляет работу с ней.

На разных этапах обработки информации достигаются разные цели и поэтому информация неоднократно перекодируется, преобразуется из вида, удобного для восприятия человеком, к виду, удобному для обработки автоматическими средствами, и наоборот. Такое представление происходит, например, при передаче телеграмм, при программировании на ЭВМ.

Рассмотрим способы представления информации в ЭВМ. Для записи, хранения и выдачи по запросу информации, обрабатываемой с помощью ЭВМ, предназначено запоминающее устройство (или память) ЭВМ.

В отличие от обычной словесной формы, принятой в письменной речи, информация в памяти ЭВМ записывается в форме *цифрового двоичного кода*. Это объясняется тем, что электронные элементы, из которых строится оперативная память, могут находиться только в одном из двух устойчивых состояний, которые можно интерпретировать как 0 или 1.

Количество информации, которое может помещаться в один элемент памяти (0 или 1), называемое *битом*, очень мало и не несет никакой смысловой нагрузки. Однако если соединить несколько таких элементов в ячейку, то тогда можно сохранить в запоминающем устройстве столько информации, сколько потребуется. Последовательность битов, рассмат-

риваемых аппаратной частью ЭВМ как единое целое, называется *машинным словом*.

Так как оперативная память ЭВМ состоит из конечной последовательности слов, а слова — из конечной последовательности битов, то объем представляемой в ЭВМ информации ограничен *емкостью памяти*, а числовая информация может быть представлена только с определенной точностью, зависящей от архитектуры памяти данной ЭВМ.

2.3.1. ЧИСЛОВАЯ СИСТЕМА ЭВМ. ПРЕДСТАВЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ БЕЗ ЗНАКА И СО ЗНАКОМ

Введем основные понятия на примере 4-битовых машинных слов. Такой размер слова обеспечивает хранение десятичных чисел только от 0 до 15 и поэтому не представляет практического значения. Однако они менее громоздки, а основные закономерности, обнаруженные на примере 4-битовых слов, сохраняют силу для машинного слова любого размера.

Предположим, что процессор ЭВМ способен увеличивать (прибавлять 1) и дополнять (инвертировать) 4-битовые слова. Например, результатом увеличения слова 1100 является 1101, а результатом дополнения этого слова является 0011. Рассмотрим слово 0000, представляющее десятичное число 0. В результате увеличения содержимое этого слова станет равным 0001, что соответствует десятичному числу 1. Продолжая последовательно увеличивать 4-битовые слова, придем к ситуации, когда, увеличивая слово 1111 (которое представляет десятичное число 15), получим в результате слово 0000, т.е. $111+1 = 0000$ ($15+1 = 0$), при этом получили неверную арифметическую операцию и вернулись в исходное состояние. Это произошло из-за того, что слово памяти может состоять только из конечного числа битов. Таким образом, числовая система ЭВМ является *конечной и циклической*.

Такой ситуации, приводящей к неверному арифметическому результату, можно избежать, если битовую конфигурацию 1111 принять за код для -1. Тогда 1110 интерпретируется как -2; 1101-3 и т.д. до 1000-8. Тем самым получили другую числовую систему — со знаком, содержащую как положительные, так и отрицательные числа. В этой системе половина четырехбитовых конфигураций, начинающаяся с единицы, интерпретируется как отрицательные числа, а другая половина, начинающаяся с 0, — как положительные числа или нуль. Поэтому старший бит числа (третий по счету, если нумерацию битов начинать с нуля справа налево) называется *знаковым битом*. Числовая система со знаком также конечна и циклическа, однако в этом случае арифметически неверный результат даст попытка увеличить число 8 на единицу. Преимущество введения числовой системы со знаком заключается в возможности представления как положительных, так и отрицательных чисел.

Если знаковый бит равен нулю, то значение числа легко вычисляется — игнорируется знаковый бит, а оставшиеся три бита интерпретируются как двоичный код десятичного числа. Например, слово 0110 представляет двоичное число 110, которое равно десятичному числу 6.

Для оценки отрицательного числа нужно изменить его знак. Рассмотрим четырехбитовое число k в системе со знаком. Тогда $-k = (-1 - k) + 1$, следовательно, для вычисления значения $-k$ необходимо вычесть k из -1 (т.е. из 1111) и затем прибавить 1 (т.е. 0001). Заметим, что операция вычитания всегда возможна, никогда не требует заема и равнозначна операции инвертирования битов вычитаемого. Например, $1111 - 1011 = 0100$, здесь в вычитаемом, равном 1011, единицы перешли в нули, а нуль — в единицу. Инвертирование битов в слове называется *дополнением до единицы*. Для определения отрицательного значения числа k надо к его дополнению до единицы прибавить единицу (согласно вышеприведенному равенству). Инвертирование битов в слове с добавлением единицы к младшему биту называется *дополнением до двух*. Например, требуется найти, какое число закодировано в слове 1001. Для этого сначала выполняем операцию инвертирования $1001 \rightarrow 0110$, а затем к полученному результату прибавляем единицу $0110 + 1 = 0111$, что является двоичным кодом числа 7. Таким образом, значением 1001 является отрицательное число 7, т.е. -7 .

2.3.2. ИНДИКАТОРЫ ПЕРЕНОСА И ПЕРЕПОЛНЕНИЯ

Рассмотрим более подробно ситуацию, приводящую при увеличении четырехбитового числа (т.е. прибавления к нему 1) к неверному арифметическому результату, возникшую из-за конечности числовой системы ЭВМ. В числовой системе без знака эта проблема возникает при увеличении слова 1111, при этом имеет место перенос единицы из знакового бита. В случае системы чисел со знаком перенос из старшего бита дает верный результат: $1111 + 0001 = 0000$ (что правильно: $-1 + 1 = 0$). Но в этой системе увеличение слова 0111 приводит к ошибочной ситуации: $0111 + 1 = 1000$ ($7 + 1 = -8$), при этом имеет место перенос в знаковый бит.

В процессоре ЭВМ (устройстве, в котором выполняются арифметические операции) содержатся два индикатора — индикатор переноса и индикатор переполнения. Каждый индикатор содержит 1 бит информации и может быть процессором установлен (в этом случае ему придается значение, равное 1) или сброшен (равен 0). *Индикатор переноса* указывает на перенос из знакового бита, а *индикатор переполнения* — на перенос в знаковый бит. Таким образом, после завершения операции, в которой происходит перенос в старший бит, процессор устанавливает индикатор переполнения, если такого переноса нет, то индикатор переполнения сбрасывается. Индикатор переноса обрабатывается аналогичным образом.

Важно то, что после выполнения операции процессором, ЭВМ сигнализирует о состоянии индикаторов и их можно проверить. Если состояние индикаторов указывает на неправильный арифметический результат, то необходимо предпринять меры, исправляющие эту ситуацию, т.е. пользователю ЭВМ представляется возможность контролировать правильность выполнения арифметических операций.

Пример 1. Рассмотрим сложение двух чисел $0101 + 0011 = 1000$. В результате выполнения операции сложения произошел перенос в знаковый бит, а переноса из знакового бита не было. Таким образом, после завершения этой операции индикатор переноса будет сброшен, а индикатор переполнения установлен. Поэтому если рассматривать эти два числа как целые без знака, то результат является арифметически правильным, так как индикатор переноса сброшен. Если рассматривать числа в системе со знаком, то бит переполнения показывает, что произошло изменение знака (перенос в знаковый бит есть, а из — нет), поэтому арифметический результат неправильный.

Пример 2. В результате сложения чисел $1101 + 0101 = 0010$ происходит перенос и в знаковый бит, и из знакового бита. Поэтому будет установлен индикатор переноса, а индикатор переполнения сброшен. Следовательно, в системе чисел без знака результат является арифметически неправильным, а в системе чисел со знаком — правильным.

Пример 3. В результате выполнения операции вычитания $1001 - 0011 = 1001 + (-0011) = 1001 + (1100 + 1) = 1001 + 1101 = 0110$ происходит перенос из знакового бита, а переноса в знаковый бит нет. Следовательно, индикатор переноса будет сброшен, а индикатор переполнения установлен, что указывает на то, что в данном примере в системе без знака результат арифметически правильный, а в системе со знаком — неправильный.

2.3.3. ПРЕДСТАВЛЕНИЕ СИМВОЛЬНОЙ ИНФОРМАЦИИ В ЭВМ

В отличие от обычной словесной формы, принятой в письменном виде, символьная информация хранится и обрабатывается в памяти ЭВМ в форме цифрового кода. Например, можно обозначить каждую букву числами, соответствующими ее порядковому номеру в алфавите: А — 01, Б — 02, В — 03, Г — 04, ... , Э — 30, Ю — 31, Я — 32. Точно так же можно договориться обозначать точку числом 33, запятую — 34 и т.д. Так как в устройствах автоматической обработки информации используются двоичные коды, то обозначения букв надо перевести в двоичную систему. Тогда буквы будут обозначаться следующим образом: А — 000001, Б — 000010, В — 000011, Г — 000100, ... , Э — 011110, Ю — 011111, Я — 100000. При таком кодировании любое слово можно представить в виде последовательности кодовых групп, составленных из 0 и 1. Например, слово ЭВМ выглядит так: 011110 000011 001110.

При преобразовании символов (знаков) в цифровой код между множествами символов и кодов должно иметь место взаимно-однозначное соответствие, т.е. разным символам должны быть назначены разные цифровые коды, и наоборот. Это условие является единственным необходимым требованием при построении схемы преобразования символов в числа. Однако существует ряд практических соглашений, принимаемых при построении схемы преобразования исходя из соображений наглядности, эффективности, стандартизации. Например, какое бы число ни назначили коду для знака 0 (не следует путать с числом 0), знаку 1 удобно назначить число, на единицу большее, чем код 0, и т.д. до знака 9. Аналогичная ситуация возникает и при кодировке букв алфавита: код для Б на единицу больше кода для А, а код для В на единицу больше кода для Б и т.д. Таким образом, из соображений наглядности и легкости запоминания целесообразно множества символов, упорядоченных по какому-либо признаку (например, лексико-графическому), кодировать также с помощью упорядоченной последовательности чисел.

Другим важным моментом при организации кодировки символьной информации является эффективное использование оперативной памяти ЭВМ. Так как общеупотребительными являются примерно 100 знаков (сюда помимо цифр, букв русского и английского алфавитов, знаков препинания, арифметических знаков входят знаки перевода строки, возврата каретки, возврата на шаг и т.п.), то для взаимно-однозначного преобразования всех знаков в коды достаточно примерно сотни чисел. Значение этого выбора заключается в том, что для размещения числа из этого диапазона в оперативной памяти достаточно одного байта, а не машинного слова. Следовательно, при такой организации кодировки достигается существенная экономия объема памяти.

При назначении кодов знакам надо также учитывать соглашения, касающиеся стандартизации кодировки. Можно назначить знаковые коды по своему выбору, но также возникнут трудности, связанные с необходимостью обмена информацией с другими организациями, использующими кодировку, отличную от нашей. В настоящее время существует несколько широко распространенных схем кодирования. Например, код BCD (Binary-Coded Decimal) — двоично-десятичный код используется для представления чисел, при котором каждая десятичная цифра записывается своим четырехбитовым двоичным эквивалентом. Этот код может оказаться полезным, когда нужно преобразовать строку числовых знаков, например, строку из числовых знаков «2537» в число 2537, над которым затем будут производиться арифметические действия. Расширением этого кода является EBCDIC (Extended Binary-Coded Decimal Interchange Code) — расширенный двоично-десятичный код обмена информацией, который преобразует как числовые, так и буквенные строки.

В ЭВМ типа PDP (или CM) применяется код ASCII (American Standard Code for Information Interchange) — американский стандар-

тный код обмена информацией. Этот код генерируется некоторыми внешними устройствами (принтером, алфавитно-цифровым печатающим устройством (АЦПУ)) и используется для обмена данными между ними и оперативной памятью ЭВМ. Например, когда нажимаем на терминале клавишу G, то в результате этого действия код ASCII для символа G (1000111) передается в ЭВМ. А если надо этот символ распечатать на АЦПУ, то его код ASCII должен быть послан на печатающее устройство.

Отечественной версией кода ASCII является код КОИ-7 (двоичный семибитовый код обмена информацией), который совпадает с ним, за исключением букв русского алфавита.

2.3.4. ФОРМАТЫ ДАННЫХ

Под *данными* будем понимать информацию, представленную в виде, пригодном для обработки автоматическими средствами, например, в двоичном коде. Формат представления данных в памяти ЭВМ зависит от ее архитектуры. Поэтому в дальнейшем для определенности, если не оговорено особо, будем иметь в виду архитектуру ЭВМ типа PDP фирмы DEC (отечественный аналог — СМ ЭВМ).

Данные, обрабатываемые ЭВМ, делятся на три группы: логические коды, числа с фиксированной запятой и числа с плавающей запятой.

Представление логических кодов. Логические коды могут размещаться в отдельных байтах и в словах. Для их представления используются все разряды: для байта от 0-го до 7-го, для слова — от 0-го до 15-го. Логическими кодами могут быть представлены символьные величины, числа без знака и битовые величины.

Символьные величины задаются в коде ASCII (КОИ-7), каждый символ занимает один байт, разряд 7 которого всегда содержит 0. Символы строки размещаются в последовательно-адресуемых байтах оперативной памяти. Например, символьная строка ABCDE (коды ASCII: А — 101₈, В — 102₈, С — 103₈, D — 104₈, E — 105₈), первый знак которой помещается в ячейку с адресом 1000 (адреса представлены в 8-ричной системе счисления), размещается в оперативной памяти следующим образом: числа без знака имеют диапазон представления от 000 до 377₈ — для байта, от 000000 до 177777₈ — для слова; битовые величины задают значения отдельных разрядов байта или слова.

Представление чисел в формате с фиксированной запятой. Числа с фиксированной запятой могут занимать байт или слово. Если число с фиксированной запятой занимает байт, то для его представления используются разряды с 0-го по 6-й. Разряд 7 называется знаковым. При размещении числа с фиксированной запятой в слове для его представления используются разряды с 0-го по 14-й. Знак числа содержится в раз-

ряде 15. Значения знакового разряда: 0 – для положительных чисел; 1 – для отрицательных чисел.

Отрицательные числа в формате с фиксированной запятой представляются в дополнительном коде (посредством операции дополнения до 2-х).

Примеры представления чисел с фиксированной запятой

Число	Байт		Слово	
	Восьмеричный код	Двоичный код	Восьмеричный код	Двоичный код
+5	005	00000101	000005	0000000000000101
+63	077	00111111	000077	0000000000111111
-5	373	11111011	177773	111111111111011
-63	315	11001101	177715	111111111001101
0	000	00000000	000000	0000000000000000

Диапазон представления чисел с фиксированной запятой: для байта — от -128_{10} до $+127_{10}$; для слова — от -32768_{10} до $+32767_{10}$. При выполнении операций над числами, представленными в формате с фиксированной запятой, они масштабируются таким образом, чтобы каждое число лежало в интервале $(-1, +1)$. Другими словами, в этом случае ЭВМ оперирует только с числами, по модулю не превосходящими единицы. При этом необходимо следить за тем, чтобы в результате операций результат не получился большим, чем $2^k - 1$, где k — число разрядов, отведенных для представления в машине. Такая опасность есть при выполнении операций сложения и деления. Опасность представляют также операции вычитания и умножения. При вычитании может получиться так, что разность станет числом меньшим, чем представляется в машине, и результат исчезнет. При многократном умножении (из-за того, что умножаются числа, меньшие единицы) может произойти то же самое. Поэтому при использовании формата представления чисел с фиксированной запятой приходится следить как за случаями возможного переполнения разрядной сетки машины, так и за случаями, связанными с появлением машинного нуля. Необходимость постоянно следить за тем, чтобы числа в машине не вышли за пределы интервала $(-1, +1)$, а также неизбежное в таких устройствах накопление абсолютной погрешности вычислений из-за перемасштабирования, при котором цифры младших разрядов (а именно в них накапливается абсолютная погрешность) передвигаются в старшие разряды, привели к тому, что в универсальных ЭВМ представление с фиксированной запятой практически перестало применяться. Оно сохраняется в специализированных вычислительных системах, где диапазон изменения чисел заранее проанализирован, в некоторых микропроцессорах и микроЭВМ.

Представление чисел в формате с плавающей запятой. Любое вещественное число x , представленное в системе счисления с основанием N , можно записать в виде

$$x = \pm mN^{\pm p},$$

где m — мантисса, p — характеристика (или порядок) числа.

Если $|m| < 1$, то запись числа называется *нормализованной слева*. Следующие примеры показывают, как можно представить любое число в форме с плавающей запятой:

а) в десятичной системе счисления

$$372,95 = 0,37295 \times 10^3;$$

$$25 = 0,025 \times 10^3 = 0,25 \times 10^2;$$

$$0,0000015 = 0,15 \times 10^{-5} = 0,015 \times 10^{-4};$$

б) в двоичной системе счисления

$$11010,1101 = 0,0110101101 \times 2^6 = 0,110101101 \times 2^5;$$

$$0,011011 = 0,11011 \times 2^{-1};$$

$$0,1 = 0,1 \times 2^0$$

(здесь порядок определяет, на сколько разрядов необходимо осуществить сдвиг относительно запятой).

Из этих примеров видно, что представление чисел в форме с плавающей запятой неоднозначно. В ЭВМ с целью минимизации погрешности при вычислениях и эффективного использования памяти применяют процедуру нормализации справа.

Число называют *нормализованным справа*, если после запятой в мантиссе стоит не нуль. Например, числа $0,00076_{10}$ и $0,00011_2$, представленные соответственно в виде $0,076 \times 10^{-2}$ и $0,011 \times 2^{-2}$, не являются нормализованными справа, а в виде $0,76 \times 10^{-3}$ и $0,11 \times 2^{-3}$ являются такими.

В дальнейшем под нормализацией числа будем понимать нормализацию справа.

Нормализованное число одинарной точности, представленное в формате с плавающей запятой, записывается в память следующим образом: знак числа — в бите 15 первого слова (0 — для положительных и 1 — для отрицательных чисел); порядок размещается в битах 7 — 14 первого слова, а мантисса занимает остальные 23 бита в двух словах. Нормализованное число двойной точности записывается в четыре слова памяти и отличается от представления чисел с одинарной точностью только тем, что продолжение мантиссы размещается в следующих за первым трех последовательных словах памяти, а всего под мантиссу в этом случае отводится 55 бит.

Порядок числа с плавающей запятой изменяется в диапазоне от -128_{10} (200_8) до $+127_{10}$ (177_8) и запоминается увеличенным на 128_{10} (200_8). Такой способ представления порядка называется *смещенным*.

Следует иметь в виду, что хотя для мантиссы отведены 23 разряда — для чисел одинарной точности и 55 разрядов — для чисел двойной точности, в операциях участвуют 24 и 56 разрядов соответственно, так как стар-

ший разряд мантиссы нормализованного числа не хранится, т.е. имеет место так называемый скрытый разряд. Однако при аппаратном выполнении операций этот разряд автоматически восстанавливается и учитывается при выполнении операций. Порядок числа также учитывает скрытый старший разряд мантиссы.

Также заметим, что нормализованная мантисса в двоичной системе счисления всегда представляется десятичным числом m , лежащим в диапазоне $0,5 \leq m < 1$.

Примеры представления чисел с плавающей запятой.

$$1) 0,1_{10} = 0,0(6314)_8 = 0,000(1100)_2 = 0,(1100)_2 \times 2^{-3}; \\ -3_{10} = (-3 + 200)_8 = 175_8 = 0111110_2.$$

Заметим, что в этом примере мантисса представлена бесконечной периодической дробью, поэтому последний учитываемый разряд мантиссы округляется.

$$2) -49,5_{10} = -61,4_8 = -110001,100_2 = -0,1100011_2 \times 2^6; \\ 6_{10} = (6 + 200)_8 = 206_8 = 10000110_2.$$

При выполнении арифметических операций над числами, представленными в формате с плавающей запятой, надо отдельно выполнять их для порядков и мантисс. При алгебраическом сложении чисел надо сначала уравнивать порядки слагаемых. При умножении порядки надо складывать, а мантиссы перемножать. При делении из порядка делимого вычитают порядок делителя, а над мантиссами совершают обычную операцию деления. После выполнения операций необходимо провести нормализацию результата, если это необходимо, что приводит к изменению порядков, так как каждый сдвиг на один разряд влево соответствует уменьшению порядка на единицу, а сдвиг вправо — увеличению его на единицу. Введение термина «плавающая запятая» как раз и объясняется тем, что двоичный порядок, определяющий фактическое положение запятой в изображении числа, корректируется после выполнения каждой арифметической операции, т.е. запятая в изображении числа плавает (изменяется ее положение) по мере изменения данной величины. А в изображении чисел с фиксированной запятой — запятая жестко зафиксирована в определенном месте.

Арифметические операции с числами в форме плавающей запятой намного сложнее таких же операций для чисел с фиксированной запятой. Но зато плавающая запятая позволяет производить операции масштабирования автоматически в самой машине и избавляет от накопления абсолютной погрешности при вычислениях (хотя не избавляет от накопления относительной погрешности).

Контрольные вопросы

1. Назовите основные способы измерения информации.
2. В чем состоит суть энтропийного подхода к измерению информации?
3. Что такое позиционная система счисления?
4. Что называется основанием системы счисления?
5. Какие числа называются базисными?
6. Какой смысл вкладывается в понятие записи и представления числа в K -ичной системе счисления?
7. Какая система счисления называется смешанной?
8. Что такое элемент и ячейка памяти?
9. Что такое машинное слово?
10. Что такое бит и байт?
11. Сформулируйте основные свойства числовой системы ЭВМ.
12. Почему при выполнении арифметических операций на ЭВМ возникают ситуации, приводящие к неверному результату?
13. Что такое дополнение до единицы и дополнение до двух?
14. Зачем нужны индикаторы переноса и переполнения?
15. Сформулируйте правила, определяющие правильность выполнения операции сложения в разных числовых системах ЭВМ (со знаком и без знака).
16. Как представляются целые и вещественные числа в памяти ЭВМ?
17. Как представляются символьные данные в памяти ЭВМ?

ГЛАВА 3

АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

Наверняка можно утверждать, что каждый, читающий эти строки, знаком с термином «алгоритм». Его применяют весьма широко и не только в области вычислительной техники и программирования. Так же несомненно и то, что у читателя сформировалось свое (пусть даже большей частью интуитивное) понимание смысла этого термина.

Термин происходит от имени средневекового математика Абу Джафара ибн Мусы аль-Хорезми. Редакция последней части имени ученого в европейских языках привела к образованию термина «алгорифм» или «алгоритм». Европейцы, начавшие осваивать современную десятичную систему счисления в XII в., познакомились с трудами арабских ученых, и труд упомянутого выше жителя Хорезма, посвященный правилам счета в десятичной системе счисления, был широко известен. Поэтому и наполнение термина «алгоритм» было следующим: операции над числами.

Через века старое, прежнее понимание этого термина стало утрачиваться, и данный термин стали применять по отношению к одному-единственному алгоритму — алгоритму Евклида.

Методические указания

Целью изучения материала третьей главы является выработка у студентов основ алгоритмического мышления. Здесь же изучается реальный язык программирования — Бейсик или Паскаль и жизненный цикл программного обеспечения.

Студенты должны з н а т ь: что алгоритм — это организованная последовательность действий, допустимых в определенных случаях, и что программа —

это алгоритм, записанный на языке, понятном программисту и ЭВМ. При этом студенты должны уметь распознавать, подходит ли данный алгоритм для решения задач данного класса. Важным моментом усвоения основ алгоритмизации является четкое понимание студентами ветвления, циклов и рекурсии; что ветвление в алгоритмах появляется тогда, когда необходимо сделать выбор одного из нескольких действий в зависимости от некоторого условия, что любой выбор можно свести к одному или нескольким ветвлениям и что при записи ветвлениям и что при записи ветвлений необходим указатель конца ветвления, отделяющий ветвление от остальной части алгоритма.

Студент должен у м е т ь: составлять «протоколь» выполнения разветвляющихся и циклических алгоритмов, мысленно совершая действия алгоритма и комментируя их; записывать разветвляющиеся и циклические алгоритмы, не допуская двусмысленности записи (из записи алгоритма должно быть понятно, где начинается и кончается ветвление и цикл); изменять ветвление и циклы при решении задач (при переходе от модели к алгоритму); использовать простейшие приемы отладки разветвляющихся и циклических программ, проводить вычислительные эксперименты по программам, написанных на языке программирования.

В результате изучения одного из языков программирования (Бейсика или Паскаля) студент должен з н а т ь: что язык программирования — это одно из средств общения с ЭВМ; правила перевода алгоритмов на язык программирования; перевод программ с одного языка на другой; понятия слова, символьной переменной; основные действия со словами, понятия кодирования и декодирования текстов. Студенты должны п о н и м а т ь: что изучать язык программирования означает: узнать, как в нем называются те или иные допустимые действия и что при решении задач на ЭВМ можно пользоваться разными методами и что одни методы могут быть эффективнее других (например, метод деления пополам обычно эффективнее метода простого прибора); что алфавит — это произвольный набор символов; что слово — это любая последовательность символов некоторого алфавита; что существуют различные способы кодирования текстов.

3.1. ОПРЕДЕЛЕНИЕ АЛГОРИТМА

Алгоритм Евклида был предназначен для нахождения наибольшего общего делителя пары натуральных чисел (m, n).

Алгоритм Евклида

1. {Нахождение остатка} $r := m \bmod n$.
2. {Замена} $m := n; n := r$.
3. {Остановка?} Если $n \neq 0$, то переход к п.1.
4. {Остановка процесса} m — искомое число.

Представленное описание алгоритма — это последовательность шагов, направленных на достижение некоторого результата (наибольшего общего делителя).

Современное содержание понятия алгоритма можно определить следующим образом. В этой книге алгоритмы обсуждаются в двух раз-

личных аспектах. Первый аспект — теоретический, а второй — прагматический, тесно связанный с программированием.

Теория алгоритмов — раздел математики, изучающий общие свойства алгоритмов. Понятие «алгоритм» сформировалось в математике в 20-х годах XX в. Началом систематической разработки теории алгоритмов можно считать 1936 г. и связывают это начало с публикацией работы А.А. Черча.

Под алгоритмом всегда (и до возникновения строгой теории) понималась процедура, которая позволяла путем выполнения последовательности элементарных шагов получать однозначный результат (независящий от того, кто именно выполнял эти шаги) или за конечное число шагов прийти к выводу о том, что решения не существует.

Конечно же, это нестрогое определение понятия алгоритма и именно попытки сформулировать такое понятие привели к возникновению теории алгоритмов. Причиной развития этой теории были внутренние проблемы математики и лишь с возникновением и развитием вычислительной техники и смежных наук выяснилось, что в основе этих наук должна лежать теория алгоритмов. Так стало очевидным прикладное значение новой науки.

В основе формализации понятия «алгоритм» лежит идея построения алгоритмической модели. Составляющими такой модели должны быть конкретный набор элементарных шагов, способы определения следующего шага и т.д. От модели также требуется простота и универсальность. Требование простоты важно для того, чтобы выделить действительно необходимые элементы и свойства алгоритма и облегчить доказательства общих утверждений об этих свойствах. Универсальность необходима для того, чтобы модель позволяла описать любой алгоритм.

Результатами теоретических исследований явились три основных класса арифметических моделей.

Первый класс моделей основан на арифметизации алгоритмов. Предполагается, что любые данные можно закодировать числами, и как следствие — всякое их преобразование становится в этом случае арифметическим вычислением, алгоритмом в таких моделях есть вычисление значения некоторой числовой функции, а его элементарные шаги — арифметические операции. Последовательность шагов определяется двумя способами. Первый способ — суперпозиция, т.е. подстановка функции в функцию, а второй — рекурсия, т.е. определение значения функции через «ранее» вычисленные значения этой же функции. Функции, которые можно построить из целых чисел и арифметических операций с помощью суперпозиций и рекурсивных определений, называются *рекурсивными функциями*. Простейшим примером рекурсивной функции является факториал:

$$0! = 1, (n + 1)! = n! (n + 1).$$

Другие примеры рекурсивной функции можно найти в § 3.2.

Второй класс моделей порожден следующей идеей. Для того чтобы алгоритм понимался однозначно, а его каждый шаг считался элементарным и выполнимым, он должен быть представлен так, чтобы его могла выполнять машина, к которой предъявляются уже упомянутые требования простоты и универсальности. Одной из таких машин явилась абстрактная машина Тьюринга. Машина Тьюринга состоит из трех частей (рис. 3.1): ленты, головки и управляющего устройства (УУ). Лента бесконечна в обе стороны и разбита на ячейки. В каждой ячейке может быть записан только один символ.

Отсутствие символа в ячейке обозначается специальным «пустым» символом « ». Головка всегда располагается над некоторой ячейкой ленты. Она может читать и писать символы, стирать их и перемещаться вдоль ленты. Число возможных символов конечно, и образует алфавит машины $A = \{a_1, \dots, a_m\}$. Головка в каждый такт работы машины находится в одном из состояний. Множество таких состояний конечно $Q = \{q_1, \dots, q_n\}$ и среди них выделяют начальное q_1 и конечное q_n состояния.

Элементарный шаг машины Тьюринга состоит из следующих действий: головка считывает символ, записанный в ячейке, над которой она находится;

считанный символ a_k и текущее состояние головки q однозначно определяют новое состояние q' , новый записываемый символ a_l и перемещение головки d_p (которое может иметь значение на ячейку влево, на ячейку вправо, остаться на месте).

Устройство управления хранит и выполняет команды машины вида $qa_k \rightarrow q'a_l d_p$.

Конкретную машину Тьюринга (и алгоритм соответственно) можно задать, перечислив элементы A и Q и команды машины.

Третий класс моделей алгоритмов очень близок к предыдущему, но не оперирует конкретными машинными механизмами. Наиболее известная алгоритмическая модель этого типа — нормальные алгоритмы Маркова.

Для нормального алгоритма задается алфавит, над которым он работает, конечное множество допустимых подстановок и порядок их применения. Если в качестве алфавита взять алфавит русского языка, а в качестве множества подстановок

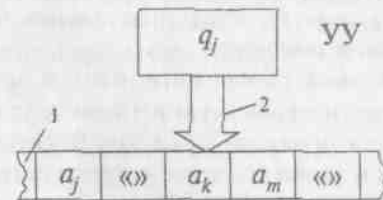


Рис. 3.1. Основные составные части машины Тьюринга: УУ — управляющее устройство; 1 — лента; 2 — головка; $A = \{a_1, \dots, a_m\}$ — алфавит машины; $Q = \{q_1, \dots, q_n\}$ — множество состояний машины

- 1) Я \rightarrow У 3) С \rightarrow М 5) Р \rightarrow Т 7) О \rightarrow х
 2) Л \rightarrow У 4) В \rightarrow б 6) Т \rightarrow Р 8) Н \rightarrow а,

то, используя следующие правила 1 — 3:

1) проверить возможность подстановок в порядке возрастания их номеров, и если она возможна (левая часть подстановки обнаружена в исходном слове), произвести подстановку (заменяв левую часть на правую);

2) если в примененной подстановке имеется символ «!», то преобразования прекращаются, а если нет, то текущее состояние становится исходным и весь процесс начинается заново;

3) если ни одна подстановка не применима, то процесс преобразования завершен.

Можно обнаружить, что по заданному алгоритму исходное слово «слон» превращается в слово «муха» по следующей цепочке: «слон» \rightarrow «суон» \rightarrow «муон» \rightarrow «мухи» \rightarrow «муха».

Алгоритм — точное предписание, которое задает алгоритмический процесс, начинающийся с произвольного исходного данного (из некоторой совокупности возможных для данного алгоритма исходных данных) и направленный на получение полностью определенного этим исходным данным результата.

Алгоритмический процесс — процесс последовательного преобразования конструктивных объектов (слов, чисел, пар слов, пар чисел, предложений и т.п.), происходящий дискретными «шагами». Каждый шаг состоит в смене одного конструктивного объекта другим.

Поскольку алгоритмы могут применяться к весьма произвольным объектам (числам, буквам, словам, графам, логическим выражениям и т.д.), в определении алгоритма используется специальный термин — «конструктивный объект», объединяющий в себе все эти возможные случаи. Так, в алгоритме Евклида под конструктивными объектами можно понимать пары чисел.

Смена конструктивных объектов в алгоритме Евклида может быть представлена следующим образом (для пары $m=10, n=4$):

$$(10, 4) \rightarrow (4, 2) \rightarrow (2, 0).$$

Как правило, для заданного алгоритма можно выделить семь характеризующих его независимых параметров:

- совокупность возможных исходных данных;
- совокупность возможных промежуточных результатов;
- совокупность результатов;
- правило начала;
- правило непосредственной переработки;
- правило окончания;
- правило извлечения результата.

Для алгоритма Евклида эти семь параметров могут быть определены следующим образом:

- 1) $I = \{(m, n) | m \geq n\}$.
- 2) $P = \{(m, n) | m \geq n\}$.
- 3) $R = \{m | m > 0\}$.
- 4) Ввести пару чисел (m, n) таких, что $m \geq n$.
- 5) $(m, n) \rightarrow (n, m \bmod n)$.
- 6) Если в паре (m, n) $n = 0$, то останов.
- 7) Результатом является первое число пары $(m, 0)$.

Вывод m на устройство вывода.

В теории алгоритмов изучаются алгоритмы, заданные в строгом, формализованном виде. Для алгоритма Евклида эта форма может быть такой:

Определяется решающее правило (функция) — f .

$$f: P \rightarrow P,$$

$$f(m, n) = \begin{cases} (m, n), n = 0; \\ (n, m \bmod n), n \neq 0. \end{cases}$$

Поскольку далее будут интересовать практические аспекты (а не теоретические) задания (описания) алгоритмов, на приведенной формализации и остановимся. На практике в программировании очень часто используется задание алгоритмов в виде блок-схем.

Блок-схема — это ориентированный граф, вершины которого могут быть одного из трех типов (рис. 3.2).

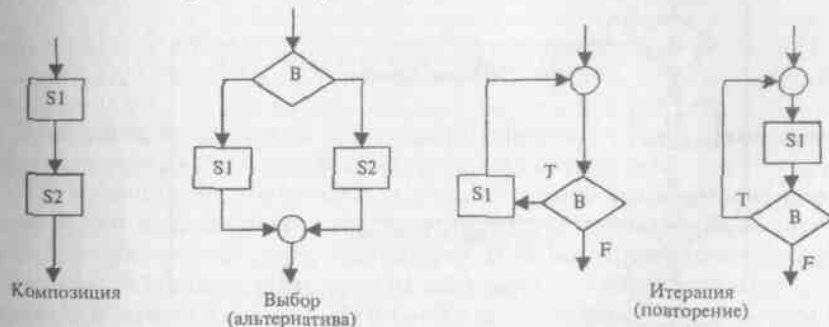


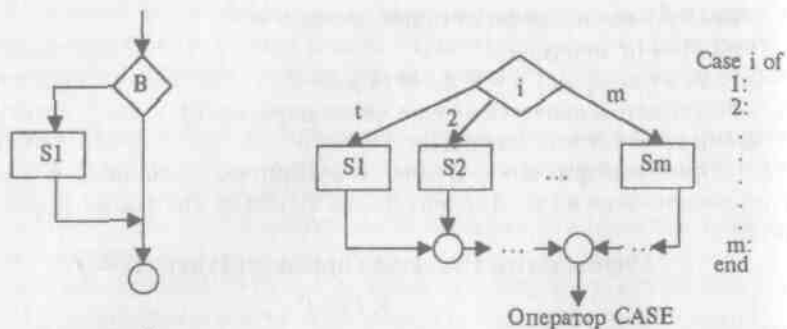
Рис. 3.2. Структуры управления программы

Функциональная вершина используется для представления функции $f: X \rightarrow Y$.

Предикатная вершина используется для представления функции (или предиката) $p: X \rightarrow (T, F)$, т.е. логического выражения, передающего управление по одной из двух возможных ветвей.

Объединяющая вершина представляет передачу управления от одной из двух входящих ветвей к одной выходящей ветви.

Структурная блок-схема — это блок-схема, которая может быть выражена как композиция из четырех элементарных блок-схем (см. рис. 3.2).



Выполнение одной альтернативы

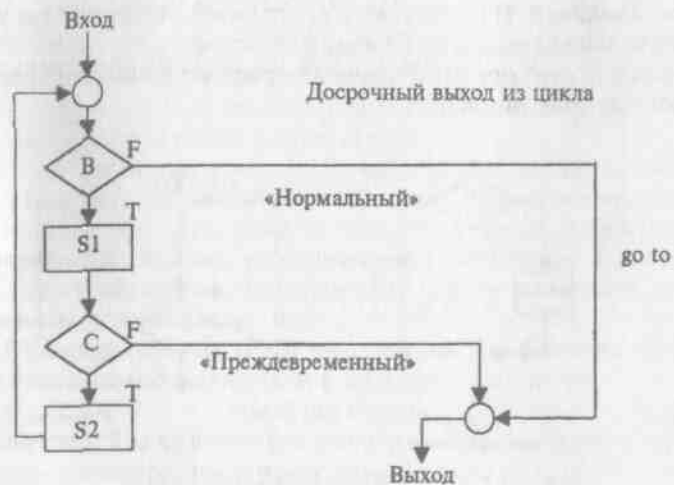


Рис. 3.3. Дополнительные структуры управления программы

Любая программа для машины может быть представлена структурной блок-схемой.

Важной особенностью приведенных структур является то, что они имеют один вход и один выход.

Структурное программирование — процесс разработки алгоритмов с помощью структурных блок-схем.

В более широком плане структурное программирование допускает большее разнообразие элементарных структур управления, чем предложенные четыре (рис. 3.3). Причиной для расширения множества структур является требование удобства и естественности.

Программирование сверху вниз — это процесс пошагового разбиения алгоритма на все более мелкие части с целью получения таких элементов, для которых можно написать конкретные команды. Структурное программирование сверху вниз — это процесс программирования сверху вниз, ограниченный использованием структурных блок-схем.

3.2. МЕТОДЫ РАЗРАБОТКИ АЛГОРИТМА

По всей видимости, существует весьма большое количество всевозможных приемов и методов разработки алгоритмов. Однако среди имеющегося разнообразия этих методов можно выделить небольшой набор основных, в том смысле, что методы из такого набора применяются часто и лежат в основе многих процедур и алгоритмов. Можно утверждать, что знание приводимых ниже методов необходимо для любого программиста. Все обсуждаемые ниже методы рассматриваются на примере решения комбинаторных задач, но сфера их применения, безусловно, намного шире.

3.2.1. МЕТОД ЧАСТНЫХ ЦЕЛЕЙ

Этот метод имеет весьма общую формулировку: «Необходимо свести трудную задачу к последовательности более простых задач».

Приведенная рекомендация выглядит столь естественной и разумной, что вряд ли вызовет у кого-нибудь возражения. Более того, любой человек очень часто использует этот метод решения стоящих перед ним задач, при этом даже не догадываясь (или не отдавая себе отчета) об имеющемся для него (метода) названии. С другой стороны, в конкретной сложной задаче часто очень трудно указать способ ее разбиения на набор более простых задач. Здесь большое значение имеет опыт и искусство специалиста. Тем не менее, несмотря на общность метода и отсутствие «точного рецепта» его применения очень важно освоить этот метод, так как он лежит в основе решения многих задач и по своей сути составляет основу алгоритмизации и программирования. Именно с вопроса «Можно ли данную задачу разбить на последовательность (набор) более простых?» и нужно начинать разработку простого алгоритма.

Неплохой иллюстрацией к применению этого метода является одна из задач сетевого планирования. Подобного рода задачи очень часто приходится решать в системах управления, и их решение является важнейшей функцией АСУ (автоматизированных систем управления).

Задача. Имеется комплекс взаимосвязанных работ. Для каждой из N работ задана ее трудоемкость в человеко-часах. Необходимо расставить K имеющихся в распоряжении рабочих так, чтобы длительность выполнения всего комплекса работ была минимальной. При этом не разрешается динамически (в ходе выполнения комплекса) перемещать рабочих с одной работы на другую.

Взаимосвязанность работ задается с помощью сетевого графика. Он представляет собой ориентированный граф без петель и контуров, состоящий из конечного множества элементов: вершин и попарно соединяющих их дуг (i, j) .

Вершины графа представляют собой события, а дуги — работы.

Работа — это трудовой процесс, для выполнения которого требуются затраты времени и ресурсов (характеристикой этих затрат и является упомянутая выше трудоемкость работы).

Событие представляет собой факт окончания всех предшествующих данному событию работ и возможность начала других работ, непосредственно следующих за данными событиями. Событие наступает в момент времени, соответствующий завершению всех работ, входящих в него.

Каждая работа сетевого графика определяется двумя событиями: предшествующим, которое указывает на начало работы и обозначается i и последующим, указывающим на окончание работы и обозначаемым j . Работа обозначается парой (i, j) . Продолжительность ее выполнения — t_{ij} , а трудоемкость — r_{ij} .

В сетевом графике существуют два особых события: *исходное*, соответствующее началу выполнения работ и обозначаемое 0, и *завершающее*, отображающее завершение намеченной цели данного комплекса работ и обозначаемое последним порядковым номером M . Остальные события графика принято нумеровать, исходя из следующего требования. Нумерация событий должна быть такой, чтобы не существовало некоторой работы (i, j) , для которой $i > j$.

Путь сетевого графика — это любая непрерывная последовательность взаимосвязанных событий и работ по направлению стрелок. Множество путей, соединяющих два события i и k , будем обозначать $L(i, k)$. Наибольший интерес в нашей задаче представляют так называемые *полные пути*, т.е. пути из 0 в M — (0, M). Полный путь сетевого графика с наибольшей продолжительностью называется *критическим* — $I_{кр}$.

$$I_{кр} = L, \min T(l), \\ l \in (0, M),$$

где $T(l)$ — продолжительность пути l , складывающаяся из продолжительности работ, составляющих данный путь. Данные выше определения поясним на конкретном примере. Пусть задан комплекс работ, изображенный на рис. 3.4.

Множество работ сетевого графика $A(N=6)$:

$$A = \{(0,1), (0,2), (1,3), (1,4), (2,4), (3,5), (4,5)\}.$$

Числа над дугами (работами) графа представляют собой трудоемкости работ.

Множество полных путей для нашего графика:

$$L(0,5) = \{I_1, I_2, I_3\},$$

$$\text{где } I_1 = \{(0,1), (1,3), (3,5)\};$$

$$I_2 = \{(0,1), (1,4), (4,5)\};$$

$$I_3 = \{(0,2), (2,4), (4,5)\}.$$

Кроме сетевого графика в постановке задачи дается количество имеющихся в распоряжении рабочих K ; для нашего примера примем $K=10$.

Следует заметить, что если бы K было равно 6 (т.е. $K=N$), то задача имела бы тривиальное решение, так как на каждую работу необходимо было бы поставить по одному рабочему. При $K=7$ появилась бы возможность поставить «лишнего» рабочего на одну из работ комплекса, и всего вариантов такого выбора было бы шесть (количество работ). При $K=8$ возможных вариантов расстановки рабочих было бы уже равным 6^2 , а в нашем случае это число равно 6^4 .

Как видно, уже в случае нашей достаточно малоразмерной задачи количество вариантов велико. Решение задачи путем полного перебора представляется поэтому крайне неэффективным, если вообще возможным.

Одним из самых простых путей решения задачи, который позволяет получить пусть не самое лучшее, но «достаточно хорошее» решение, состоит в следующем. Разбить задачу на 4 шага. На первом шаге решить вопрос, куда поставить 7-го работника, на втором — считать расстановку, полученную на предыдущем шаге, фиксированной (т.е. уже расставленных рабочих полагаем закрепленными за определенными работами), решаем задачу, куда поставить 8-го работника и т.д. Очевидно, что сложную задачу мы разбили на четыре последовательные, более простые задачи. В общем случае количество таких простых задач будет равно $(K-N)$. Каждая из этих однотипных задач может быть сформулирована следующим образом.

Необходимо принять решение о добавлении одного работника на одну из работ комплекса так, чтобы время выполнения всего комплекса по получающейся расстановке было минимальным.

При конкретной расстановке работников к каждой работе (i, j) приписываем k_{ij} — количество работников, и время ее выполнения t_{ij} определяется с помощью простого соотношения $t_{ij} = r_{ij} / k_{ij}$.

В нашей задаче при исходной расстановке шести работников по одному на каждую работу времена на выполнение численно равны трудоемкостям, а длительности полных путей

$$T(I_1) = t_{01} + t_{13} + t_{35} = 1 + 2 + 4 = 7;$$

$$T(I_2) = t_{01} + t_{14} + t_{45} = 1 + 2 + 1 = 4;$$

$$T(I_3) = t_{02} + t_{24} + t_{45} = 2 + 3 + 1 = 6.$$

Продолжительность выполнения всего комплекса работ, таким образом, получается равной 7 часам и определяется продолжительностью самого длинного из полных путей — критическому пути I_1 .

Очевидно, что для того чтобы снизить длительность выполнения всего комплекса, необходимо поместить имеющегося рабочего на одну из работ этого пути.

Путь I_1 состоит из работ $(0,1)$, $(1,3)$ и $(3,5)$. Рассматривая постановку одного рабочего на одну из этих работ и сравнивая получающиеся варианты, выбираем

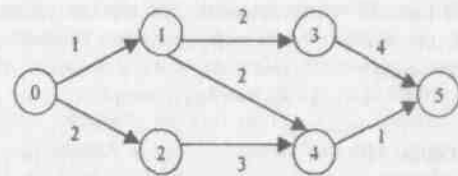


Рис. 3.4. Пример сетевого графика выполнения работ

из них вариант с наименьшей продолжительностью выполнения всего комплекса работ (в нашем случае это будет постановка рабочего на работу (3,5)). Таким образом, задача на добавление одного работника решена, и можно переходить к следующей задаче последовательности.

В общем случае, при поиске решения по данному алгоритму придется проанализировать не более $(K - N) \times N$ вариантов.

Конечно, от нашего алгоритма нельзя ожидать, что он во всех случаях будет находить наилучшую расстановку, но в практике очень часто требуется получить «неплохое решение» с минимальными затратами машинного времени и оперативной памяти; при таких условиях описанный алгоритм представляется приемлемым решением задачи.

3.2.2. МЕТОД ПОДЪЕМА

Этот метод, как и предыдущий, можно отнести к одному из общих «рецептов» разработки алгоритмов. Его суть заключается в следующей процедуре. Алгоритм начинается с принятия начального предположения или построения начального решения задачи. Затем начинается (насколько возможно) быстрое движение «вверх» от начального уровня по направлению к лучшим решениям. Когда алгоритм достигает точки, из которой больше невозможно двигаться «наверх», он останавливается. Попробуем наполнить конкретным содержанием эту общую формулировку. Разберем сам алгоритм подъема и его работу на задаче предыдущего параграфа.

Первым шагом алгоритма подъема является построение начального решения. В нашей задаче о расстановке рабочих начальным решением будет какая-либо расстановка рабочих по работам комплекса. Можно предположить много разных вариантов процедуры начальной расстановки. Рассмотрим одну из простейших. На первом шаге начальной расстановки примем

$k_{ij} = [K/N]$ для любых i, j ($[]$ — означает целую часть числа).

На втором шаге оставшееся количество рабочих $K \bmod N$ распределяется так. Ставится один рабочий на работу с наибольшей трудоемкостью и, если резерв рабочих еще не исчерпан, ищется следующая по трудоемкости работа и проводится постановка очередного рабочего и т.д. Процесс заканчивается, когда резерв рабочих исчерпан. В рассматриваемом примере расстановка может быть следующей: $k_{01} = 1, k_{02} = 2, k_{13} = 1, k_{24} = 2, k_{34} = 2, k_{14} = 1, k_{45} = 1$, а длительности полных путей при начальной расстановке будут:

$$T(I_1) = t_{01} + t_{13} + t_{35} = 1/1 + 2/1 + 4/2 = 5;$$

$$T(I_2) = t_{01} + t_{14} + t_{45} = 1/1 + 2/1 + 1/1 = 4;$$

$$T(I_3) = t_{02} + t_{24} + t_{45} = 2/2 + 3/2 + 1/1 = 3,5.$$

Как видим, путь I_1 оказался критическим путем. Второй элемент алгоритма подъема — процедура восхождения от худшего к лучшему — в

нашей задаче может быть определена так. Надо взять одного рабочего с работы, не лежащей на критическом пути, и переместить на одну из работ критического пути. Если результат — длительность выполнения комплекса работ — улучшен, то повторить процесс, если нет, то попробовать переместить рабочего с другой работы и т.д. Если ни одно из возможных перемещений не приводит к улучшению результата, то алгоритм подъема завершает свою работу.

3.3. ПРОГРАММИРОВАНИЕ С ОТХОДОМ НАЗАД

Основой программ искусственного интеллекта, независимо от того, к чему он прилагается — программированию игр, выбору решений, распознаванию образов и т.п., — является программирование перебора вариантов. Программирование перебора вариантов — это сложная задача, так как алгоритмы перебора ищут решения не по заданным правилам вычислений, а путем проб и ошибок, и схема не укладывается в схемы циклов, имеющихся в языках программирования. Ситуация зачастую осложняется тем, что прямыми методами перебор всех возможных вариантов невозможно осуществить из-за их огромного количества.

Метод программирования с отходом назад позволяет осуществить организованный исчерпывающий поиск требуемого решения задачи. При этом часто удается избежать перебора всех возможных вариантов.

Рассмотрим пример, который демонстрирует основные свойства, общие для всех алгоритмов с отходом назад, и на его основе составим общую блок-схему алгоритмов перебора вариантов с отходом назад.

Задача о восьми ферзях. Найти все способы расстановки на шахматной доске 8 ферзей так, чтобы они не угрожали друг другу, т.е. чтобы на доске 8×8 никакие два ферзя не стояли на одной вертикали, горизонтали или диагонали.

Очевидно, что больше 8 ферзей на доске 8×8 расставить нельзя, так как все ферзи должны стоять на разных горизонталях (или вертикалях), а их всего 8. Покажем, что 8 ферзей расставить можно. Например, так, как показано на рис. 3.5.

Однако значительно труднее подсчитать общее число таких расстановок, в чем собственно и состоит задача.

Множество всех возможных расстановок ферзей, среди которых надо отыскать требуемые, удобно представить с помощью ориентированного дерева и по нему осуществлять поиск нужного варианта (рис. 3.6). Ориентированное дерево — это связный граф без циклов с выделенной вершиной (корнем), в котором между корнем и любой другой вершиной существует единственный путь. У дерева

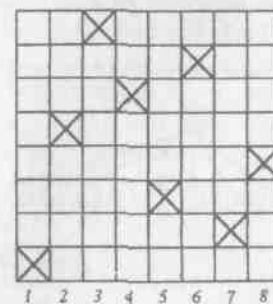


Рис. 3.5. Пример возможной расстановки для задачи о восьми ферзях

будет 9 уровней, номер уровня i определяет номер вертикали, на которую устанавливается i -й ферзь. Восемь ветвей, выходящих из каждой вершины i -го уровня, соответствует восьми возможным горизонтальным позициям ферзя, устанавливаемого на $(i+1)$ -й вертикали.

По условию задачи ферзи не должны бить друг друга. Поэтому из рассмотрения исключаем вершины дерева, которые соответствуют уже занятым горизонталям и диагоналям. Тогда алгоритм перебора возможных вариантов сводится к прохождению всех связанных вершин дерева: движемся вниз по дереву (начиная, например, с левой ветви) до тех пор, пока не достигнем вершины последнего уровня. Если попытка установить $(i+1)$ -го ферзя оказывается неудачной, то отходим назад на один уровень и проверяем, можем ли спуститься опять по другой ветви. Если такой спуск возможен, то опять по самой левой из непройденных ветвей спускаемся вниз (отход в бок), иначе отходим назад еще на один уровень и пытаемся спуститься по следующей из пройденных ветвей и т.д., пока не переберем все 8 горизонтальных полей для $(i+1)$ -й вертикали. Если успешно достигнута вершина последнего уровня, то значит найден очередной требуемый вариант расстановки ферзей. Тогда возвращаемся к корню и начинаем движение вниз по следующей из непройденных ветвей и т.д. до тех пор, пока не останется непройденных ветвей. При такой организации перебора вариантов возможно не больше $8^8 = 2^{24}$ вариантов, что приблизительно равно 10^6 вариантов. На самом деле их будет гораздо меньше, так как должно выполняться условие, что ферзи не бьют друг друга. Введем два понятия, фундаментальные для программирования перебора вариантов с отходом назад: номер хода и номер варианта, и обозначим их соответственно через i и j .

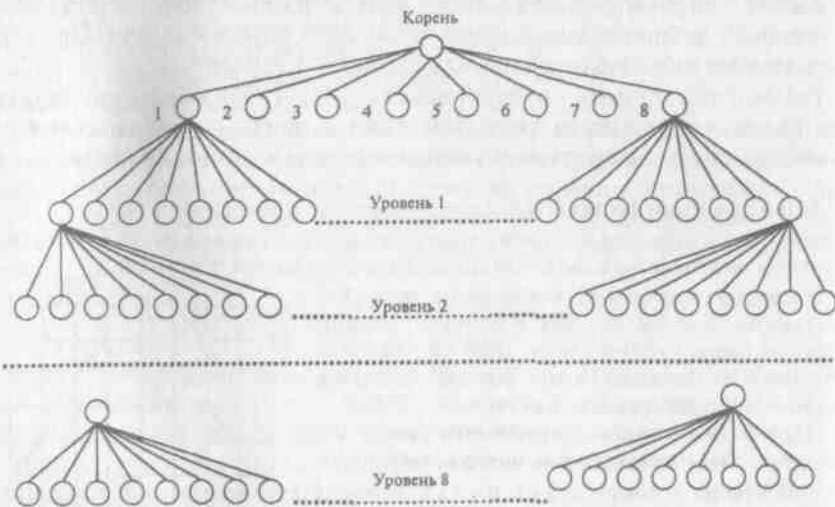


Рис. 3.6. Дерево полного перебора для задачи о восьми ферзях

В нашей задаче номером хода будет порядковый номер ферзя, которого хотим поставить (или уровень нашего графа), а номером варианта — порядковый номер попытки установить этого ферзя, после того, как положение предыдущих ферзей зафиксировано. Размещая одного ферзя за другим, в лучшем случае раз-

местим последнего 8-го ферзя и выведем на печать полученную расстановку ферзей. Однако наверняка придется столкнуться и с такой ситуацией: исчерпав все варианты расстановки i -го ферзя на i -й вертикали (или на i -м уровне дерева), т.е. не сумев его поставить на доску так, чтобы его не били другие $(i-1)$ уже расставленные ферзи, нам надо вернуться на ход назад (на $(i-1)$ -й уровень) и перейти к следующему варианту расстановки предыдущего $(i-1)$ -го ферзя. Для этого надо иметь последний рассмотренный вариант установки $(i-1)$ -го ферзя, т.е. помнить его. Далее, увеличив номер варианта, продолжаем просмотр вариантов установки $(i-1)$ -го ферзя.

Таким образом, мы движемся вперед, увеличивая номер хода. Для каждого очередного хода движемся вбок, подбирая допустимый вариант, и переходим вперед к следующему ходу, если вариант подобран. Если вариант подобрать нельзя, то возвращаемся на ход назад и продолжаем движение вбок, начиная с последнего варианта, рассмотренного на этом ходе. Установив последнего ферзя, записываем решение и идем назад на поиск нового решения и т.д., пока не переберем все возможные пути.

Блок-схема алгоритма перебора вариантов в данной задаче представлена на рис. 3.7.

Договоримся об обозначениях. Возьмем доску размером $n \times n$, n ферзей и для определенности положим $n = 8$.

Будем ставить i -го ферзя на i -ю вертикаль. Для записи положения i -го ферзя введем массив $roza[i]$ размерности n . Тогда в разделе «прописка варианта» будет занесено подходящее j такой, что i -й ферзь, стоящий на пересечении i -й вертикали и j -й горизонтали, не бьется ранее установленными $i-1$ ферзями. Раздел «запись решения и его номера» сводится к печати номера расстановки и массива $roza[i]$, содержащего искомую расстановку. При движении назад нам надо установить номер последнего варианта предыдущего хода. Эта проблема решается автоматически, так как последний вариант из $roza[i]$ будет занесен в j .

При установке первого ферзя нужно будет записать вертикаль и горизонталь, предшествующие i -й вертикали и горизонтали, для этого будем использовать фиктивные вертикаль $i = 0$ и горизонталь $j = 0$.

При установке i -го ферзя необходимо осуществлять проверку, бьется ли поле i -го ферзя предыдущими, уже установленными ферзями. Для этого надо проверить, стоят ли ферзи на одной горизонтали, на одной восходящей (слева снизу — направо вверх) диагонали, и на одной заходящей диагонали (слева сверху — направо вниз).

Для организации проверки введем три логических массива:

- $gr[n]$ — список горизонталей;
- $vd[2n]$ — список восходящих диагоналей;
- $zd[2n]$ — список заходящих диагоналей.

Тогда полю (i, j) , где $0 \leq i \leq n$, $0 \leq j \leq n$, соответствует вертикаль i , горизонталь j , восходящая диагональ $n+j-i$, заходящая диагональ $i+j$ (см. рис. 3.8)

При «прописке» надо будет занести в массивы gr , vd , zd отметку о том, что заняты горизонталь и две соответствующие диагонали того поля, куда установлен новый ферзь ($gr[j]=true$; $vd[n+j-i]=true$; $zd[i+j]=true$). В разделе «выписка», наоборот, надо убрать отметки о занятости этого поля ($gr[j]=false$; $vd[n+j-i]=false$; $zd[i+j]=false$). Это легко осуществить, так как на одной горизонтали или диагонали может находиться только один ферзь.

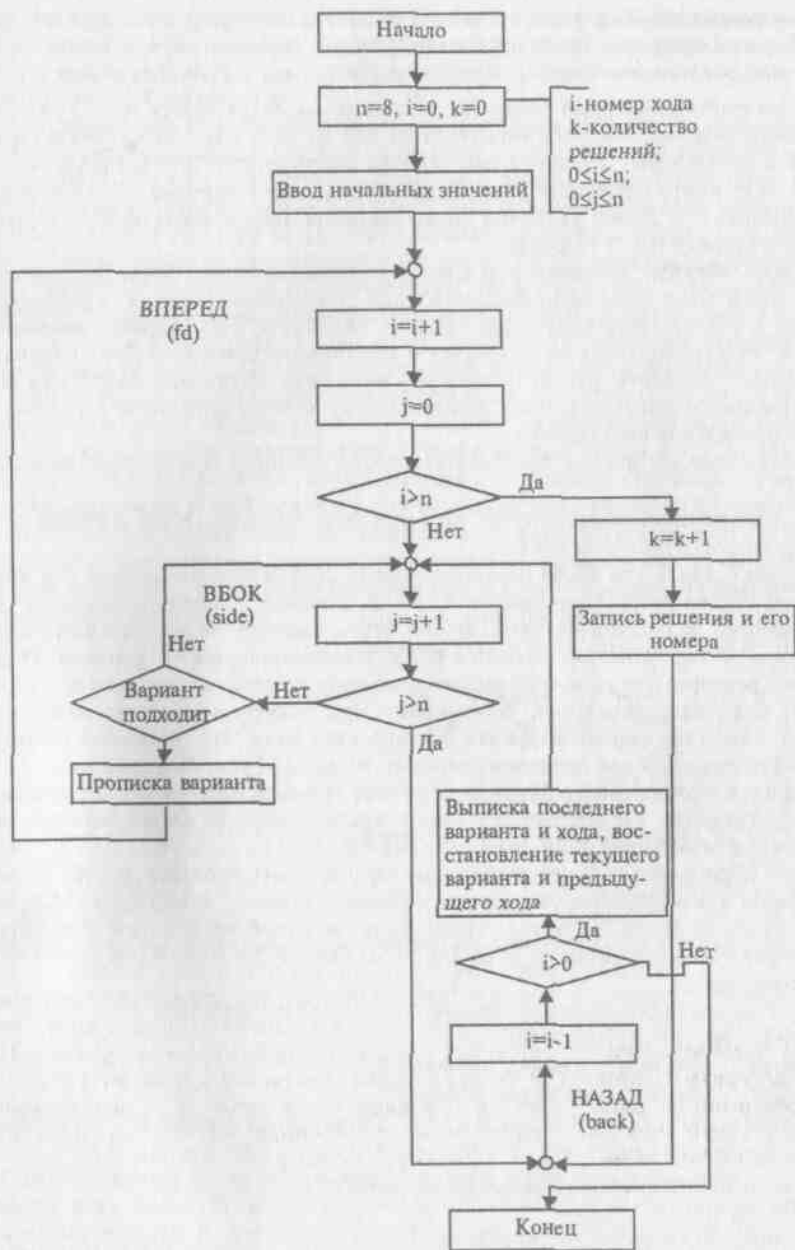


Рис. 3.7. Блок-схема алгоритма перебора вариантов при программировании с отходом назад

Программа «Ферзи»
 program ferzi;
 label fd,
 side, rez, back, prend;
 const n=8; n2=16;
 var i,j,k,m:integer;
 gr:array[1..n]of boolean;
 vd, zd:array[1..n2] of
 boolean;
 poza:array[1..n] of integer;
 BEGIN i:=0; k:=0;
 for m:=1 to n do
 gr[m]:=false;
 for m:=1 to n2 do
 begin
 vd[m]:=false;
 zd[m]:=false;
 end;
 fd: i:=i+1; j:=0;
 if i > n then goto rez;
 side: j:=j+1;
 if j > n then goto back;
 if gr[j] or vd[j+i+n] or zd[j+i] then goto side;
 gr[j]:=true; vd[j-i+n]:=true; zd[j+i]:=true;
 poza[i]:=j; goto fd;
 rez: k:=k+1; write("k=",k,'var=');
 for m:=1 to n do write(poza[m]);
 back: i:=i-1;
 if i=0 then goto prend;
 j:=poza[i];
 gr[j]:=false; vd[j-i+n]:=false;zd[j+i]:=false;
 goto side;
 prend: writeln; writeln ("ONLY", k, 'VAR')
 END

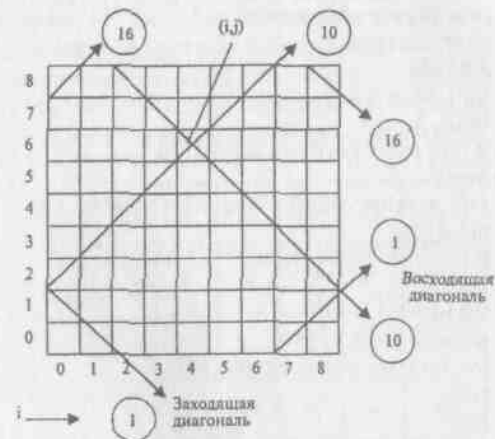


Рис. 3.8. Организация проверки варианта

Наиболее сложным разделом программы перебора вариантов является движение назад и связанная с этим необходимость восстановления номера последнего варианта предыдущего хода. Двигаясь вперед, пересчитываем информацию о позиции, и бывает трудно однозначно восстановить предыдущую. Этот вопрос автоматически решается при рекурсивной организации программы. Поэтому алгоритмы с возвратом часто наиболее естественно выражаются в терминах рекурсии. Однако следует иметь в виду, что в этом случае при движении вперед и приходится переписывать столько информации, что скорость работы становится недопустимо низкой, например, для программирования игр.

Программа «Ферзи» (рекурсивный вариант)
 program queens_rec;
 var m,k:integer; gr:array[1..8] of boolean;
 vd:array[2..16] of boolean;
 zd:array[-7..7] of boolean;

```

poza:array[1..8] of integer;
procedure try(i:integer);
var j,l:integer;
BEGIN
for j:=1 to 8 do
begin
if gr[j] and vd[i+j] end zd[i-j] then
begin
gr[j]:= false; vd[i+j]:=false; zd[i-j]:=false;
poza[i]:=j;
if i<8 then try(i+1)else
begin
k:k+1;
write(" k=" ,k,var=");
for l:=1 to n do write(poza[l]);
end;
end;
gr[j]:=true; vd[i+j]:=true; zd[i-j]:=true
end
END
BEGIN
k:=0; for m:=1 to 8 do gr[m]:=true;
for m:=2 to 16 do vd[m]:=true;
for m:=7 to 1 do zd[m]:=true;
try(1);
END.

```

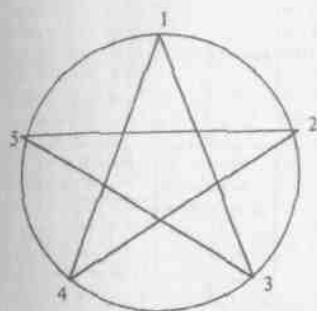
3.4. АЛГОРИТМЫ ВЕТВЕЙ И ГРАНИЦ

Алгоритмы ветвей и границ, как и большая часть алгоритмов, описанных ранее, применяются для решения переборных задач. Как и алгоритм с отходами, они исследуют древовидную модель пространства решений и ориентированы на поиск в некотором смысле оптимального решения (из конечного множества возможных решений-вариантов). В целях упрощения понимания сути алгоритмов такого рода рассмотрим одну конкретную задачу, в которой такой алгоритм весьма хорошо работает и достаточно прост в понимании.

Задача о коммивояжере. Суть задачи заключается в следующем. Дано множество городов. Все города пронумерованы (произвольно) от 1 до N . Возможность проезда из города в город задается при помощи сети (графа). В нашем конкретном примере задачи будем рассматривать сеть городов, изображенную на рис. 3.9.

Кроме того, для каждого пути из города в город задается его некоторая характеристика. Это может быть стоимость проезда, время проезда, расход горючего для проезда и т.п. Если, к примеру, стоимость обратного и прямого про-

ездов для каждой пары городов будут одинаковыми, то матрица стоимостей будет симметричной. В изображенной на рис. 3.9 задаче взят более общий случай, когда стоимости обратного и прямого проездов неодинаковы ($1 \rightarrow 2$ — 18 ед., $2 \rightarrow 1$ — 4 ед.). Необходимо проложить замкнутый путь — тур, который проходит через все города, и, пользуясь которым, посещаем каждый город лишь единожды и затрачиваем при этом минимальную денежную сумму на весь тур.



	1	2	3	4	5
1	∞	18	28	22	24
2	4	∞	12	28	29
3	22	19	∞	8	9
4	7	49	18	∞	9
5	29	16	10	10	∞

Рис. 3.9. Граф переездов и матрица стоимостей для задачи о коммивояжере

Поставленная задача может быть довольно просто решена путем конечного перебора. Множество туров для рассматриваемой задачи можно изобразить следующим образом:

- | | | |
|--|---|---|
| 1. $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 1$ | 9. $1 \rightarrow 3 \rightarrow 4 \rightarrow 2 \rightarrow 5 \rightarrow 1$ | 17. $1 \rightarrow 4 \rightarrow 5 \rightarrow 2 \rightarrow 3 \rightarrow 1$ |
| 2. $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 4 \rightarrow 1$ | 10. $1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 2 \rightarrow 1$ | 18. $1 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow 2 \rightarrow 1$ |
| 3. $1 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 5 \rightarrow 1$ | 11. $1 \rightarrow 3 \rightarrow 5 \rightarrow 2 \rightarrow 4 \rightarrow 1$ | 19. $1 \rightarrow 5 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$ |
| 4. $1 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow 1$ | 12. $1 \rightarrow 3 \rightarrow 5 \rightarrow 4 \rightarrow 2 \rightarrow 1$ | 20. $1 \rightarrow 5 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 1$ |
| 5. $1 \rightarrow 2 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 1$ | 13. $1 \rightarrow 4 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 1$ | 21. $1 \rightarrow 5 \rightarrow 3 \rightarrow 2 \rightarrow 4 \rightarrow 1$ |
| 6. $1 \rightarrow 2 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 1$ | 14. $1 \rightarrow 4 \rightarrow 2 \rightarrow 5 \rightarrow 3 \rightarrow 1$ | 22. $1 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 2 \rightarrow 1$ |
| 7. $1 \rightarrow 3 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 1$ | 15. $1 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 5 \rightarrow 1$ | 23. $1 \rightarrow 5 \rightarrow 4 \rightarrow 2 \rightarrow 3 \rightarrow 1$ |
| 8. $1 \rightarrow 3 \rightarrow 2 \rightarrow 5 \rightarrow 4 \rightarrow 1$ | 16. $1 \rightarrow 4 \rightarrow 3 \rightarrow 5 \rightarrow 2 \rightarrow 1$ | 24. $1 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1$ |

Можно заметить, что в случае полностью связанной сети (каждый город связан с каждым новым путем) количество возможных туров равно величине $(N-1) \cdot (N-2) \cdot \dots = (N-1)!$ ($N=5$, туров — 24 шт.). Безусловно, ни для человека, ни тем более для ЭВМ расчет двадцати четырех (как в нашей задаче) стоимостей туров не представляет сколь-нибудь серьезной задачи. Однако при больших N из-за быстрого роста факториала в общем случае может потребоваться огромное количество расчетов (даже для современных ЭВМ). Поэтому не плохо было бы иметь в своем распоряжении алгоритм, позволяющий сокращать число анализируемых вариантов (туров). Но прежде чем перейти к описанию такого алгоритма, опишем способ организации полного перебора всевозможных туров. В качестве способа организации полного перебора вариантов возьмем уже знакомый по предыдущему параграфу алгоритм с отходами назад. Приводимый ниже способ полного перебора позволяет производить перебор вариантов и в случае неполностью связанной сети

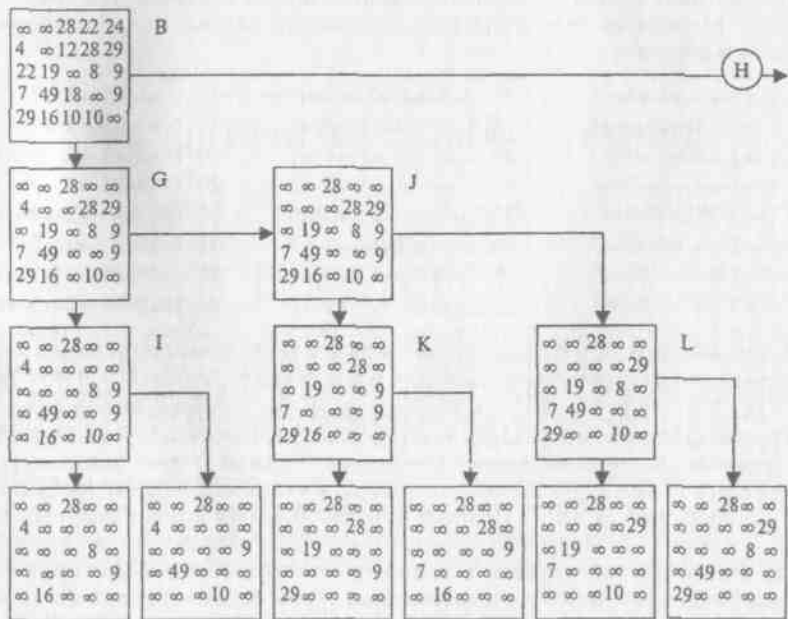
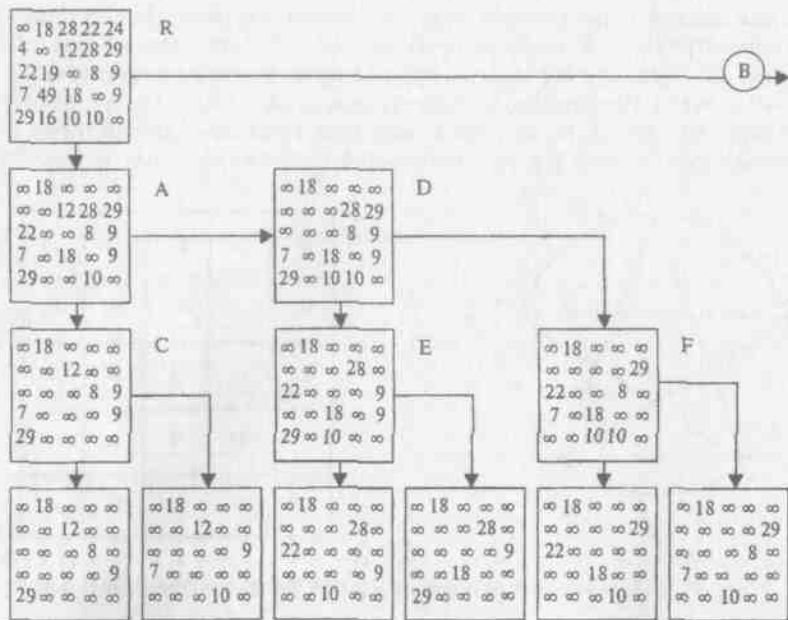
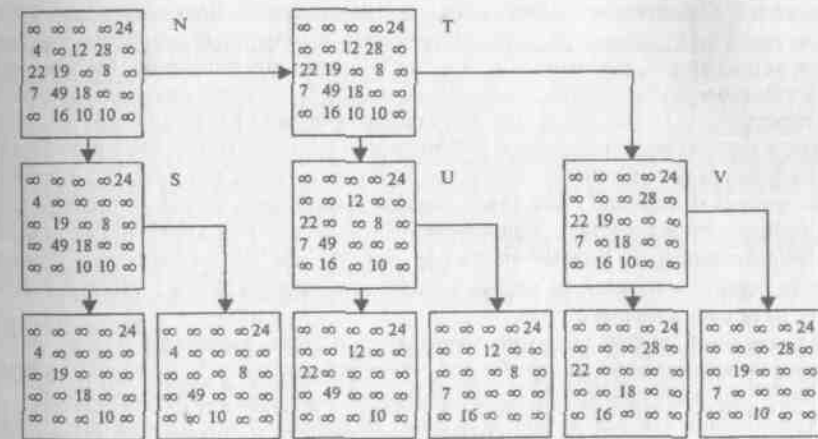
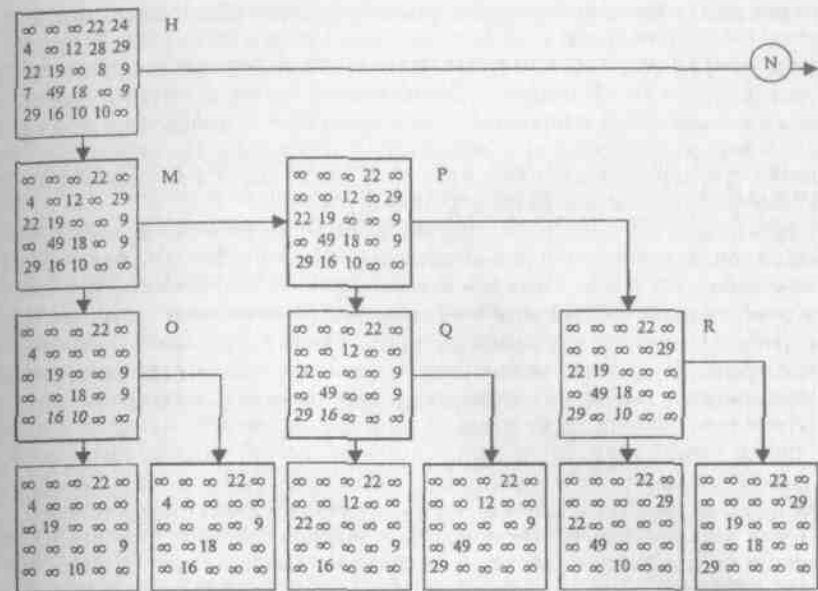


Рис. 3.10. Схема алгоритма перебора



Продолжение рис. 3.10

(когда какие-либо недиагональные элементы матрицы стоимостей равны бесконечности или отсутствует соответствующий перезд). Работа алгоритма перебора иллюстрируется рис. 3.10, а сам алгоритм заключается в следующих манипуляциях с матрицей стоимостей.

В исходной матрице выбирается стоимость, не равная бесконечности, т.е. выбирается путь, принимаемый за базу так называемого *ветвления*. В примере, изображенном на рис. 3.10, в качестве правила для выбора такого пути использовалась следующая процедура. Производился поиск в матрице строки i , имеющей не менее двух стоимостей $< \infty$, и среди $C_{ij} \neq \infty$ выбиралась в качестве базы для ветвления стоимость с наименьшим индексом j . После выбора базы ветвления производится само ветвление. Оно заключается в построении по исходной матрице двух новых матриц стоимостей.

Первая из них получается путем приравнивания бесконечности стоимостей, находящихся в той же строке, что и база ветвления, а затем и стоимостей, находящихся в том же столбце, что и база. Сама базовая стоимость на бесконечность не меняется. После этих действий производится приравнивание бесконечности стоимости C_{ji} — симметричной (относительно главной диагонали) базовой стоимости C_{ij} , производится предотвращение циклов (описание ниже) и затем получившаяся матрица проверяется на то, имеет ли она элемент, являющийся единственным не бесконечным элементом строки (столбца) и в то же время не являющийся таковым в столбце (строке). Если такого элемента нет, то работа по получению первой матрицы завершается, а если таковой найден, то с этим элементом — стоимостью производятся все ранее описанные действия, как будто бы он был выбран в качестве базового для ветвления.

Вторая матрица получается из исходной заменой базового элемента на бесконечность, после чего обработка получившейся матрицы идет так же, как и построение первой матрицы, но начиная с проверки на появление нового базового элемента.

Обратим внимание на смысл проводимых операций. Сущность ветвления заключается в разбиении исходной задачи на две подзадачи. В первой из них, которой соответствует первая из двух построенных в результате ветвления матриц, фиксируется переезд, выбранный в качестве базового при ветвлении. Он фиксируется в том смысле, что любой тур в этой подзадаче будет содержать зафиксированный переезд — (i, j) . Раз из города i производится переезд в город j , то из этого города нельзя производить переезд в любой другой (кроме j) город, что отражается в приравнивании бесконечности элементов i -й строки. Также, раз в город j производится переезд из города i , то в город j нельзя производить переезд из любого другого города, и как следствие приравнивание бесконечности элементов j -го столбца. Приравнивание C_{ji} бесконечности обусловлено тем, что раз из города i производится переезд в город j , то нельзя производить переезд из j в i , так как в этом случае не получим полного тура.

Во второй подзадаче, напротив, запрещается переезд, выбранный в качестве базового, а приравнивание стоимости этого переезда бесконечности и приводит к тому, что любой полный тур в этой подзадаче не будет содержать «базового» переезда.

Получившиеся в результате ветвления матрицы на следующих шагах рассматриваются в качестве исходных, и с ними производятся те же описанные выше действия, что приводит к построению дерева, изображенного на рис. 3.10. Листьями деревьев являются матрицы, описывающие каждый из 24 возможных туров (в рассматриваемой задаче). Стоимости этих туров определяются суммой не равных бесконечности элементов матрицы. Проиллюстрируем описанные выше шаги на примере исходной матрицы (матрицы, помеченной «R», см. рис. 3.10). В качестве базового элемента для ветвления принимается элемент $C_{12} = 18$. Здесь и далее на рисунке переход вниз связан с построением первой матрицы ветвления, а переход ВПРАВО (а также ВПРАВО-ВНИЗ) — второй матрицы.

При построении первой матрицы выполнены следующие действия (матрица А):

- 1) $C_{13} = C_{14} = C_{15} = \infty$;
- 2) $C_{32} = C_{42} = C_{52} = \infty$;
- 3) $C_{21} = \infty$.

При построении второй матрицы $C_{12} = \infty$. Заметим, что в проведенных действиях отсутствовало предотвращение циклов. Раскрыть смысл этой не описанной ранее операции поможет пример. Рассмотрим процесс построения матрицы C из матрицы А. Он состоит из следующих шагов:

- 0) в качестве базового выбран C_{23} ;
- 1) $C_{24} = C_{25} = \infty$ (остальные элементы строки уже равны ∞);
- 2) $C_{43} = C_{53} = \infty$;
- 3) C_{32} уже равен ∞ ;

4) предотвращение циклов. Заметим, что к этому моменту фиксировано два переезда: $1 \rightarrow 2$ и $2 \rightarrow 3$. Переезд из 3 в 2 оказался запрещенным ($C_{32} = \infty$), но осталась еще одна возможность возникновения цикла, а именно, переезд $3 \rightarrow 1$. Если использовать последний, то получится цикл $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$, что противоречит условию, накладываемому на каждый полный тур. Как следствием этих рассуждений $C_{31} = \infty$.

Формализуя процесс предотвращения циклов, можно записать: в качестве базового элемента выбран C_{ij} :

- 1) $C_{ij} = \infty$;
- 2) проверка столбца i : если найден элемент C_{ki} такой, что является единственным, не равным бесконечности в строке, то $C_{kj} = \infty$. Таким же образом начинается обработка столбца k , если такового нет, то процесс предотвращения циклов завершен.

Таким образом, определили полный перебор вариантов. Перейдем к описанию уже собственно алгоритма ветвей и границ, а именно, определим вычисление границ — вторую компоненту (ветвление уже описано) алгоритма. В предлагаемом алгоритме рассчитываются так называемые *нижние границы* для каждой из возникающих подзадач (матриц в дереве перебора).

Основной шаг при вычислении нижних границ называется *приведением*, и оно основано на следующих утверждениях:

1. Каждый тур содержит только один элемент из каждого столбца и каждой строки матрицы стоимостей.

2. Если вычесть константу из каждого элемента некоторой строки (или столбца), то стоимость каждого тура при новой матрице стоимостей C ровно на эту константу меньше стоимости такого же тура при исходной матрице C . Процесс приведения матрицы иллюстрируется на рис. 3.11.

Для каждой строки матрицы выполняется:

- поиск минимального элемента строки;
- вычитание найденного минимального элемента из стоимостей строки.

Аналогичные действия производятся затем и для каждого столбца матрицы.

Сумма констант приведения $h = \sum_{i=1}^{10} h_i = 48$ есть нижняя граница стоимости любого тура в этой задаче. Что касается алгоритма ветвления и получения матриц

подзадач, то он практически не изменяется, за исключением процедуры выбора базового элемента.

∞	18	28	22	24	$h_1=18$
4	∞	12	28	29	$h_2=4$
22	19	∞	8	9	$h_3=8$
7	49	18	∞	9	$h_4=7$
29	16	10	10	∞	$h_5=10$

$h_6=h_7=h_8=h_9=0; h_{10}=1$

Рис. 3.11. Процесс приведения матрицы в задаче о коммивояжере

В качестве кандидатов в базовый элемент рассматриваются элементы матрицы, равные нулю ($C_{12}, C_{21}, C_{34}, C_{35}, C_{41}, C_{53}, C_{54}$; рис. 3.12).

Среди них в качестве базового выбирается элемент, имеющий максимальную сумму минимальных элементов по строке и столбцу, в которых он расположен. Например, в нашем случае:

для C_{12} минимальный в строке $C_{24} = 4$, минимальный в столбце $C_{52} = 6$ и их сумма равна 10;

для C_{21} $8+0 = 8$;

для C_{34} $0+0 = 0$;

для C_{35} $0+1 = 1$;

для C_{41} $1+0 = 1$;

для C_{53} $0+8 = 8$;

для C_{54} $0+0 = 0$.

Следовательно, в качестве базового элемента для ветвления выбирается элемент C_{12} .

	48	56	56	56
∞	0	10	4	5
0	∞	8	24	24
14	11	∞	0	0
0	42	11	∞	1
19	6	0	0	∞

↓ 58 ↓ 72 ↓ 76

Рис. 3.12. Перебор вариантов в задаче о коммивояжере с помощью алгоритма ветвей и границ

Далее производится ветвление (см. рис. 3.12). Строится первая матрица ветвления, производится ее приведение и сумма констант приведения суммируется к уже имевшейся нижней границе (равной 48), что дает 56 как нижнюю границу стоимости любого тура в данной подзадаче. Этот процесс продолжается до по-

лучения первого тура и его стоимости, после чего производится возврат на шаг вверх по дереву и исследование ветвей, порождаемых движением ВНИЗ (т.е. ветвей, порождаемых вторыми матрицами процесса ветвления; на данном рисунке в отличие от предыдущего первые матрицы ветвления изображаются слева от порождающей вершины). Ветвь не имеет смысла исследовать, если нижняя граница для туров в данной подзадаче превышает стоимость наилучшего тура, найденного в данный момент. В нашей конкретной задаче, как видно из представленного рисунка, все ветви в дереве, ведущие вниз, имеют нижние границы, превышающие стоимость тура $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 4 \rightarrow 1$, равную 56. Следовательно, это и есть искомым тур задачи. Рассмотренный пример хорошо показывает, как много вариантов в ряде случаев удается исключить из рассмотрения. Алгоритмы ветвей и границ являются одним из наиболее эффективных методов решения ряда переборных задач. Как правило, эти алгоритмы сложны для понимания, но выигрыш, получаемый в результате их применения, стоит тех усилий, которые приходится затрачивать либо на разработку собственного подобного алгоритма, либо на понимание предлагаемого готового алгоритма.

3.5. ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Первые несколько поколений ЭВМ строились на классических принципах, сформулированных американским математиком Джоном фон Нейманом в 1946 г., когда начались разработки цифровых ЭВМ с программным управлением. Одним из основных принципов Д. фон Неймана является принцип хранимой программы. Под программой вычислительной машины понимается описание алгоритма решения задачи, заданное на языке вычислительной машины. Таким образом, языки программирования — это формальные языки общения человека с ЭВМ, предназначенные для описания совокупности инструкций, выполнение которых обеспечивает правильное решение требуемой задачи, т.е. для описания подлежащих обработке данных (информации) и алгоритмов (программ). Основная роль языков программирования заключается в планировании действий по обработке информации. Любой язык программирования основан на системе понятий, на основе которой человек может выразить свои соображения.

Теоретическую основу языков программирования составляют алгоритмические языки. В настоящее время для ЭВМ разработано значительное количество языков программирования. Они отличаются друг от друга различными свойствами, а значит, и областью применения. Существуют разные подходы к классификации языков программирования. На рис. 3.13 представлена схема, в которой языки программирования систематизированы по возможностям и ориентации на конкретную сферу применения.

Первыми используемыми языками программирования были машинные языки, задаваемые системами команд ЭВМ. Класс машинно-

зависимых языков представлен ассемблером. Язык ассемблера делает доступными все программно-управляемые компоненты ЭВМ. Поэтому он применяется для написания программ, явно использующих специфику конкретной аппаратуры. Ассемблер — это наиболее трудоемкий язык программирования, и из-за его низкого уровня не удастся построить средства отладки, которые существенно снизили бы эту трудоемкость. Программирование на языке ассемблера оперирует с терминами команд аппаратной части машины, и поэтому требует от программиста детальных знаний технических компонентов компьютера. Каждый компьютер имеет такую систему программирования.



Рис. 3.13. Классификация языков программирования

Как правило, эта система программирования разрабатывается и поставляется фирмой — изготовителем ЭВМ. Применение машинных языков требует излишней детализации пути решения задачи, так как даже наиболее современные ЭВМ способны понимать лишь относительно простые инструкции, а человек способен концентрировать детали в выражениях более общего характера. Использование языка ассемблера ограничивается областью системного программирования, т.е. для программирования микропроцессоров, для разработки операционных систем (ОС) или отдельных компонентов ОС, программ обмена между системным блоком и периферийным устройствами (драйверов) и т.д.

Поэтому с развитием программирования появились языки программирования, ориентированные на определенные проблемные области, соответствующие более высокому уровню абстракции. Эти языки получили название алгоритмических языков высокого уровня (АЯВУ). Их отличительной чертой является способность выразить задачу более лаконично и сжато, обеспечивая программиста возможностью более четко

представлять свои задачи и разрабатывать эффективные методы их решения. Преобразование описания задачи на АЯВУ в описании на машинном языке осуществляется специальной программой, называемой транслятором. Уже имеется целый ряд языков программирования, ориентированных на те или иные области применения. Наиболее близкими к машинным языкам являются машинно-ориентированные языки, позволяющие использовать особенности ЭВМ для повышения эффективности программ.

К классу машинно-ориентированных языков можно отнести язык Си. Этот язык является результатом попытки объединить достоинства низкоуровневых возможностей АЯВУ. Язык Си часто называют языком ассемблера со встроенными структурами данных. Использование структур данных позволяет более систематически подходить к реализации задачи на языке Си и сокращает объем текстов разрабатываемых программ. Особенностью данного языка является максимальное использование возможностей конкретной вычислительной архитектуры на основе битовых операций, функций и назначений. Благодаря этому программы на языке Си компактны и работают очень быстро. Однако синтаксис языка достаточно сложен, поэтому чтение текстов программ на нем требует определенного навыка. Язык Си первоначально был ориентирован прежде всего на разработку системных программ. Он, в частности, послужил главным инструментом для создания операционных систем MS DOS и UNIX. В настоящее время язык применяется главным образом для создания системных и прикладных программ, в которых скорость работы и объем памяти являются основными параметрами.

Большинство языков программирования являются машинно-независимыми, что позволяет использовать однажды записанную в таком языке программу на различных ЭВМ. В зависимости от подхода к описанию задачи языки программирования разделяют на процедурно-ориентированные и проблемно-ориентированные языки. **П р о ц е д у р н о о р и е н т и р о в а н н ы е я з ы к и п р о г р а м м и р о в а н и я** представляют программисту средства задания и детального определения последовательности действий, которые ЭВМ должна выполнить для решения задачи, т.е. определяющие «как должна действовать машина». **П р о б л е м н о - о р и е н т и р о в а н н ы е я з ы к и п р о г р а м м и р о в а н и я** (или функциональные) определяют задачу в терминах функции, которую надо выполнить машине, т.е. определяющие «что должно быть сделано». **П р о ц е д у р н о - о р и е н т и р о в а н н ы е я з ы к и** обычно имеют операторную структуру и содержат средства выражения характерных алгоритмических действий, таких как вычисление выражений, проверка условий, циклические вычисления, включения программ и др. **П р о б л е м н о - о р и е н т и р о в а н н ы е я з ы к и** представляют своим пользователям средства определения более или менее широкого набора функций, подлежащих

исполнению. Следует заметить, что четкой границы между этими категориями языков нет, и в пределах одного языка могут сочетаться средства обеих категорий.

Большинство современных языков программирования ориентировано на тот или иной круг задач. Так наиболее широко используемыми в мировой практике вычислений являются:

для обработки экономической информации — Кобол и PL/I;

для решения инженерных и научных задач — Фортран (исторически первый язык высокого уровня);

для обучения программированию — Бейсик, Паскаль, Лого;

для задач искусственного интеллекта — Пролог, Лисп;

для описания задач моделирования дискретных событий — Симула-1, Смолток;

для управления реальными объектами — Модула-2, Ада;

для манипуляции с текстами — Снобол, Комит и др.

Наиболее широко представлен класс универсальных языков программирования. Среди них можно выделить такие популярные языки высокого уровня, как Бейсик, Паскаль, Фортран, Кобол, Модула-2, PL/I и ряд других.

Исторически одним из самых распространенных языков стал Бейсик. Это объясняется, прежде всего, тем, что Бейсик прост в освоении и использовании. Написать на нем небольшую программу в 20 — 30 строк и тут же получить результат ее работы можно буквально за несколько минут. В Бейсик, как правило, встраиваются удобные функции для работы с экраном дисплея, клавиатурой, магнитными накопителями, принтером, коммуникационными каналами. Это позволяет относиться к Бейсику как к продолжению аппаратуры ПЭВМ. Чтобы освоить какую-нибудь особенность или режим работы аппаратных средств, проще всего написать и выполнить соответствующую программу на Бейсике.

Для различных типов ПЭВМ, которые существенно отличаются друг от друга, были разработаны соответствующие версии языка. Для ПЭВМ типа IBM PC и совместимых с ними ПЭВМ наиболее удачной считается версия фирмы Microsoft. Она обеспечивает использование Бейсика для решения задач обработки больших массивов данных (работа с файлами), инженерно-технических и научных расчетов (с помощью большого набора математических функций), обработки текстов (за счет эффективной работы со знаковыми последовательностями), а также решения комплексных задач (за счет отдания оверлейных программных структур). Появление мощных компиляторов, таких, например, как Quik Basic и Visual Basic фирмы Microsoft, поставило этот язык в ряд с другими языками высокого уровня и придает ему дополнительную популярность.

Язык Паскаль можно считать одним из самых распространенных, хотя он и создавался как учебный. Со временем Паскаль зарекомендовал

себя как отличный инструмент для решения серьезных задач, так как его разработчик специально конструировал язык, позволяющий создавать хорошо структурированные программы. Причиной популярности этого языка у пользователей IBM PC и совместимых с ними ПЭВМ стало появление оригинальной версии языка Паскаль — Турбо-Паскаль фирмы Borland International. Турбо-Паскаль характеризуется такими важными особенностями, как полноэкранное редактирование и убавление, графика, звуковое сопровождение и связи с дисковой ОС. Система программирования на Турбо-Паскале сама является резидентной программой. Она позволяет пользователю вводить его программы и выполнять их немедленно, не тратя время на компилирование. Турбо-Паскаль создан как инструмент быстрой разработки не очень больших программ (с числом строк до 500). Более длинные программы приходится сегментировать и использовать оверлейные структуры.

Стремление к созданию подлинно универсального и эффективного инструмента программирования привело к разработке нового языка — Модула-2. Этот язык предложен известным швейцарским ученым Никлаусом Виртом — автором Паскаля.

Основная цель, поставленная при создании этого языка, состояла в том, чтобы обеспечить высокоуровневыми языковыми средствами коллективную разработку надежных, эффективных инструментов программирования и использовать возможности аппаратуры. Таким образом, этот новый язык призван заполнить нишу между Паскалем и Си.

В язык Модула-2 целиком вошли все удачные средства и конструкции языка Паскаль, высокоуровневое представление низкоуровневых возможностей (например, оставаясь на уровне АЯВУ, можно оперировать машинно-независимыми регистрами и отдельными командами).

Язык Фортран — первый язык программирования высокого уровня, активно используется и на современных персональных компьютерах. Близость его конструкции к традиционной архитектуре ЭВМ (имеется в виду традиционная фон-неймановская архитектура) сделала Фортран необычайно популярным. Применяется Фортран главным образом при разработке прикладных систем, ориентированных на научные исследования, инженерные задачи, автоматизацию проектирования и другие области, где накоплены обширные библиотеки стандартных программ.

Язык Кобол был разработан специально для решения экономических задач. В отличие от Фортрана Кобол дает возможность составлять более удобочитаемые программы, которые могут быть понятны и непрограммисту. В программах на Коболе особенно проявляется самодокументируемость, что облегчает их исправление и усовершенствование, а при обработке данных сложной структуры он бывает эффективнее Паскаля. Кобол, будучи широко распространенным на больших и средних машинах, на ПЭВМ используется мало, хотя фирмой Microsoft разработано несколько версий языка для операционных систем MS DOS и XEMIX.

Наиболее удачной версией языка Кобол на сегодняшний день является Кобол/U, в который встроены средства генерации отчетов с использованием языка РПГ. Фирмой IBM в развитие идей Фортрана, Алгола и Кобола был предложен язык PL/I, который получил наибольшее распространение на больших машинах. PL/I разрабатывался как универсальный язык программирования, поэтому располагает большим набором средств обработки цифровой и текстовой информации. Однако эти достоинства делают его весьма сложным для изучения и использования.

Класс проблемно-ориентированных языков представлен гораздо скромнее, чем класс универсальных языков программирования. К этому классу можно отнести языки Лого, РПГ и систему программирования OP88.

Язык Лого был создан с целью обучения школьников основам алгоритмического мышления и программирования. Лого — диалоговый процедурный язык, реализованный на основе интерпретатора с возможностью работы со списками и на их основе с текстами, оснащенный развитыми графическими средствами, которые доступны для детского восприятия. Этот язык реализован для большинства ПЭВМ, применяемых в школах. Хотя язык был разработан задолго до того, как началось массовое распространение персональных компьютеров, именно с их появлением этот язык привлек внимание и приобрел популярность.

РПГ, или генератор отчетов, представляет собой язык, включающий многие понятия и выражения, которые связаны с машинными методами составления отчетов и проектирования форм выходных документов. Язык имеет ограниченную область применения и используется главным образом для печати отчетов, записанных в одном или нескольких файлах базы данных. Многие системы, реализованные на ПЭВМ типа IBM PC и располагающие языком РПГ, имеют также другой язык высокого уровня, применимый для вычислительных процедур, для которых РПГ не подходит. Это относится, прежде всего, к реализации языка Кобол.

Интересные возможности представляет система программирования GPSS фирмы Westi, ориентированная на моделирование систем с помощью событий. В терминах этого языка легко описывается и исследуется класс моделей массового обслуживания, а также другие системы, работающие в реальном масштабе времени.

В последние годы внимание разработчиков программного обеспечения (ПО) обращает на себя объектно-ориентированный подход к программированию, наиболее полно идеология которого реализована в языках Форт и Смолток. Язык Форт относится к числу своеобразных языков. Он сочетает в себе свойства операционной системы, интерпретатора и компилятора одновременно. Основной чертой языка Форт является его открытость, которая позволяет строить новые определения функций на базе ранее определенных. Программист может легко доба-

вить новые операции, типы данных или определения основного языка. Форт позволяет поддерживать многозадачный режим работы, широко используя принцип реентерабельности (одновременного доступа) программ. Структура языка позволяет создавать очень компактные трансляторы. Программирование на этом языке требует специальных навыков, поэтому Форт находит применение при решении сложных задач имитационного моделирования, в графических системах, в системах искусственного интеллекта как средство построения баз знаний, в промышленных разработках. Другим языком, относящимся к классу объектно-ориентированных, является Смолток. Он предназначен для решения нечисловых задач и находит широкое применение при проектировании систем искусственного интеллекта. Программа на этом языке состоит из множества функциональных автономных объектов и средств взаимодействия между ними. Каждый объект содержит некоторую структуру данных и процедуры манипулирования ими. Внешние свойства объекта проявляются в функциях, которые объект выполняет для других объектов. В современных реализациях этого языка широко используется многооконный режим.

К функциональным языкам программирования можно отнести языки Лисп, Пролог и Снобол. Язык Лисп является прекрасным инструментальным средством для построения программ с использованием методов искусственного интеллекта. Имеется несколько реализаций Лисп-трансляторов для ПЭВМ различных классов. Особенность этого языка заключается в удобстве динамического создания новых объектов. В качестве порождаемых программой объектов могут фигурировать и сами программы, которые внешне ничем не отличаются от данных. Это обуславливает возможность построения адаптирующихся и самоизменяющихся программ. Память в языке Лисп используется динамически — когда создается новый объект, то для него из незанятой памяти берется столько ячеек, сколько нужно для хранения всех элементов. При этом не требуется никакого заблаговременного резервирования памяти, как в других языках (например, в Паскале). При уничтожении объекта занятая им память автоматически освобождается. Если язык Лисп известен давно, то язык Пролог получил широкую известность в связи с японским проектом создания вычислительных систем пятого поколения. Одной из первых реализаций был интерпретатор микро-Пролог фирмы Programming Logic Associates для машин фирмы IBM. Затем появилась мощная система программирования Турбо-Пролог, разработанная фирмой Borland International для этих же машин. Она предназначалась для создания широкого класса систем искусственного интеллекта, в том числе персональных экспертных систем. Версия Турбо-Пролог значительно отличается от стандартного Пролога. Отличия, прежде всего, касаются наличия в нем встроенных средств типизации данных и большей структурированности исходных текстов программ.

Язык Снобол располагает мощными средствами манипулирования строками и сравнения с образцом. Главным образом Снобол используется для обработки текстов на естественном языке и применяется в экспертных системах. Известны некоторые версии языка, реализованные для персональных компьютеров, но применение языка ограничено сферой искусственного интеллекта.

Нетрудно заметить, что АЯВУ, который был бы идеальным для всех случаев, не существует. Наиболее важная задача программного обеспечения заключается в том, чтобы определить, какой язык является «наилучшим» в каждой конкретной ситуации. Во многих случаях такой выбор диктуется очень простыми причинами — доступностью того или иного транслятора и умением составлять программы на данном языке. Если, однако, в распоряжении пользователя имеется достаточно большой выбор языков программирования, то необходимо учитывать следующие обстоятельства:

назначение разрабатываемой программы (будет ли она использоваться временно или постоянно, будет ли она модернизироваться и развиваться);

время выполнения программы (имеется ли в виду соотношение вычислительных процедур и процедур ввода-вывода);

ожидаемый размер программы (хватит ли памяти для реализации целиком всей программы или следует ее разделить на отдельные взаимодействующие модули);

необходимость сопряжения разрабатываемой программы с другими пакетами или программами, в том числе составленными на других языках программирования;

предусматривается ли возможность переноса программы на другие типы ЭВМ;

основные типы данных, с которыми будет работать программа (целые и вещественные числа, строки, списки и другие типы структур);

характер и уровень использования аппаратных средств (дисплея, клавиатуры, НМД и др.);

возможность и целесообразность использования имеющихся стандартных библиотек программ, процедур, функций.

С учетом этих критериев возможности языков могут весьма сильно различаться, поэтому правильный выбор АЯВУ является непростой задачей. Для программиста или даже коллектива программистов характерно начинать использование ПЭВМ с языка Бейсик или Паскаль. На Бейсике (и его разновидностях) реализовано довольно много хороших прикладных систем. Появление компиляторов для Бейсика делает этот язык еще более привлекательным, так как позволяет быстро переходить от экспериментальной, интерпретируемой версии программы к ее окончательной версии. В последнее время появились разработки программ на этом языке для систем искусственного интеллекта и экспертных систем.

Появление информационных и вычислительных сетей и распределенных баз данных вызвало появление специализированных языков программирования, обслуживающих эти области использования ЭВМ. Потребовалось создание специальных средств организации человеко-машинного взаимодействия, диалога человека с ЭВМ (QBE, SQL).

3.6. ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Жизненный цикл программного обеспечения включает в себя шесть этапов:

анализ требований,
определение спецификаций,
проектирование,
кодирование,
тестирование,
сопровождение.

Рассмотрим их.

Анализ требований. При разработке программного обеспечения он исключительно важен. Ошибки, допущенные на этом этапе, даже при условии безупречного выполнения последующих этапов могут привести к тому, что разработанный программный продукт не будет соответствовать требованиям практики, сферы его применения. Для создания конкурентоспособных продуктов в ходе выполнения этого этапа должны быть получены четкие ответы на следующие вопросы:

Что должна делать программа?

В чем состоят реальные проблемы, разрешению которых она должна способствовать?

Что представляют собой входные данные?

Какими должны быть выходные данные?

Какими ресурсами располагает проектировщик?

Определение спецификаций. В определенной степени этот этап можно рассматривать как формулировку выводов, следующих из результатов предыдущего этапа. Требования к программе должны быть представлены в виде ряда спецификаций, явно определяющих рабочие характеристики будущей программы. В число таких характеристик могут входить скорость выполнения, объем потребляемой памяти, гибкость применения и др.

Проектирование. На этом этапе создается общая структура программы, которая должна удовлетворять спецификациям; определяются общие принципы управления и взаимодействия между различными компонентами программы.

Кодирование. Заключается в переводе на язык программирования конструкций, записанных на языке проектирования.

Тестирование. На этом этапе производится всесторонняя проверка программ. Тестирование более подробно рассмотрено ниже.

Сопровождение. Это этап эксплуатации системы. Каким бы изощренным ни было тестирование программ, к сожалению, в больших программных комплексах чрезвычайно тяжело устранить абсолютно все ошибки. Устранение обнаруженных при эксплуатации ошибок — первейшая задача этого этапа. Однако это далеко не все, что выполняется при сопровождении. Выполняемый в ходе сопровождения анализ опыта эксплуатации программы позволяет обнаруживать «узкие места» или неудачные проектные решения в тех или иных частях программного комплекса. В результате такого анализа может быть принято решение о проведении работ по совершенствованию разработанной системы. Кроме описанного выше сопровождение может включать в себя проведение консультаций, обучение пользователей системы, оперативное снабжение пользователей информацией о новых версиях системы и т.п. Качественное проведение этапа сопровождения в большой степени определяет коммерческий успех программного продукта.

Рассмотрим этап тестирования программ более подробно. Существуют три аспекта проверки программы на:

- правильность;
- эффективность реализации;
- вычислительную сложность.

Проверка правильности удостоверяет, что программа делает в точности то, для чего она была предназначена. Математическая безупречность алгоритма не гарантирует правильности его перевода в программу. Аналогично, ни отсутствие диагностических сообщений компилятора, ни разумный вид получаемых результатов не дают достаточной гарантии правильности программы. Как правило, проверка правильности заключается в разработке и проведении набора тестов. Кроме этого, для расчета программ иногда можно сверить получаемые решения с уже известным решением. В общем случае, нельзя дать общего решения для проведения проверки на правильность программы.

Проверка вычислительной сложности, как правило, заключается в экспериментальном анализе сложности алгоритма или экспериментальном сравнении двух алгоритмов и более, решающих одну и ту же задачу.

Проверка эффективности реализации направлена на отыскание способа заставить правильную программу работать быстрее или расходовать меньше памяти. Чтобы улучшить программу, пересматриваются результаты реализации в процессе построения алгоритма. Не рассматривая все возможные варианты и направления оптимизации программ, приведем здесь некоторые полезные способы, направленные на увеличение скорости выполнения программ.

Первый способ основан на следующем правиле. Сложение и вычитание выполняются быстрее, чем умножение и деление. Це-

лочисленная арифметика быстрее арифметики вещественных чисел. Таким образом, $X+X$ лучше, чем $2X$, а $i+0,5$ хуже, чем $(2i+j)0,5$ или $(i+i+j)0,5$. При выполнении операций над целыми числами следует помнить, что благодаря применению двоичной системы счисления умножение на числа, кратные двум, можно заменить соответствующим количеством сдвигов влево. Поэтому $10A$ выполняется дольше, чем $Ashl\ 3+ Ashl1$. Второй способ заключается в удалении избыточных вычислений.

Пример.

```
Root 1: = (-b + sqrt(sqr(b) - 4ac))/(2.0a);
Root 2: = (-b - sqrt(sqr(b) - 4ac))/(2.0a);
Лучшим решением является следующее.
DenomA: = a+a;
DenomC: = c+c;
Discrim: = sqrt(bb - DenomA * DenomC);
Root1: = (-b + Discrim)/DenomA;
Root1: = (-b - Discrim)/DenomA.
```

Легко увидеть, что в первом случае потребовалось выполнить четыре операции сложения/вычитания, шесть операций умножения, две операции деления, два вызова функции *sqr* и два вызова функции *sqrt*, во втором — пять операций сложения/вычитания, два умножения, два деления и одно обращение к функции *sqrt*.

Третий способ проверки эффективности реализации основан на способности некоторых компиляторов строить коды для вычисления логических выражений так, что вычисления прекращаются, если результат становится очевидным. Например, в выражении $A \text{ or } B \text{ or } C$, если A имеет значение «истина», то переменные B и C уже не проверяются. Таким образом, можно сэкономить время, разместив переменные A, B, C так, чтобы первой стояла переменная, которая вероятнее всего будет истинной, а последней та, которая реже всего принимает истинное значение.

Однако следует быть осторожным в следующем примере: $Root(A) \text{ or } B \text{ or } C$. $Root(A)$ может и чаще принимает значение «истина», но представляет собой вызов функции, возможно выполняющей сложные и длительные вычисления. Тогда может оказаться, что запись $B \text{ or } C \text{ or } Root(A)$ является более эффективной.

Четвертый прием — исключение циклов.

Пример. Рассмотрим следующий фрагмент:

```
for i: = 1 to 1000 do
A[i]: = 0;
for i: = 1 to 1000 do
for j: = 1 to 10 do
A[i]: = A[i] + C[i,j];
можно переписать так:
for i: = 1 to 1000 do begin
```



```

B := C[i, 1];
for j := 2 to 10 do
A[j] := B B := B + C[i, j];
end.

```

В данном примере выигрыш достигается, во-первых, за счет уменьшения количества циклов (два, а не три), а во-вторых, за счет того, что с введением временной переменной *B* уменьшено количество операций вычисления адресов элементов массива.

Пятый прием — разворачивание циклов.

Пример. Запись
for i := 1 to 1000 do;
for j := 1 to 3 do;
A[i] := A[i] + C[i, j];
можно переписать так: for i := 1 to 1000 do;
A[i] := A[i] + C[i, 1] + C[i, 2] + C[i, 3].

Выигрыш в скорости вычислений вполне очевиден.

Это далеко не полный перечень способов оптимизации. Здесь приведены лишь самые очевидные из них. Следует, кроме того, заметить, что не всегда стоит увлекаться погоней за быстродействием, так как при этом чаще всего ухудшается удобочитаемость программ. В том случае, когда выигрыш получается «мизерный», вряд ли стоит предпочитать его ясности и читабельности программы.

Контрольные вопросы и упражнения

1. Дайте определения терминам «алгоритм» и «алгоритмический процесс».
 2. Что такое блок-схема, алгоритм?
 3. Сформулируйте определение метода частных целей.
 4. Дайте сравнительную характеристику метода подъема и метода частных целей разработки алгоритмов.
 5. В чем суть программирования с отходом назад?
 6. Что такое ветви и границы в алгоритмах ветвей и границ?
 7. Назовите этапы жизненного цикла программного обеспечения.
 8. Опишите основные способы повышения эффективности реализации программы.
- В задачах этой главы необходимо разработать алгоритм и составить программу на алгоритмическом языке Паскаль.
9. В интервале от *a* до *b* найти все парные простые числа. Парными простыми числами называют два простых числа, разность между которыми равна 2. Например, 3 и 5, 11 и 13, 17 и 19.
 10. Найдите все трехзначные числа, сумма цифр которых равна заданному числу *n*.

11. Число Армстронга — такое число из *k* цифр, для которого сумма *k*-х степеней его цифр равна самому числу. Например, $153 = 1^3 + 5^3 + 3^3$. Найти все числа Армстронга из двух, трех и четырех цифр.

12. Палиндром — такое сочетание цифр, которые читаются одинаково слева направо и справа налево. Например, 121, 55, 4884.

Найти все палиндромы, для которых их квадраты также палиндромы (в заданном интервале от *a* до *b*).

13. Счастливым будем считать такое число из шести цифр, в котором сумма левых трех цифр равна сумме правых трех цифр. Например, 457961; $4+5+7 = 9+6+1 = 16$.

Найти все счастливые билеты и подсчитать их количество (номера билетов от 0 до 999999). Если в числе меньше шести цифр, то недостающие начальные цифры считаются нулями.

14. Дано натуральное число *n*. Среди чисел $1, \dots, n$ найти все такие, запись которых совпадает с последними цифрами записи их квадрата (например: $6^2 = 36$, $25^2 = 625$).

15. Найти наименьшее общее кратное двух заданных чисел.

16. Дано натуральное число *n*. Получить все пифагоровы тройки натуральных чисел, каждое из которых не превосходит *n*, т.е. все такие тройки натуральных чисел *a, b, c*, что $a^2 + b^2 + c^2$ ($a \leq b \leq c \leq n$).

17. Найти все совершенные числа в интервале от *a* до *b*. Совершенным называется такое натуральное число, которое равно сумме всех своих делителей, за исключением самого числа, например: $28 = 1+2+4+7+14$.

ГЛАВА 4

ОБЩАЯ ХАРАКТЕРИСТИКА ПРОЦЕССОВ СБОРА, ПЕРЕДАЧИ, ОБРАБОТКИ И ХРАНЕНИЯ ИНФОРМАЦИИ

Любая система, как социально-экономическая, так и система живой и неживой природы, действует в постоянной взаимосвязи с внешней средой — системами более высокого или более низкого уровней. Взаимосвязь осуществляется посредством информации, которая по потокам прямой связи передает цель функционирования, различные команды управления от системы более высокого уровня к системам низового звена, а по потокам обратной связи — все сведения, необходимые для регулирования функционального процесса. Универсальной является не только схема функционирования любой системы управления, но и понятие информации как важнейшего ее элемента, охватывающего все стороны жизнедеятельности. Как указывалось в первой главе, понятие «информация» означает сообщение о каком-либо факте, событии, объекте, явлении и т.п. Однако в теории информации и кибернетике под информацией понимается не каждое сообщение, а лишь такое, которое содержит не известные ранее его получателю факты, дополняющие его представление об изучаемом или анализируемом объекте (процессе). Другими словами, информация — сведения, которые должны снять в большей или меньшей степени существующую до их получения неопределенность у получателя, пополнить систему его понимания объекта полезными сведениями.

Совокупность больших сложных человеко-машинных информационных систем является важнейшей составляющей инфраструктуры общества, где информация выступает одним из главных ресурсов его жизнедеятельности. Поэтому является чрезвычайно важным понимание студентом сложного технологического процесса сбора, передачи и обработки информации. На это и направлено содержание этой главы учебника.

Методические указания

Необходимо подчеркнуть две особенности содержания четвертой главы. Во-первых, сложность технологического процесса сбора, передачи, обработки и хранения информации. И, во-вторых, важность информационных систем как составляющей инфраструктуры общества, где информация выступает одним из главных ресурсов его жизнедеятельности. Поэтому студент из содержания главы должен четко уяснить, что автоматизированная информационная технология представляет собой совокупность методов и способов сбора, передачи, накопления, хранения, поиска и обработки информации на основе применения средств вычислительной техники и связи.

Студент должен знать:

- этапы работы системы восприятия информации;
- типовой процесс обработки сигнала;
- структуру управленческой информации;
- отличительные особенности экономической информации как важнейшей составляющей управленческой информации;
- способы передачи и обработки информации;
- основные режимы взаимодействия пользователя с ЭВМ;
- виды запоминающих устройств для хранения информации.

Студент должен помнить, что так как информация весьма разнообразна по содержанию и виду обслуживаемой ею человеческой деятельности, то каждый вид информации имеет свои особые технологии обработки, смысловую ценность, формы представления и отображения на физическом носителе, требования к точности, достоверности, оперативности отображения фактов, явлений, процессов.

4.1. ВОСПРИЯТИЕ ИНФОРМАЦИИ

Восприятие информации — процесс преобразования сведений, поступающих в техническую систему или живой организм из внешнего мира, в форму, пригодную для дальнейшего использования. Благодаря восприятию информации обеспечивается связь системы с внешней средой (в качестве которой могут выступать человек, наблюдаемый объект, явление или процесс и т.д.). Восприятие информации необходимо для

любой информационной системы, коль скоро она претендует на какую-либо полезность.

Современные информационные системы, создаваемые на базе ЭВМ, в качестве своей составной части имеют более или менее (в зависимости от цели системы) развитую систему восприятия. Система восприятия информации может представлять собой довольно сложный комплекс программных и технических средств. Для развитых систем восприятия можно выделить несколько этапов переработки поступающей информации: предварительная обработка для приведения входных данных к стандартному для данной системы виду, выделение в поступающей информации семантически и прагматически значимых информационных единиц, распознавание объектов и ситуаций, коррекция внутренней модели мира. В зависимости от анализаторов (входящих в комплекс технических средств системы восприятия) организуется восприятие зрительной, акустической и других видов информации. Кроме того, различают статическое и динамическое восприятие. В последнем случае особо выделяют системы восприятия, функционирующие в том же темпе, в каком происходят изменения в окружающей среде. Важнейшей проблемой восприятия информации является проблема интеграции информации, поступающей из различных источников и от анализаторов разного типа в пределах одной ситуации. Кратко рассмотрим процесс восприятия наиболее важного вида информации — зрительной. Можно выделить несколько уровней зрительного восприятия:

1. Получение изображения, поступающего от рецепторов. Как правило, к ЭВМ подключают специальные устройства цифрового ввода изображения, в которых яркость каждой точки изображения кодируется одним или несколькими двоичными числами.

2. Построение образной модели. На этом уровне с помощью специально разработанных алгоритмов происходит обнаружение объектов в описании сцены и разбиение изображений на значимые сегменты. Эффективность алгоритмов анализа сцен определяет скорость работы системы восприятия.

3. Построение образно-семантической модели. На этом уровне за счет информации, имеющейся во внутренней модели внешнего мира, и за счет знаний, хранящихся в ней, опознаются выделенные на предшествующем уровне объекты и между ними устанавливаются пространственные, временные и другие виды отношений. В технических системах на этом уровне восприятия используются методы распознавания образов. Полученные знания о текущей ситуации могут использоваться в дальнейшей работе. Иллюстрацией к описанию уровней восприятия может служить рис. 4.1, поясняющий в общих чертах работу системы зрительного восприятия текстовой информации.

С точки зрения информационной системы в целом, система восприятия осуществляет первичную обработку собираемой извне информации.

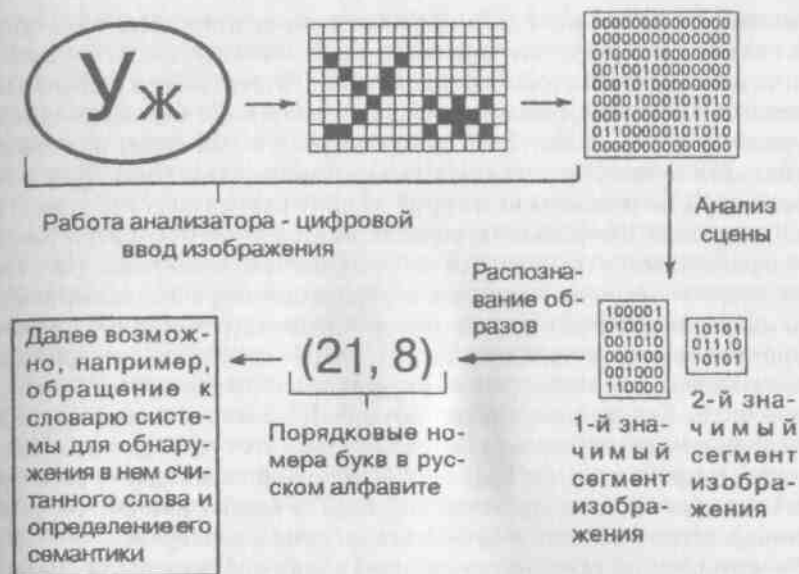


Рис. 4.1. Работа системы зрительного восприятия текстовой информации

В свою очередь, для системы восприятия первичную обработку информации производит система сбора информации. Нередко на практике встречаются информационные системы, не обладающие развитой системой восприятия информации (из-за отсутствия необходимости в такой). В последнем случае система восприятия представляет собой просто систему сбора информации. Вопросам сбора информации и посвящен следующий параграф.

4.2. СБОР ИНФОРМАЦИИ

Из изложенного выше легко сделать вывод, что система сбора информации может представлять собой сложный программно-аппаратный комплекс. Как правило, современные системы сбора информации не только обеспечивают кодирование информации и ее ввод в ЭВМ, но и выполняют предварительную (первичную) обработку этой информации. Сбор информации — это процесс получения информации из внешнего мира и приведение ее к виду, стандартному для данной информационной системы. Обмен информацией между воспринимающей информацией системой и окружающей средой осуществляется посредством сигналов.

Сигнал можно определить как средство перенесения информации в пространстве и времени. В качестве носителя сигнала могут выступать

звук, свет, электрический ток, магнитное поле и т.п. Подобно живым организмам, воспринимающим сигналы из внешней среды с помощью специальных органов (обоняния, осязания, слуха, зрения), технические системы для приема сигналов из окружающего мира оснащаются специальными устройствами. Вне зависимости от носителя информации (сигнала) типичный процесс обработки сигнала может быть охарактеризован следующими шагами. На первом шаге исходный (первичный) сигнал с помощью специального устройства (датчика) преобразуется в эквивалентный ему электрический сигнал (электрический ток). На втором шаге вторичный (электрический) сигнал в некоторый выделенный момент времени оцифровывается специальным устройством — аналого-цифровым преобразователем (АЦП). АЦП значению электрического сигнала ставит в соответствие некоторое число из конечного множества таких чисел. Таким образом, датчик и АЦП, связанные вместе, составляют цифровой измерительный прибор. Если этот прибор оснастить некоторым устройством для хранения измеренной величины — регистром, то на следующем шаге по команде от ЭВМ можно ввести это число в машину и подвергнуть затем любой необходимой обработке.

Конечно, не все технические средства сбора информации работают по описанной схеме. Так, клавиатура, предназначенная для ввода алфавитно-цифровой информации от человека, не имеет в своем составе АЦП. Здесь первичный сигнал — нажатие клавиши — непосредственно преобразуется в соответствующий нажатой клавише цифровой код. Но в любом случае, будь то цифровой измерительный прибор, клавиатура или иное устройство ввода информации в ЭВМ, в конечном счете поступающая в ЭВМ информация представлена в виде цифрового кода — двоичного числа. Современные системы сбора информации (например, в составе автоматизированных систем управления) могут включать в себя большое количество (тысячи) цифровых измерительных приборов и всевозможных устройств ввода информации (от человека в ЭВМ, от ЭВМ к ЭВМ и т.п.). Такое комплексирование средств приводит к необходимости управления процессом сбора информации и к разработке соответствующего программного (и аппаратного) обеспечения. Совокупность технических средств ввода информации в ЭВМ, программ, управляющих всем комплексом технических средств, и программ, обеспечивающих ввод информации с отдельных устройств ввода (драйверов устройств), — вот что представляет собой современная развитая система сбора информации.

Так как информация очень разнообразна по содержанию и виду обслуживаемой ею человеческой деятельности (научная, производственная, управленческая, медицинская, экономическая, экологическая, правовая и др.), то каждый вид информации имеет свои особенные технологии обработки, смысловую ценность, формы представления и отображения на физическом носителе, требования к точности, достоверности, оперативности отражения фактов, явлений, процессов. Технология обработки

информации с применением комплекса технических средств вызывает необходимость манипулировать с отдельными информационными элементами, обеспечивать их изучение и формализованное описание, идентификацию для удобства обработки, хранения и передачи. Информация, представленная в формализованном виде, получила название *данные*.

Информация, являясь сложным по структуре образованием, размещается на физических носителях (бумажных или магнитных документах, в виде сигналов, передаваемых по каналам связи) и может находиться в статичном или динамичном состояниях. Статичное состояние информации связано с ее более или менее длительным организованным хранением, накоплением в информационных фондах и базах данных (БД). Под базой данных понимается вся необходимая для решения задач конкретной области совокупность данных, организованная по определенным правилам, позволяющим обеспечить независимость данных от прикладных программ, удобство хранения, поиска, манипулирования данными, которые записаны на машинных носителях. При этом каждый элемент строго идентифицируется для автоматизации процесса поиска, пополнения, обновления данных. Динамичное состояние — постоянное движение в виде потоков — присуще информации, реализующей в человеко-машинных, автоматизированных системах функцию обмена сведениями с помощью знаковых символов. Приведенные особенности информации тщательно изучаются при создании систем автоматизированной обработки в процессе ее синтаксического, семантического и прагматического анализа.

Синтаксический анализ устанавливает важнейшие параметры информационных потоков, включая необходимые количественные характеристики, для выбора комплекса технических средств сбора, регистрации, передачи, обработки, накопления и хранения информации. Семантический анализ позволяет изучить информацию с точки зрения смыслового содержания ее отдельных элементов, находить способы языкового соответствия (язык человека, язык ЭВМ) при однозначном распознавании вводимых в систему сообщений.

Прагматический анализ проводится с целью определения полезности информации, используемой для управления, выявления практической значимости сообщений, применяемых для выработки управляющих воздействий. Учитывая, что полезность информации является функцией времени и что одна и та же информация в разное время может быть полезной либо бесполезной в зависимости от того, сколько новых сообщений об управляемом объекте она несет пользователю, принятые критерии оценки увязываются с достоверностью и своевременностью поступающих сообщений.

Информация, которая обслуживает процессы производства, распределения, обмена и потребления материальных благ и обеспечивает решение задач организационно-экономического управления народным

хозяйством и его звеньями, называется *управленческой*. Это разнообразные сведения экономического, технологического, социального, юридического, демографического и другого содержания. В информационном процессе, каким является управленческая деятельность, информация выступает как один из возможных ресурсов наряду с энергетическими, материальными, трудовыми, финансовыми. В технологии обработки первичные сведения о производственных и хозяйственных операциях, людях, выпуске продукции, фактах приобретения и продажи товаров выполняют роль предметов труда, а получаемая результатная информация — продукта труда; она используется для анализа и принятия управленческих решений. Важнейшей составляющей управленческой информации является информация экономическая, которая отражает социально-экономические процессы как в сфере производства, так и в производственной сфере, во всех органах и на всех уровнях отраслевого регионального управления. Рассмотрим особенности управленческой информации, в частности экономической, оказывающие влияние на организацию ее автоматизированной обработки.

Экономическая информация отражает акты производственно-хозяйственной деятельности с помощью системы натуральных и стоимостных показателей. Во всех случаях при этом используются количественные величины, цифровые значения. Эта особенность экономической информации предопределяет возможность широкого применения вычислительной техники в экономике. Следующей отличительной чертой экономической информации является ее цикличность. Для большинства производственных и хозяйственных процессов характерна повторяемость составляющих их стадий и информации, отражающей эти процессы. Цикличность экономической информации позволяет, однажды создав программу машинного счета, многократно использовать ее. Это значительно упрощает проектирование автоматизированной обработки данных. Важное значение для обработки имеет форма представления информации. Экономическая информация непременно отражается в материальных носителях: в первичных и сводных документах, в машинных носителях (в магнитных лентах и дисках, в перфолентах), передается по каналам связи. Для повышения достоверности передача и обработка ведутся лишь юридически оформленной информации (при наличии подписи на документе или на электронном сообщении, указании кода, передающего сообщения и т.п.). Отличительной чертой экономической информации является ее объемность. Качественное управление экономическими процессами невозможно без детальной информации о них. Совершенствование управления и возрастание объемов производства сопровождаются увеличением сопутствующих ему информационных потоков. Экономические показатели описывают разные сущности, как простые, так и сложные. Каждая сущность (объект, явление, процесс) имеет определенные свойства. Например, материал обладает весом, га-

баритами, имеет цену, относится к конкретному виду материальных ресурсов и т.п. Совокупность сведений, отражающих какую-либо сущность, называют *информационной совокупностью*. В зависимости от степени общности рассматриваемого объекта информационные совокупности могут быть разного уровня, иметь иерархическую структуру. Данные о поставщике например, включают его имя и адрес, номенклатуру поставляемой продукции, условия поставки, фактические сведения о произведенных поставках и т.д.

Степень детализации информационных совокупностей, однако, небеспредельна. Информационная совокупность, неделимая далее на более мелкие смысловые единицы, получила название *реквизита* по аналогии с реквизитом документа, как наиболее часто используемым в экономической работе носителем информации. Синонимами термина «реквизит» являются *слово, элемент данных, атрибут*, которыми пользуются при описании информационных систем и для определения объемов экономической информации в качестве единиц измерения.

Различают два вида реквизитов: *реквизиты-признаки* и *реквизиты-основания*. Первые характеризуют качественные свойства отражаемых сущностей. Вторые представляют собой количественные величины, характеризующие данную сущность.

Сочетание одного реквизита-основания с одним или несколькими соответствующими ему реквизитами-признаками образует показатель. *Показатель* — качественно определенная величина, дающая количественную характеристику отображаемому объекту (явлению, предмету, процессу). Показатель является информационной совокупностью наименьшего состава, достаточной для образования самостоятельного сообщения или формирования документа. Например, информационная совокупность «500 т стали» состоит из реквизита-основания «500» и реквизитов-признаков — «т» и «сталь», что вполне отражает экономический смысл сообщения и потому является показателем.

Не следует думать, однако, что реквизиты-основания обязательно представляют в числовой форме, а реквизиты-признаки — в текстовой. Важна не форма представления (цифровая или символьная), а то, какое свойство сущности (количественное или качественное) данный реквизит отражает. В любом документе каждый реквизит помимо его значения имеет определенное наименование.

Умение определить количество и состав реквизитов в документе позволяет оценить его уровень информативности, рассчитать при необходимости объемы информации. Зная максимальную разрядность каждого реквизита, легко определить объем информации в документе; зная число таких документов, можно рассчитать общий объем информации.

Информацию, циркулирующую в любом экономическом объекте, можно рассматривать с разных точек зрения в зависимости от целей

анализа. Классификация экономической информации позволяет выделять определенные группы данных, применительно к которым производится изучение информационных потоков.

Наиболее общим направлением классификации экономической информации можно считать ее деление по таким признакам, как место возникновения, участие в процессах обработки и хранения, отношение к функциям управления, стабильность, способ отражения. Рассмотрим, как подразделяется экономическая информация по этим направлениям классификации.

Для управления любым экономическим объектом необходимо располагать и манипулировать определенными сведениями о его фактическом и желаемом состояниях. Совокупность информации конкретного экономического объекта образует информационную систему, в которой различают входящую, исходящую, внутреннюю и внешнюю информацию.

Информацию, поступающую в информационную систему, называют *входящей*. Информационная система, обрабатывая входящие данные, порождает новую — *результатную информацию (сводную)*. Передаваемая за пределы данной информационной системы информация называется *исходящей*. Если сведения поступают в информационную систему от объектов управления, то такая информация будет *входящей внутренней*, если из внешнего мира (например, для предприятий из министерства, от других организаций), информация называется *входящей внешней*.

Аналогичным образом подразделяются и выходящие сведения. Входящую внутреннюю информацию в условиях предприятия называют *первичной*. Она возникает в процессе первичного учета хозяйственных операций — измерение и регистрация данных — в ходе производственной деятельности объекта управления.

По критерию соответствия отражаемым явлениям экономическая информация может быть отнесена к достоверной и недостоверной. К этому классификационному признаку примыкает оценка своевременности и несвоевременности информационного отображения производственных и хозяйственных операций, получения исходной и результатной информации в установленные сроки.

По отношению к процессам обработки и хранения различают следующие виды экономической информации: *исходную, производственную, хранимую без обработки, результатную, промежуточную*.

С точки зрения отражаемых функций управления экономическая информация подразделяется на: *плановую, прогнозную, нормативную, конструкторско-технологическую, учетную, финансовую*.

Важное значение, особенно при создании машинных информационных систем, имеет подразделение информации в зависимости от степени стабильности на *постоянную* (условно-постоянную) и *переменную*. Первая остается без изменений или же подвергается незначительным корректировкам в течение более или менее длительного периода вре-

мени. Это различные справочные сведения, нормативы и расценки и т.п. Переменная информация отражает результаты выполнения производственно-хозяйственных операций, соответствует их динамизму и, как правило, участвует в одном технологическом цикле машинной обработки.

Для оценки уровня стабильности информации используют коэффициент стабильности $K_{ст}$, рассчитываемый по формуле

$$K_{ст} = \frac{ИС_{общ} - ИС_{изм}}{ИС_{общ}}$$

где $ИС_{общ}$ — общее число информационных совокупностей; $ИС_{изм}$ — число информационных совокупностей, изменивших свои значения за рассматриваемый период (год).

Обычно, если значение коэффициента стабильности не ниже 0,85 ($K_{ст} = 0,85$), информационную совокупность принято считать *условно-постоянной*.

Большую часть условно-постоянной информации при использовании вычислительной техники рекомендуется хранить на машинных носителях. При этом отпадает необходимость включать эти реквизиты в состав показателей первичного документа, за счет чего можно значительно упростить их формы, сократить трудоемкость заполнения. Использование массивов условно-постоянной информации в технологии автоматизированной обработки данных обеспечивает повышение достоверности результатной информации, позволяет дополнять ее необходимыми справочными сведениями и тем самым более углубленно и разносторонне охарактеризовать объект, процесс, явление.

Информационная технология решения задач включает следующие важнейшие процедуры, которые могут быть сгруппированы по функционально-временным стадиям: сбор и регистрация информации, передача ее к месту обработки, машинное кодирование данных, хранение и поиск, вычислительная обработка, тиражирование информации, использование информации, т.е. принятие решений и выработка управляющих воздействий.

Как правило, информация подвергается всем процедурам преобразования, но в ряде случаев некоторые процедуры могут отсутствовать. Последовательность их выполнения также бывает различной, но при этом некоторые процедуры могут повторяться. Состав процедур преобразования и особенности их выполнения во многом зависят от экономического объекта, ведущего автоматизированную обработку информации. Рассмотрим особенности выполнения основных процедур преобразования информации.

Сбор и регистрация информации происходят по-разному в различных экономических объектах. Наиболее сложна эта процедура в автоматизированных управленческих процессах промышленных предприятий, фирм и т.п., где производятся сбор и регистрация первичной

учетной информации, отражающей производственно-хозяйственную деятельность объекта.

Особое значение при этом придается достоверности, полноте и своевременности первичной информации. На предприятии сбор и регистрация информации происходят при выполнении различных хозяйственных операций (прием готовой продукции, получение и отпуск материалов и т.п.). Сначала информацию собирают, затем ее фиксируют. Учетные данные могут возникать на рабочих местах в результате подсчета количества обработанных деталей, прошедших сборку узлов, изделий, выявление брака и т.д. Для сбора фактической информации производится измерение, подсчет, взвешивание материальных объектов, получение временных и количественных характеристик работы отдельных исполнителей. Сбор информации, как правило, сопровождается ее регистрацией, т.е. фиксацией информации на материальном носителе (документе или машинном носителе). Запись в первичные документы в основном осуществляется вручную, поэтому процедуры сбора и регистрации остаются пока наиболее трудоемкими. В условиях автоматизации управления предприятием особое внимание придается использованию технических средств сбора и регистрации информации, совмещающих операции количественного измерения, регистрации, накоплению и передаче информации по каналам связи в ЭВМ с целью формирования первичного документа.

4.3. ПЕРЕДАЧА ИНФОРМАЦИИ

Необходимость передачи информации для различных объектов обосновывается по-разному. Так, в автоматизированной системе управления предприятием она вызвана тем, что сбор и регистрация информации редко территориально отделены от ее обработки. Процедуры сбора и регистрации информации, как правило, осуществляются на рабочих местах, а обработка — в вычислительном центре.

Передача информации осуществляется различными способами: с помощью курьера, пересылка по почте, доставка транспортными средствами, дистанционная передача по каналам связи. Дистанционная передача по каналам связи сокращает время передачи данных. Для ее осуществления необходимы специальные технические средства. Некоторые технические средства сбора и регистрации, собирая автоматически информацию с датчиков, установленных на рабочих местах, передают ее в ЭВМ.

Взаимодействие между территориально удаленными объектами осуществляется за счет обмена данными. Доставка данных производится по заданному адресу с использованием сетей передачи данных. В современных условиях большое распространение получила распределенная об-

работка информации, при этом сети передачи данных превращаются в информационно-вычислительные сети. Информационно-вычислительные сети (ИВС) представляют наиболее динамичную и эффективную отрасль автоматизированной технологии процессов ввода, передачи, обработки и выдачи информации. Важнейшим звеном ИВС является канал передачи данных, структурная схема которого представлена на рис. 4.2.



Рис. 4.2. Структурная схема канала передачи данных: УПД — устройство подготовки данных; НКС — непрерывный канал связи; ДКС — дискретный канал связи; УПДс — устройство повышения достоверности

Непрерывный канал связи (НКС) совместно с функционирующими на его концах модемами образует дискретный канал связи (ДКС). В свою очередь, ДКС и устройства повышения достоверности (УПДс) образуют канал передачи данных.

В НКС элементы данных передаются в виде физических сигналов, которые описываются непрерывными функциями времени. Большинство НКС оказываются непригодными для передачи сигналов, отображающих данные, без предварительного их согласования. Для такого преобразования предусматривают специальные устройства — модемы. Модем представляет собой совокупность модулятора и демодулятора. С помощью модулятора информационный сигнал воздействует на некоторый параметр сигнала-переносчика, благодаря чему спектр сигнала смещается в область частот, для которых наблюдается наименьшее затухание в выбранном НКС. Обратную операцию, переход от модулированного сигнала (сигнала-переносчика) к модулирующему (информационному сигналу), осуществляет демодулятор. Понятие ДКС позволяет, отвлекаясь от физической природы процессов, происходящих в НКС, представлять совокупность НКС, и модемов на его концах как некоторый «черный ящик», на вход которого подается последовательность кодовых символов — входное сообщение. Это входное сообщение может представлять собой некоторый текст на русском языке, а может быть, и последовательность нулей и единиц. В первом случае говорят, что входной алфавит ДКС — это обычный алфавит русского языка, во втором — двоичный алфавит (или двоичный код). Аналогичным образом можно описать и примеры для выходного алфавита. В простейшем случае алфавиты на входе и выходе ДКС совпадают. На практике же могут использоваться и ДКС с несовпадающими входным и выходным алфавитами, да и сами алфавиты далеко не ог-

раничены теми примерами, которые были приведены (русский и двоичный). Чаще всего, особенно в теоретических исследованиях и практике вычислительных сетей, рассматриваются ДКС с двоичным алфавитом, когда входное и выходное сообщения представляют собой двоичные кодовые последовательности.

Наконец, завершая общее описание канала связи, рассмотрим УПДс. УПДс может представлять собой специальную аппаратуру, предназначенную для повышения достоверности передачи данных, а может, особенно в современных информационно-вычислительных сетях, представлять собой специальную программу и ЭВМ, на которой она выполняется, может являться как элементом канала связи, так и элементом системы обработки информации. В качестве простейшего способа повышения достоверности передачи информации может использоваться контроль на четность. Суть этого способа заключается в следующем. На входе в канал связи УПД производит подсчет числа «1» в двоичной кодовой последовательности — входном сообщении. Если число «1» оказывается нечетным, в хвост передаваемого сообщения добавляется «1», а если нет, то «0». На принимающем конце канала связи УПД производят аналогичный подсчет, и если контрольная сумма (число «1» в принятой кодовой последовательности) оказывается нечетной, то делается вывод о том, что при передаче произошло искажение информации, в противном случае принятая информация признается правильной (неискаженной). В описанном способе используется один добавочный контрольный разряд. Это позволяет обнаруживать ошибку передачи в случае искажения одного-единственного разряда в сообщении. Этот очень простой способ применяют при передаче данных на большие расстояния. В тех случаях, когда вероятность искажения информации при передаче велика, требуются более изощренные методы, рассмотрение которых требует специальных знаний (прежде всего знания теории вероятности) и выходит за рамки нашего изложения. Но и в последних случаях, когда используется так называемое помехоустойчивое кодирование, очень часто можно выделить некоторую часть, содержащую символы исходной информационной последовательности, и контрольные разряды (их может быть несколько в отличие от нашего примера). Помехоустойчивые коды позволяют не только принимать решение о правильности передачи информации, но и в ряде случаев производить ее исправление. При контроле на четность единственный способ получить достоверную информацию — повторная передача сообщения. В случае корректирующих кодов, что очень важно при высокой стоимости передачи, имеется возможность исправлять ошибки на принимающем конце канала связи, избегая, таким образом, повторной передачи информации.

Дистанционно может передаваться как первичная информация с мест ее возникновения, так и результатная в обратном направлении. В этом случае результатная информация отражается на различных устройствах: дисплеях, табло, печатающих устройствах. Поступление информации по

каналам связи в центр обработки в основном осуществляется двумя способами: на машинном носителе и непосредственно в ЭВМ при помощи специальных программных и аппаратных средств.

Дистанционная передача постоянно развивается и совершенствуется. Особое значение этот способ передачи информации имеет в многоуровневых межотраслевых системах, где применение дистанционной передачи значительно ускоряет прохождение информации с одного уровня управления на другой и сокращает общее время обработки данных.

Машинное кодирование — процедура машинного представления (записи) информации на машинных носителях в кодах, принятых в ЭВМ. Такое кодирование информации осуществляется путем переноса данных первичных документов на магнитные диски, информация с которых затем вводится в ЭВМ для обработки.

Запись информации на машинные носители — трудоемкая операция, в процессе которой возникает наибольшее количество ошибок. Поэтому обязательно выполняются операции контроля записи разными методами на специальных устройствах либо на ЭВМ. Подготовленные и проконтролированные машинные носители хранятся в соответствующем подразделении центра обработки, где ведутся их учет, комплектация, а также выдача для обработки и решения задач на ЭВМ.

4.4. ОБРАБОТКА ИНФОРМАЦИИ

В современных развитых информационных системах машинная обработка информации предполагает последовательно-параллельное во времени решение вычислительных задач. Это возможно при наличии определенной организации вычислительного процесса. Вычислительная задача, формируемая источником вычислительных задач (ИВЗ), по мере необходимости решения обращается с запросами в вычислительную систему. Организация вычислительного процесса предполагает определение последовательности решения задач и реализацию вычислений. Последовательность решения задается, исходя из их информационной взаимосвязи, когда результаты решения одной задачи используются как исходные данные для решения другой. Процесс решения определяется принятым вычислительным алгоритмом. Вычислительные алгоритмы должны объединяться в соответствии с требуемой технологической последовательностью решения задач в вычислительный граф системы обработки информации. Поэтому в вычислительной системе можно выделить систему диспетчирования (СД), которая определяет организацию вычислительного процесса, и ЭВМ (возможно и не одну), обеспечивающую обработку информации.

Каждая вычислительная задача, поступающая в вычислительную систему, может быть рассмотрена как некоторая заявка на обслуживание.

Последовательность вычислительных задач во времени создает поток заявок. В соответствии с требованиями на организацию вычислительного процесса возможно перераспределение поступающих задач на основе принятой схемы диспетчирования. Поэтому в структуре вычислительной системы должны быть предусмотрены соответствующие накопители и устройства диспетчирования, которые обеспечивают реализацию оптимальной организации вычислительного процесса.

На рис. 4.3 представлена обобщенная структурная схема вычислительной системы. ИВЗ формирует входной поток заявок на их решение.

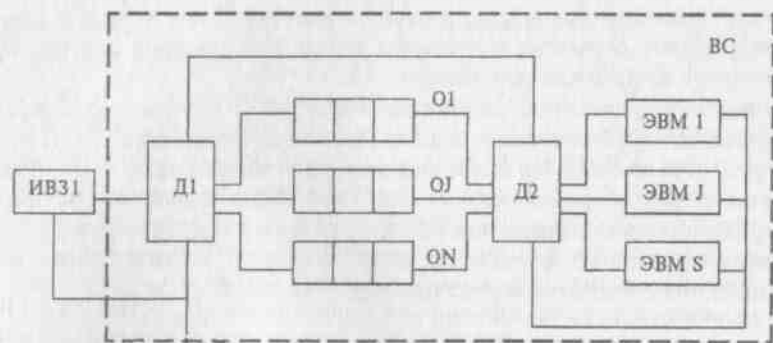


Рис. 4.3. Обобщенная структура вычислительной системы: ИВЗ — информационно-вычислительная заявка; Д — диспетчер; О — очередь заявок на обслуживание

С помощью диспетчера Д1 реализуется обоснование поступившей заявки и постановка ее в очередь О1...ОН, которые реализуются на ячейках оперативной памяти. Заявки отображаются кодами и ожидают начала обслуживания в зависимости от информационной взаимосвязи между задачами. Диспетчер Д2 выбирает из очередей заявку на обслуживание, т.е. передает вычислительную задачу для обработки ЭВМ. Обслуживание обычно осуществляется в соответствии с принятым планом организации вычислительного процесса. Процесс выбора заявки из множества называется диспетчированием. Обычно выбирается заявка, имеющая преимущественное право на обслуживание. При этом инициируется соответствующая программа, реализующая вычислительный алгоритм решения задачи. При отсутствии заявок в очередях диспетчер Д2 переключает процессоры ЭВМ в состояние ожидания. В общем случае в вычислительной системе реализуется параллельное обслуживание за счет наличия нескольких ЭВМ (ЭВМ1...ЭВМS). Можно считать, что процесс обслуживания осуществляется в два этапа. Сначала заявки ставятся в очередь с помощью диспетчера Д1, а на следующем этапе они обслуживаются путем выбора заявок из очереди диспетчером Д2. Диспетчеры Д1 и Д2 реализуются программным путем и представляют собой управляющие программы. Информационные процессы в автоматизированных

системах организационного управления реализуются с помощью ЭВМ и других технических средств. По мере развития вычислительной техники совершенствуются и формы ее использования. Существуют разнообразные способы доступа и общения с ЭВМ. Индивидуальный и коллективный доступ к вычислительным ресурсам зависит от степени их концентрации и организационных форм функционирования. Централизованные формы применения вычислительных средств, которые существовали до массового использования ПЭВМ, предполагали их сосредоточение в одном месте и организацию информационно-вычислительных центров (ИВЦ) индивидуального и коллективного пользования (ИВЦКП).

Деятельность ИВЦ и ИВЦКП характеризовалась обработкой больших объемов информации, использованием нескольких средних и больших ЭВМ, квалификационным персоналом для обслуживания техники и разработки программного обеспечения. Централизованное применение вычислительных и других технических средств позволяло организовать их надежную работу, планомерную загрузку и квалификационное обслуживание. Централизованная обработка информации наряду с рядом положительных сторон (высокая степень загрузки и высокопроизводительное использование оборудования, квалифицированный кадровый состав операторов, программистов, инженеров, проектировщиков вычислительных систем и т.п.) имела ряд отрицательных черт, порожденных прежде всего отрывом конечного пользователя (экономиста, плановика, нормировщика и т.п.) от технологического процесса обработки информации.

Децентрализованные формы использования вычислительных ресурсов начали формироваться со второй половины 80-х годов, когда сфера экономики получила возможность перейти к массовому использованию персональных ЭВМ (ПЭВМ). Децентрализация предусматривает размещение ПЭВМ в местах возникновения и потребления информации, где создаются автономные пункты ее обработки. К ним относятся абонентские пункты (АП) и автоматизированные рабочие места.

Обработка экономической информации на ЭВМ производится, как правило, централизованно, а на мини- и микроЭВМ — в местах возникновения первичной информации, где организуются автоматизированные рабочие места специалистов той или иной управленческой службы (отдела материально-технического снабжения и сбыта, отдела главного технолога, конструкторского отдела, бухгалтерии, планового отдела и т.п.). Автоматизированное рабочее место (АРМ) специалиста включает персональную ЭВМ (ПЭВМ), работающую автономно или в вычислительной сети, набор программных средств и информационных массивов для решения функциональных задач. Обработка экономической информации на ПЭВМ начинается при полной готовности всех устройств машины. Оператор или пользователь при выполнении работы на ПЭВМ руководствуется специальной инструкцией по эксплуатации технических и программных средств.

В начале работы в машины загружаются программа и различные информационные массивы (условно-постоянные, переменные, справочные), каждый из которых сначала, как правило, обрабатывается для получения каких-либо резульатных показателей, а затем массивы объединяются для получения сводных показателей.

При обработке экономической информации на ЭВМ выполняются арифметические и логические операции. Арифметические операции обработки данных в ЭВМ включают все виды математических действий, обусловленных программой. Логические операции обеспечивают соответствующее упорядочение данных в массивах (первичных, промежуточных, постоянных, переменных), подлежащих дальнейшей арифметической обработке. Значительное место в логических операциях занимают такие виды сортировальных работ, как упорядочение, распределение, подбор, выборка, объединение. В ходе решения задач на ЭВМ, в соответствии с машинной программой, формируются резульатные сводки, которые печатаются машиной. Печать сводок может сопровождаться процедурой тиражирования, если документ с резульатной информацией необходимо предоставить нескольким пользователям.

Технология электронной обработки информации — человеко-машинный процесс исполнения взаимосвязанных операций, протекающих в установленной последовательности с целью преобразования исходной (первичной) информации в резульатную. Операция представляет собой комплекс совершаемых технологических действий, в результате которых информация преобразуется. Технологические операции разнообразны по сложности, назначению, технике реализации, выполняются на различном оборудовании, многими исполнителями. В условиях электронной обработки данных преобладают операции, выполняемые автоматически на машинах и устройствах, которые считывают данные, выполняют операции по заданной программе в автоматическом режиме при участии человека или сохраняя за пользователем функции контроля, анализа и регулирования.

Построение технологического процесса определяется следующими факторами: особенностями обрабатываемой информации, ее объемом, требованиями срочности и точности обработки, типами, количеством и характеристиками применяемых технических средств. Они ложатся в основу организации технологии, которая включает установление перечня, последовательности и способов выполнения операций, порядка работы специалистов и средств автоматизации, организацию рабочих мест, установление временных регламентов взаимодействия и т.п. Организация технологического процесса должна обеспечить его экономичность, комплексность, надежность функционирования, высокое качество работ. Это достигается использованием системотехнического подхода к проектированию технологии и решения экономических задач. При этом имеет место комплексное взаимосвязанное рассмотрение всех

факторов, путей, методов построения технологии, применение элементов типизации и стандартизации, а также унификации схем технологических процессов.

Технология автоматизированной обработки информации строится на следующих принципах:

интеграции обработки данных и возможности работы пользователей в условиях эксплуатации автоматизированных систем централизованного хранения и коллективного использования данных (банков данных);

распределенной обработки данных на базе развитых систем передачи;

рациональное сочетание централизованного и децентрализованного управления и организации вычислительных систем;

моделирование и формализованное описание данных, процедур их преобразования, функций и рабочих мест исполнителей;

учет конкретных особенностей объекта, в котором реализуется машинная обработка информации.

Организация технологии обработки информации на отдельных ее этапах имеет свои особенности, что дает основание для выделения внемашиной и внутримашинной технологии. *Внемашиная технология* (ее нередко именуют предбазовой) объединяет операции сбора и регистрации данных, запись данных на машинные носители с контролем. *Внутримашинная технология* связана с организацией вычислительного процесса в ЭВМ, организацией массивов данных в памяти и их структуризацией, что дает основание называть ее еще и внутривазовой. Учитывая, что средствам, составляющим техническую базу внемашиного и внутримашинного преобразования информации, посвящены последующие главы учебного пособия, кратко рассмотрим лишь особенности построения названных технологий.

Основной этап информационного технологического процесса связан с решением функциональных задач на ЭВМ. Внутримашинная технология решения задач на ЭВМ, как правило, реализует следующие типовые процессы преобразования экономической информации: формирование новых массивов информации; упорядочение информационных массивов; выборка из массива некоторых частей записи, слияние и разделение массивов; внесение изменений в массив; выполнение арифметических действий над реквизитами в пределах записей, в пределах массивов; над записями нескольких массивов. Решение каждой отдельной задачи или комплекса задач требует выполнения следующих операций: ввод программы машинного решения задачи и размещения ее в памяти ЭВМ; ввод исходных данных; логический и арифметический контроль введенной информации; исправление ошибочных данных; компоновка входных массивов и сортировка введенной информации; вычисления по заданному алгоритму; получение выходных массивов информации; редактирование выходных форм; вывод информации на экран и

машинные носители; печать таблиц с выходными данными. Выбор того или иного варианта технологии определяется прежде всего как объемно-временными особенностями решаемых задач, периодичностью, срочностью, требованиями к скорости связи пользователя с ЭВМ, так и режимных возможностей технических средств — в первую очередь ЭВМ.

Различают следующие режимы взаимодействия пользователя с ЭВМ: пакетный и интерактивный (запросный, диалоговый). Сами же ЭВМ могут функционировать в следующих режимах: одно- и многопрограммном, разделении времени, реального времени, телеобработки. При этом предусматривается цель удовлетворения потребности пользователей в максимально возможной автоматизации решения разнообразных задач.

Пакетный режим был наиболее распространен в практике централизованного решения экономических задач, когда большой удельный вес занимали задачи отчетности о производственно-хозяйственной деятельности экономических объектов разного уровня управления. Организация вычислительного процесса при пакетном режиме строилась без доступа пользователя к ЭВМ. Его функции ограничивались подготовкой исходных данных по комплексу информационно-взаимосвязанных задач и передачей их в центр обработки, где формировался пакет, включающий задание для ЭВМ на обработку, программы, исходные, нормативно-расценочные и справочные данные. Пакет вводился в ЭВМ и реализовывался в автоматическом режиме без участия пользователя и оператора, что позволяло минимизировать время выполнения заданного набора задач. При этом работа ЭВМ могла проходить в однопрограммном или многопрограммном режиме, что предпочтительнее, так как обеспечивалась параллельная работа основных устройств машины. В настоящее время пакетный режим реализуется применительно к электронной почте.

Интерактивный режим предусматривает непосредственное взаимодействие пользователя с информационно-вычислительной системой, может носить характер запроса (как правило, регламентированного) или диалога с ЭВМ.

Запросный режим необходим пользователям для взаимодействия с системой через значительное число абонентских терминальных устройств, в том числе удаленных на значительное расстояние от центра обработки. Такая необходимость обусловлена решением оперативных задач справочно-информационного характера, какими являются, например, задачи резервирования билетов на транспорте, номеров в гостиничных комплексах, выдача справочных сведений и т.п. ЭВМ в подобных случаях реализует систему массового обслуживания, работает в режиме разделения времени, при котором несколько независимых абонентов (пользователей) с помощью устройств ввода-вывода имеют в процессе решения своих задач непосредственный и практически одно-

временный доступ к ЭВМ. Этот режим позволяет дифференцированно в строго установленном порядке предоставлять каждому пользователю время для общения с ЭВМ, а после окончания сеанса отключать его.

Диалоговый режим открывает пользователю возможность непосредственно взаимодействовать с вычислительной системой в допустимом для него темпе работы, реализуя повторяющийся цикл выдачи задания, получения и анализа ответа. При этом ЭВМ сама может инициировать диалог, сообщая пользователю последовательность шагов (представление меню) для получения искомого результата.

Обе разновидности интерактивного режима (запросный, диалоговый) основываются на работе ЭВМ в режимах реального времени и телеобработки, которые являются дальнейшим развитием режима разделения времени. Поэтому обязательными условиями функционирования системы в этих режимах являются, во-первых, постоянное хранение в запоминающих устройствах ЭВМ необходимой информации и программ и лишь в минимальном объеме поступление исходной информации от абонентов и, во-вторых, наличие у абонентов соответствующих средств связи с ЭВМ для обращения к ней в любой момент времени.

Рассмотренные технологические процессы и режимы работы пользователей в системе «человек — машина» особенно четко проявляются при интегрированной обработке информации, которая характерна для современного автоматизированного решения задач в многоуровневых информационных системах.

Хранение и накопление информации вызвано многократным ее использованием, применением постоянной информации, необходимостью комплектирования первичных данных до их обработки.

Хранение информации осуществляется на машинных носителях в виде информационных массивов, где данные располагаются по установленному в процессе проектирования группировочному признаку.

Поиск данных — это выборка нужных данных из хранимой информации, включая поиск информации, подлежащей корректировке или замене запроса на нужную информацию.

Развитие организационных форм вычислительной техники строится на сочетании централизованной и децентрализованной — смешанной — форм. Предпосылкой появления смешанной формы явилось создание сетей ЭВМ на основе различных средств связи. Сети ЭВМ предполагают объединение в систему с помощью каналов связи вычислительных средств, программных и информационных ресурсов (баз данных, баз знаний). Сетями могут охватываться различные формы использования ЭВМ, причем каждый абонент имеет возможность доступа не только к своим вычислительным ресурсам, но и к ресурсам всех остальных абонентов, что создает ряд преимуществ при эксплуатации вычислительной системы.

В последнее время организация применения компьютерной техники претерпевает значительные изменения, связанные с переходом к созданию интегрированных информационных систем. *Интегрированные информационные системы* создаются с учетом того, что они должны осуществлять согласованное управление данными в пределах предприятия (организации), координировать работу отдельных подразделений, автоматизировать операции по обмену информацией как в пределах отдельных групп пользователей, так и между несколькими организациями, отстоящими друг от друга на десятки и сотни километров. Основой для построения подобных систем служат локальные вычислительные сети (ЛВС). Характерной чертой ЛВС является предоставление возможности пользователям работать в универсальной информационной среде с функциями коллективного доступа к данным.

В последние 3 — 4 года компьютеризация вышла на новый уровень: активно создаются вычислительные системы различной конфигурации на базе персональных компьютеров (ПК) и более мощных машин. Состоящие из нескольких автономных компьютеров с общими совместно используемыми внешними устройствами (диски, ленты) и единым управлением, они позволяют обеспечить более надежную защиту компьютерных ресурсов (устройств, баз данных, программ), повысить отказоустойчивость, обеспечить простоту модернизации и наращивания мощности системы.

Все больше внимания уделяется развитию не только локальных, но и распределенных сетей, без которых немислимо решение современных задач информатизации.

В зависимости от степени централизации вычислительных ресурсов роль пользователя и его функции меняются. При централизованных формах, когда у пользователей нет непосредственного контакта с ЭВМ, его роль сводится к передаче исходных данных на обработку, получению результатов, выявлению и устранению ошибок. При непосредственном общении пользователя с ЭВМ его функции в информационной технологии расширяются. Он сам вводит данные, формирует информационную базу, решает задачи, получает результаты, оценивает их качество. У пользователя открываются реальные возможности решать задачи с альтернативными вариантами, анализировать и выбирать с помощью системы в конкретных условиях наиболее приемлемый вариант. Все это реализуется в пределах одного рабочего места. От пользователя при этом требуется знание основ информатики и вычислительной техники.

В завершение данного параграфа заметим, что процесс обработки информации был описан на самом верхнем уровне («вид сверху»). Более детальное рассмотрение этого процесса, изучение его характеристик и различных моделей обслуживания (диспетчирования) является содержанием специальных дисциплин.

4.5. ХРАНЕНИЕ ИНФОРМАЦИИ

Для хранения больших массивов информации применяются разнообразные запоминающие устройства. Очень большие массивы информации хранятся на внешних запоминающих устройствах (ВЗУ). К ним относятся запоминающие устройства на:

- дисках (ЗУД);
- магнитном барабане (МБ);
- магнитных лентах (МЛ);
- перфорационных лентах (ПЛ);
- магнитных картах (МК).

Приведенные ВЗУ относятся к классу ЗУ с перемещением носителя информации. Достоинством таких ЗУ, наряду с большой емкостью, является низкая стоимость хранения единицы информации, а недостатком — наличие механических узлов перемещения, накладывающим ограничением на скорость работы. С точки зрения организации хранения информации ВЗУ подразделяются на ЗУ с несменяемым носителем, или ЗУ со сменным носителем (МБ, МК, Д), позволяющими создавать библиотеки и архивы с практически неограниченным объемом данных. Перемещение носителя при считывании может быть непрерывным (МБ, ЗУД) или старто-стопным (МЛ, ПЛ, Д), при котором оно происходит только во время обращения к ВЗУ. Выборка блоков информации из ЗУ при этом производится по принципу последовательного или произвольного обращения. В последнем случае блок информации с произвольным адресом выбирается за постоянный промежуток времени. По организации связи различают ВЗУ, работающие под управлением машины (подключающиеся к ней автоматически без вмешательства оператора) и неуправляемые машиной (требующие участия оператора в установке блоков с хранимой информацией). Перспективы ВЗУ, использующие фотооптический способ с высокоскоростным сканированием, способ термопластической записи с применением записи с оптическим воспроизведением и др.

Запоминающее устройство на дисках — это ЗУ, в котором в качестве накопителя используется магнитный диск. Оно состоит из накопителя (пакет дисков), блока выборки (набор магнитных головок с пневматическим или гидравлическим приводом и электронная система преобразования кода адреса в соответствующее перемещение головок), блока записи-считывания числа (набор усилителей воспроизведения и записи) и блока местного управления. Скорость работы ЗУ определяется скоростью вращения дисков и принятой системой выборки. Среднее время обращения к ЗУД составляет 15 — 150 мс, а емкость — $2 \cdot 10^7$ — $2 \cdot 10^9$ бит. Используется в качестве внешнего запоминающего устройства для хранения больших объемов информации и больших библиотек программ.

Запоминающее устройство на магнитном барабане — это ЗУ, в котором в качестве накопителя информации применяют магнитный барабан (МБ). Для реализации обращения к ЗУ на МБ используется дорожка синхронизирующих маркеров, нанесенная на МБ, в процессе изготовления. Маркерные сигналы, считываемые магнитной головкой, после усиления подаются на счетчик адресов, устанавливаемый в нулевое положение перед приходом первого синхронизирующего маркера. Содержимое счетчика сравнивается с содержимым регистра адреса. В момент совпадения показателя счетчика с кодом адреса, заданным командой, выдается сигнал обращения, по которому производится запись или считывание числа. Время обращения к ЗУ на МБ определяется временем оборота МБ и составляет десятки миллисекунд. На поверхности МБ размещается до 10^7 — 10^8 бит информации. ЗУ на МБ используют в основном в качестве запоминающего устройства внешнего.

Контрольные вопросы

1. Что такое восприятие информации?
2. Перечислите основные этапы переработки информации в системах восприятия информации.
3. Что такое сбор информации?
4. Опишите типичный процесс сбора информации.
5. Что такое канал связи?
6. Охарактеризуйте непрерывный и дискретный каналы связи.
7. Назовите основные элементы канала связи.
8. Опишите простейший способ повышения достоверности передачи информации.
9. Дайте обобщенную структурную схему вычислительной системы.
10. Как Вы понимаете преимущества и недостатки централизованной и децентрализованной форм обработки информации?
11. Опишите принципы технологии автоматизированной обработки информации.
12. Расскажите о режимах взаимодействия пользователя с ЭВМ.
13. Какие Вы знаете внешние запоминающие устройства для хранения информации?

ГЛАВА 5

ТЕХНИЧЕСКИЕ СРЕДСТВА РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ

Вычислительная техника в развитии информатики играет особую роль. Возникновение информатики как научного направления в середине XX столетия связано именно с появлением ЭВМ, ориентированных на хранение и преобразование информации. Появление ЭВМ, их быстрое развитие и внедрение в различные сферы общественной деятельности привело к возникновению нового научно-технического направления, которое называется «вычислительная техника». Специалисты по вычислительной технике разрабатывают новые структуры и принципы работы вычислительных машин, модифицируют и качественно улучшают элементарную базу, на которой строятся компьютеры, создают компьютерные сети и т.п. Работа специалистов по вычислительной технике тесно связана с информатикой и программированием. Многие задачи теоретической информатики являются основополагающими и для специалистов по вычислительной технике, а без усилий программистов, разрабатывающих программное обеспечение ЭВМ, никакая вычислительная машина не сможет работать. Поэтому в раздел прикладной информатики включена тема о вычислительных средствах.

Методические указания

Вычислительные машины являются одним из основных инструментальных средств информатики. Исходя из этого студенты должны знать:
функциональную и структурную организацию ЭВМ;
принципы фон-Неймана;
различие шинной и канальной организации ЭВМ;

последовательность обработки информации в ЭВМ.

Студенты должны понимать:
как построена информационная модель ЭВМ;
какие классы команд управляют работой ЭВМ;
эволюцию развития ЭВМ.

5.1. ОПРЕДЕЛЕНИЕ И ПРИНЦИПЫ ОРГАНИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ В ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВАХ

Обработка информации и представление результатов обработки в удобном для человека виде производится с помощью вычислительных средств. Научно-технический прогресс привел к созданию разнообразных вычислительных средств: электронных вычислительных машин (ЭВМ), вычислительных систем (ВС), вычислительных сетей (ВСТ). Они различаются структурной организацией и функциональными возможностями.

Дать определение такому явлению, как ЭВМ, представляется сложным. Достаточно сказать, что само по себе название ЭВМ, т.е. электронные вычислительные машины, не отражает полностью сущность концепции. Слово «электронные» подразумевало электронные лампы в качестве элементной базы, современные ЭВМ правильнее следовало бы называть микроэлектронными. Слово «вычислительный» подразумевает, что устройство предназначено для проведения вычислений, однако анализ программ показывает, что современные ЭВМ не более 10 — 15% времени тратят на чисто вычислительную работу — сложение, вычитание, умножение и т.д. Основное время затрачивается на выполнение операций пересылки данных, сравнения, ввода-вывода и т.д. То же самое относится и к англоязычному термину «компьютер», т.е. «вычислитель». К понятию ЭВМ можно подходить с нескольких точек зрения.

Представляется разумным определить ЭВМ с точки зрения ее функционирования. Целесообразно описать минимальный набор устройств, который входит в состав любой ЭВМ, и тем самым определить состав минимальной ЭВМ, а также сформулировать принципы работы отдельных блоков ЭВМ и принципы организации ЭВМ как системы, состоящей из взаимосвязанных функциональных блоков.

Если же рассматривать ЭВМ как ядро некоторой информационно-вычислительной системы, может оказаться полезным показать информационную модель ЭВМ — определить ее в виде совокупности блоков переработки информации и множества информационных потоков между этими блоками.

Принципы фон-Неймана. Большинство современных ЭВМ строится на базе принципов, сформулированных американским ученым, одним из

«отцов» кибернетики Дж. фон-Нейманом. Впервые эти принципы были опубликованы фон-Нейманом в 1945 г. в его предложениях по машине EDVAC. Эта ЭВМ была одной из первых машин с хранимой программой, т.е. с программой, запомненной в памяти машины, а не считываемой с перфокарты или другого подобного устройства. В целом эти принципы сводятся к следующему:

1) основными блоками фон-неймановской машины являются блок управления, арифметико-логическое устройство, память и устройство ввода-вывода (рис. 5.1);

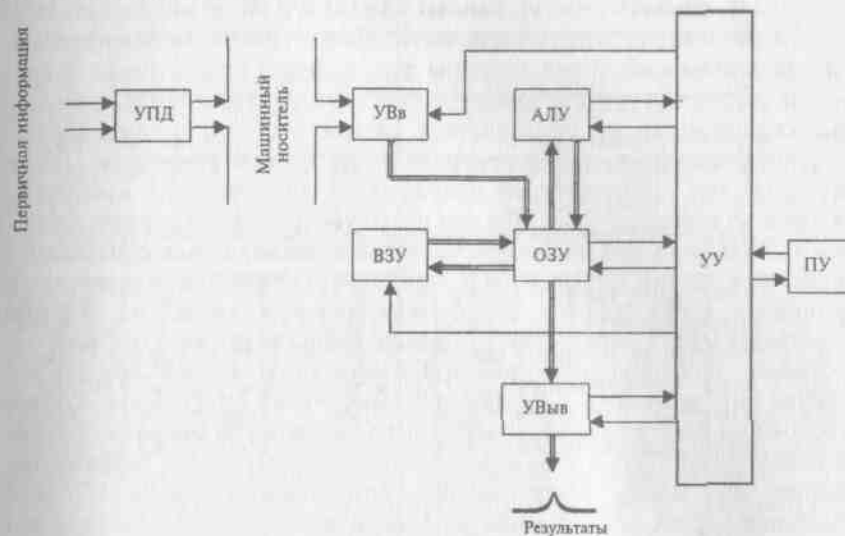


Рис. 5.1. Обобщенная структурная схема ЭВМ: УПД — устройство подготовки данных; УВв — устройство ввода информации; ОЗУ — оперативное запоминающее устройство; ВЗУ — внешнее запоминающее устройство; АЛУ — арифметико-логическое устройство; УУ — устройство управления; ПУ — пульт управления; УВыв — устройство вывода информации

2) информация кодируется в двоичной форме и разделяется на единицы, называемые словами;

3) алгоритм представляется в форме последовательности управляющих слов, которые определяют смысл операции. Эти управляющие слова называются командами. Совокупность команд, представляющая алгоритм, называется программой;

4) программы и данные хранятся в одной и той же памяти. Разнотипные слова различаются по способу использования, но не по способу кодирования;

5) устройство управления и арифметическое устройство обычно объединяются в одно, называемое центральным процессором. Они опреде-

ляют действия, подлежащие выполнению, путем считывания команд из оперативной памяти. Обработка информации, предписанная алгоритмом, сводится к последовательному выполнению команд в порядке, однозначно определяемом программой.

Принципы фон-Неймана практически можно реализовать множеством различных способов. Здесь приведем два из них: ЭВМ с шинной и канальной организацией. Перед тем как описать принципы функционирования ЭВМ, введем несколько определений. *Архитектура ЭВМ* — абстрактное определение машины в терминах основных функциональных модулей, языка, структур данных. Архитектура не определяет особенности реализации аппаратной части ЭВМ, времени выполнения команд, степени параллелизма, ширины шин и других аналогичных характеристик. Архитектура отображает аспекты структуры ЭВМ, которые являются видимыми для пользователя: систему команд, режимы адресации, форматы данных, набор программно-доступных регистров. Одним словом, термин «архитектура» используется для описания возможностей, предоставляемых ЭВМ. Весьма часто употребляется термин *конфигурация ЭВМ*, под которым понимается компоновка вычислительного устройства с четким определением характера, количества, взаимосвязей и основных характеристик его функциональных элементов. Термин «*организация ЭВМ*» определяет, как реализованы возможности ЭВМ.

Команда — совокупность сведений, необходимых процессору для выполнения определенного действия при выполнении программы. Команда состоит из *кода операции*, содержащего указание на операцию, которую необходимо выполнить, и нескольких *адресных полей*, содержащих указание на места расположения операндов команды. Способ вычисления адреса по информации, содержащейся в адресном поле команды, называется *режимом адресации*. Множество команд, реализованных в данной ЭВМ, образует ее *систему команд*.

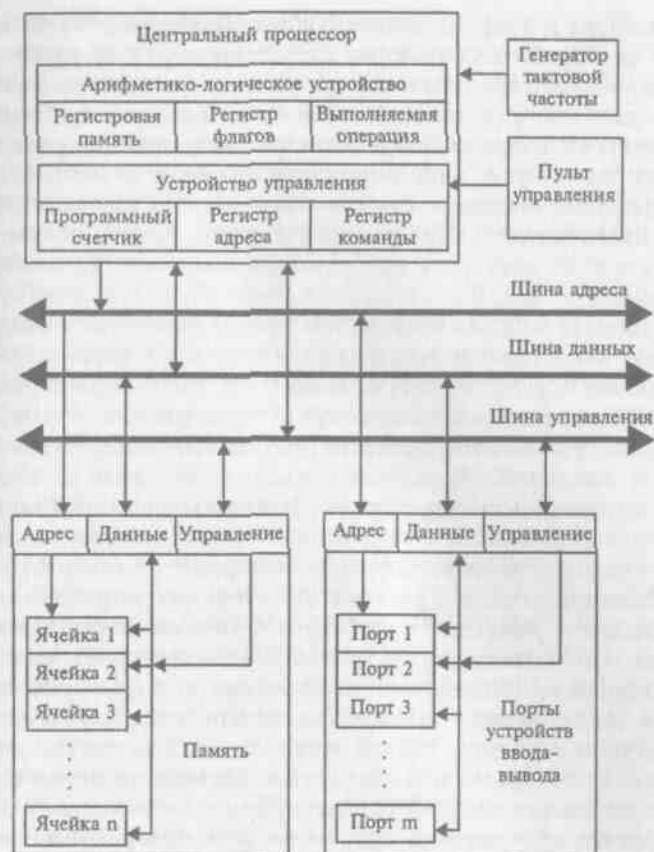


Рис. 5.2. Упрощенная схема ЭВМ с шинной организацией

5.2. ФУНКЦИОНИРОВАНИЕ ЭВМ С ШИННОЙ ОРГАНИЗАЦИЕЙ

Шинная организация является простейшей формой организации ЭВМ. В соответствии с приведенными выше принципами фон-Неймана подобная ЭВМ имеет в своем составе следующие функциональные блоки (рис. 5.2). *Центральный процессор (ЦП)* — функциональная часть ЭВМ, выполняющая основные операции по обработке данных и управлению работой других блоков. Это наиболее сложный компонент ЭВМ как с точки зрения электроники, так и с точки зрения функциональных возможностей. Центральный процессор состоит из следующих взаимосвязанных составных элементов: арифметико-логического устройства, устройства управления и регистров.

Арифметико-логическое устройство (АЛУ) выполняет основную работу по переработке информации, хранимой в оперативной памяти. В нем выполняются арифметические и логические операции. Кроме того, АЛУ вырабатывает управляющие сигналы, позволяющие ЭВМ автоматически выбирать путь вычислительного процесса в зависимости от получаемых результатов. Операции выполняются с помощью электронных схем, каждая из которых состоит из нескольких тысяч элементов. Микросхемы имеют высокую плотность и быстродействие. На современном технологическом уровне все АЛУ можно разместить на одном кристалле полупроводникового элемента размером с конторскую скрепку.

Арифметико-логическое устройство формирует по двум входным переменным одну — выходную, выполняя заданную функцию (сложение,

вычитание, сдвиг и т.д.). Выполняемая функция определяется микрокомандой, получаемой от устройства управления. АЛУ содержит в своем составе устройство, хранящее характеристику результата выполнения операции над данными и называемое *флаговым регистром*. Отметим пока, что отдельные разряды этого регистра указывают на равенство результата операции нулю, знак результата операции (+ или -), правильность выполнения операции (наличие переноса за пределы разрядной сетки или переполнения). Программный анализ флагов позволяет производить операции ветвления программы в зависимости от конкретных значений данных.

Кроме того, в АЛУ имеется набор программно-доступных быстродействующих ячеек памяти, которые называются *регистрами процессора*.

Регистры составляют основу архитектуры процессора. Среди обязательного набора регистров можно отметить следующие. *Регистр данных* — служит для временного хранения промежуточных результатов при выполнении операций. *Регистр-аккумулятор* — регистр временного хранения, который используется в процессе вычислений (например, в нем формируется результат выполнения команды умножения). *Регистр-указатель стека* — используется при операциях со стеком, т.е. такой структурой данных, которая работает по принципу: последним вошел — первым вышел, т.е. последнее записанное в него значение извлекается из него первым. Пока отметим только, что стеки используются для организации подпрограмм. *Индексные, указательные и базовые регистры* используются для хранения и вычисления адресов операндов в памяти. Регистры-счетчики используются для организации циклических участков в программах. *Регистры общего назначения*, имеющиеся во многих ЭВМ, могут использоваться для любых целей. Точное назначение такого регистра определяет программист при написании программы. Они могут использоваться для временного хранения данных, в качестве аккумуляторов, а также в качестве индексных, базовых, указательных регистров. Количество регистров и связей между ними оказывает существенное влияние на сложность и стоимость процессора. Однако, с другой стороны, наличие большого количества регистров с богатым набором возможностей упрощает программирование и повышает гибкость программного обеспечения. Кроме перечисленных регистров в состав АЛУ могут входить *внутренние системные регистры*, не доступные программно и используемые во время внутренних пересылок информации при выполнении команд.

Устройство управления (УУ) — часть центрального процессора. Оно вырабатывает распределенную во времени и пространстве последовательность внутренних и внешних управляющих сигналов, обеспечивающих выборку и выполнение команд. На этапе цикла выборки команды УУ интерпретирует команду, выбранную из программной памяти. На этапе выполнения команды в соответствии с типом реализуе-

мой операции УУ формирует требуемый набор команд низкого уровня для арифметико-логического устройства и других устройств. Эти команды задают последовательность простейших низкоуровневых операций, таких, как пересылка данных, сдвиг данных, установка и анализ признаков, запоминание результатов и др. Такие элементарные низкоуровневые операции называют *микрооперациями*, а команды, формируемые устройством управления, называются *микрокомандами*. Последовательность микрокоманд, соответствующая одной команде, называется *микропрограммой*.

В простейшем случае УУ имеет в своем составе три устройства — *регистр команды*, который содержит код команды во время ее выполнения; *программный счетчик*, в котором содержится адрес очередной подлежащей выполнению команды; *регистр адреса*, в котором вычисляются адреса операндов, находящихся в памяти. Для связи пользователя с ЭВМ предусмотрен *пульт управления*, который позволяет выполнять такие действия, как сброс ЭВМ в начальное состояние, просмотр регистра или ячейки памяти, запись адреса в программный счетчик, пошаговое выполнение программы при ее отладке и т.д.

Память (ПМ) — устройство, предназначенное для запоминания, хранения и выборки программ и данных. Память состоит из конечного числа ячеек, каждая из которых имеет свой уникальный номер или адрес. Доступ к ячейке осуществляется указанием ее адреса. Память способна выполнять два вида операций над данными — чтение с сохранением содержимого и запись нового значения со стиранием предыдущего. Как уже говорилось выше, каждая ячейка памяти может использоваться для хранения либо порции данных, либо команды. В большинстве современных ЭВМ минимально адресуемым элементом памяти является *байт* — поле из 8 бит. Совокупность битов, которые арифметико-логическое устройство может одновременно поместить в регистр или обработать, называют обычно *машинным словом*.

Оперативная память (ОП) — функциональный блок, хранящий информацию для УУ (команды) и АЛУ (данные). Задачи, решаемые с помощью ЭВМ, требуют хранения в памяти различного количества информации, зависящего от сложности реализуемого алгоритма, количества исходных данных и т.п. Поэтому память должна вмещать достаточно большое количество информации, т.е. должна иметь большую емкость. С другой стороны, память должна обладать достаточным быстродействием, соответствующим быстродействию других устройств ЭВМ. Чем больше емкость памяти, тем медленнее к ней доступ, так как время доступа (т.е. быстродействие) определяется временем, необходимым для выборки из памяти или записи в нее информации. Поэтому в ЭВМ существует несколько запоминающих устройств, различающихся емкостью и быстродействием (табл. 5.1).

Таблица 5.1

Устройства памяти	Время доступа, с	Емкость, бит
регистры	$(2-20) \cdot 10^{-9}$	10^3-10^4
оперативная память	$(0,2-20) \cdot 10^{-6}$	10^6-10^8
внешняя память	10-100	$10^{11}-10^{12}$

Оперативная память собирается на ферритовых сердечниках или полупроводниковых микросхемах и состоит из отдельных ячеек.

Периферийные устройства (ПУ). В их число входят устройства двух типов: устройства внешней памяти, предназначенные для долговременного хранения данных большого объема и программ, и коммуникационные устройства, предназначенные для связи ЭВМ с внешним миром (с пользователем, другими ЭВМ и т.д.). Обмен данными с внешним устройством осуществляется через *порты ввода-вывода*. Порт (в переводе с англ. *port* — ворота, дверь, отверстие) — это абстрактное понятие, на самом деле несуществующее. По аналогии с ячейками памяти порты можно рассматривать как ячейки, через которые можно записать в ПУ, или наоборот — прочитать из него. Так же как и ячейки памяти, порты имеют уникальные номера — адреса портов ввода-вывода.

Система шин. Объединение функциональных блоков в ЭВМ осуществляется посредством следующей системы шин: *шины данных*, по которой осуществляется обмен информацией между блоками ЭВМ; *шины адреса*, используемой для передачи адресов (номеров ячеек памяти или портов ввода-вывода, к которым производится обращение); и *шины управления* для передачи управляющих сигналов. Совокупность этих трех шин называют *системной шиной*, *системной магистралью* или *системным интерфейсом*. Состав и назначение шины, правило их использования, виды передаваемых по шине сигналов и другие характеристики шины могут существенно различаться у разных видов ЭВМ. Однако есть принципиально общие закономерности в организации шин. Шина состоит из отдельных проводников (*линий*). Сигналы по линиям шины могут передаваться либо импульсами (наличие импульса соответствует логической 1, а отсутствие импульса — 0), либо уровнем напряжения (например, высокий уровень — логическая 1, низкий — 0). *Шириной шины* называется количество линий (проводников), входящих в состав шины. Ширина шины адреса определяет размер адресного пространства ЭВМ. Если, например, количество линий адреса, используемых для адресации памяти, равно 20, то общее количество адресуемых ячеек памяти составит 2^{20} , т.е. примерно один миллион ячеек (точнее, 1 048 576 ячеек).

Обычно на шине в любой момент можно выделить два активных устройства. Одно из них называется *задатчиком* и инициирует операцию обмена данными (формирует адреса и управляющие сигналы), другое

называется *исполнителем* и выполняет операцию (дешифрирует адреса и управляющие сигналы и принимает или передает данные). В большинстве случаев задатчиком является ЦП. Память всегда выступает только в качестве исполнителя.

Функционирование ЭВМ с шинной структурой можно описать следующим обобщенным алгоритмом (рис. 5.3):

1. Инициализация. После включения ЭВМ или операции сброса в регистры центрального процессора заносятся некоторые начальные значения. Обычно в процессе инициализации в память ЭВМ помещается программа, называемая первичным загрузчиком. Основное назначение первичного загрузчика — загрузить в память с устройства внешней памяти операционную систему. Эта программа может быть размещена в энергонезависимом устройстве памяти или автоматически считываться с некоторого устройства внешней памяти. Мы не будем здесь подробно останавливаться на механизмах загрузки операционной системы, тем более что они могут существенно различаться для разных типов ЭВМ. Пока будем полагать, что в памяти некоторым образом оказалась первая из подлежащих выполнению программ. Программному счетчику присваивается начальное значение, равное адресу первой команды программы, указанной выше.

2. Центральный процессор производит операцию считывания команды из памяти. В качестве адреса ячейки памяти используется содержимое программного счетчика.

3. Содержимое считанной ячейки памяти интерпретируется процессором как команда и помещается в регистр команды. Устройство управления приступает к интерпретации прочитанной команды. По полю кода операции из первого слова команды устройство управления определяет ее длину и, если это необходимо, организует дополнительные операции считывания, пока вся команда полностью не будет прочитана процессором. Вычисленная длина команды прибавляется к исходному содержимому программного счетчика, и когда команда полностью прочитана, программный счетчик будет хранить адрес следующей команды.



Рис. 5.3. Обобщенный алгоритм функционирования фон-неймановской ЭВМ

4. По адресным полям команды устройство управления определяет, имеет ли команда операнды в памяти. Если это так, то на основе указанных в адресных полях режимов адресации вычисляются адреса операндов и производятся операции чтения памяти для считывания операндов.

5. Устройство управления и арифметико-логическое устройство выполняют операцию, указанную в поле кода операции команды. Во флаговом регистре процессора запоминаются признаки результата операции (равно нулю или нет, знак результата, наличие переполнения и т.д.).

6. Если это необходимо, устройство управления выполняет операцию записи для того, чтобы поместить результат выполнения команды в память.

7. Если последняя команда не была командой **ОСТАНОВИТЬ ПРОЦЕССОР**, то описанная последовательность действий повторяется, начиная с шага 1. Описанная последовательность действий центрального процессора с шага 1 до шага 6 называется *циклом процессора*.

Большинство мини- и микроЭВМ имеют шинную организацию и их поведение описывается приведенным выше алгоритмом. В различных конкретных ЭВМ реализация этого алгоритма может несколько отличаться. Так, например, по-разному может осуществляться синхронизация задатчиков и исполнителей, процессор может считывать из памяти не одну команду, а сразу несколько и хранить их в специальной очереди команд. Часто используемые программой команды и данные могут храниться не в основной памяти ЭВМ, а в быстродействующей буферной памяти и т.д. Таким образом, функционирование любой фон-неймановской ЭВМ описывается алгоритмом, близким к приведенному выше, и представляет собой последовательность достаточно простых действий.

5.3. ФУНКЦИОНИРОВАНИЕ ЭВМ С КАНАЛЬНОЙ ОРГАНИЗАЦИЕЙ

В основе этого типа организации ЭВМ лежит множественность каналов связи между устройствами и функциональная специализация узлов. Упрощенная схема организации ЭВМ с каналами приведена на рис. 5.4. Сравним схему ЭВМ с каналами и описанную выше схему ЭВМ с шинной организацией.

Все фон-неймановские ЭВМ очень похожи друг на друга и алгоритм функционирования центрального процессора по сути ничем не отличается от описанного выше.

Помимо уже знакомого набора устройств (центральный процессор, память, устройства ввода-вывода) в состав ЭВМ с каналной организацией входят устройства, называемые каналами. Канал — это специализированный процессор, осуществляющий всю работу по управлению



Рис. 5.4. Упрощенная схема ЭВМ с каналной организацией. ЦП — центральный процессор; ВУ — внешние устройства

контроллерами внешних устройств и обмену данными между основной памятью и внешними устройствами. Устройства группируются по характерной скорости и подключаются к соответствующим каналам. «Быстрые» устройства (например, накопители на магнитных дисках) подсоединяются к селекторным каналам. Такое устройство получает селекторный канал в монопольное использование на все время выполнения операции обмена данными. «Медленные» устройства подключаются к мультиплексным каналам. Мультиплексный канал разделяется (мультиплексируется) между несколькими устройствами, при этом возможен одновременный обмен данными с несколькими устройствами. Доступ к оперативной памяти может получить и центральный процессор, и один из каналов. Для управления очередностью доступа имеется контроллер оперативной памяти. Он определяет приоритетную дисциплину доступа при одновременном обращении нескольких устройств к памяти. Наименьший приоритет имеет центральный процессор. Среди каналов больший приоритет имеют медленные каналы. Таким образом, приоритет обратно пропорционален частоте обращения устройств к памяти.

За счет существенного усложнения организации ЭВМ упрощается архитектура ввода-вывода. Связь между отдельными узлами осуществляется по схеме, напоминающей треугольник. Операции обмена данными становятся более простыми. Канал, по сути, представляет собой специализированный «интеллектуальный» контроллер прямого доступа к памяти. Для ускорения обмена данными реализованы несколько трактов обмена данными (процессор — основная память и каналы — основная память). О своем состоянии канал может информировать процессор с помощью прерываний. Все контроллеры внешних устройств под-

ключаются к «своим» каналам с помощью стандартного интерфейса. Свобода подключения внешних устройств сохраняется благодаря стандартному протоколу интерфейса, при этом появляется возможность группировать устройства по характеристикам.

Результатом введения каналов (специализированных процессоров ввода-вывода) является большая стандартизация и упрощение процессов обмена. С другой стороны, вводятся некоторые ограничения. Например, сохраняется только одна схема, напоминающая схему прямого доступа, с обменом информацией между процессором и каналом по прерываниям.

Канал, являясь хотя и специализированным, но все-таки процессором, выполняет свою *канальную программу*. Она состоит из канальных команд и хранится в оперативной памяти. Длина канальной программы произвольна, последняя команда канальной программы содержит признак конца. Подготовку канальной программы и загрузку ее в оперативную память осуществляет операционная система.

Отметим некоторые особенности канальных машин. Несколько подряд идущих канальных команд могут образовывать цепочку данных. В этом случае имеется одна команда обмена, например чтения физической записи из нескольких адресов оперативной памяти со счетчиков. Одна физическая запись распределяется в несколько адресов оперативной памяти.

В ЭВМ с канальной организацией процессор практически полностью освобождается от рутинной работы по организации ввода-вывода. Управление контроллерами внешних устройств и обмен данными берет на себя канал. Наличие нескольких трактов передачи данных снимает трудности, связанные с блокировкой единственного тракта передачи данных (системной шины), что повышает скорость обмена. Все это дает возможность производить обмен данными с внешними устройствами параллельно с основной вычислительной работой центрального процессора. В результате общая производительность системы существенно возрастает. Удорожание схемы окупается.

Одной из первых машин с каналами была ЭВМ второго поколения IBM-704. Ярким примером ЭВМ с каналами являются машины семейства IBM-360/370. Появление этих ЭВМ произвело переворот в вычислительной технике, и на долгие годы они стали образцом для подражания у создателей ЭВМ. Хотя в настоящее время эти машины ушли в прошлое, они оставили богатое наследие в виде интересных архитектурных решений, программных и алгоритмических разработок. В настоящее время схемы со специализированными процессорами ввода-вывода часто встречаются в ЭВМ различных типов. Несомненно, идея схемы с каналами не умерла, и к ней еще неоднократно будет возвращаться.

5.4. ИНФОРМАЦИОННАЯ МОДЕЛЬ ЭВМ

Обработка чисел, символьной информации, логическая обработка, обработка сигналов — это все частные случаи общего понятия под названием «*обработка информации*». Для ЭВМ характерен признак: информация представляется с помощью двоичных целых чисел. Существует три этапа обработки информации:

- хранение двоичной информации;
- передача от одного хранилища к другому;
- преобразование.

ЭВМ можно представить как совокупность узлов, соединенных каналом связи. Узлы соединяют в себе функции хранения и преобразования. По каналам связи передается информация от узла к узлу. Мы будем говорить о потоках информации в каналах связи (рис. 5.5). Некоторые узлы могут иметь специальную функцию ввода информации в систему и вывода из нее.

Показанная на рис. 5.5 модель не имеет ограничений на связи между отдельными узлами. Реализовать такую систему весьма сложно. Реально существующие системы имеют ряд ограничений на связи и четкое функциональное назначение отдельных узлов. Функции отдельного узла могут зависеть от его состояния. Состояние узла описывается значениями его внутренних полей (регистров), может определяться процессом его функционирования или задаваться извне. Состояние узла будем называть его *режимом*. Физически режим может определяться значением регистра узла. Тогда установить режим узла означает присвоить регистру определенное значение.

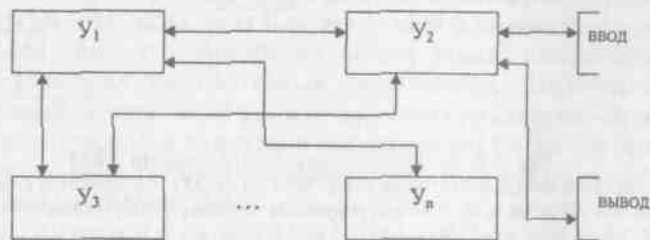


Рис. 5.5. Информационная модель ЭВМ: U — узлы

По каналам связи узлы могут обмениваться либо значащей информацией (сообщениями), либо управляющей. Под *сообщениями* будем понимать последовательности двоичных цифр, сохраняемые или обрабатываемые узлом. Управляющая информация определяет режимы узлов и каналов связи.

Информационная модель позволяет определить основные характеристики ЭВМ:

- Узлы хранения имеют:
 - емкость — максимальную, среднюю или минимальную;
 - скорость выборки;
 - разрядность выборки.
- Преобразующие узлы имеют скорость преобразования.
- Каналы определяются:
 - скоростью передачи информации (пропускная способность);
 - разрядностью передачи.

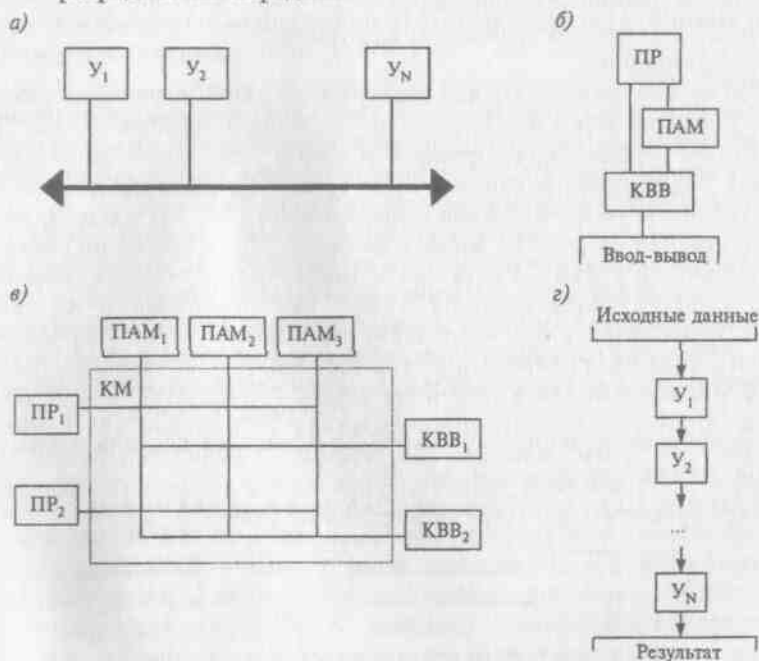


Рис. 5.6. Различные схемы организации ЭВМ:

ЭВМ с шинной организацией (а); канальная ЭВМ (б); ЭВМ с перекрестной коммутацией (в); конвейерная ЭВМ (г); КМ — коммутационная матрица; КВВ — канал ввода-вывода

Из множества возможных соединений отбираются несколько типовых схем, обеспечивающих простоту, возможность реконфигурации (расширения), надежность, стандартизацию и т.д. Можно отметить следующие схемы (рис. 5.6):

- с шинной организацией;
- специализированные процессоры (каналы);
- схемы с коммутацией;
- архитектуры с распределенными функциями (распределенный интеллект);
- с конвейерной организацией.

Рассмотрим некоторые частные модели ЭВМ, имеющие широкое распространение или представляющие теоретический интерес.

Шинная организация. В этой схеме все устройства симметрично подсоединяются к одному каналу, называемому общей шиной. Симметрия подключения гарантирует свободное подключение новых устройств, т.е. система имеет теоретически неограниченное развитие. Некоторые узлы могут иметь специфические свойства, например процессор, оперативная память, внешние накопители данных. Между ними организуется обмен информацией. Так как потоки информации ограничены возможностями одного канала, эта схема имеет принципиальные ограничения скорости работ.

Канальная организация. В этой схеме операции обмена данными с внешними устройствами организуются через специализированный узел — канал ввода-вывода. Благодаря этому можно организовывать обработку информации параллельно с вводом-выводом.

Организация с перекрестной коммутацией. Идея структурной организации таких ЭВМ заключается в том, что все связи между узлами осуществляются с помощью специального устройства — коммутационной матрицы. Коммутационная матрица может связывать между собой любую пару узлов, причем таких пар может быть сколько угодно — связи не зависят друг от друга. В такой схеме нет конфликтов из-за связей, есть конфликты только из-за ресурсов. Возможность одновременной связи нескольких пар устройств позволяет достичь очень высокой производительности комплекса.

Архитектура с распределенными функциями. Это являлось основной идеей японского проекта ЭВМ пятого поколения. В настоящее время эта идея осталась не реализованной. Суть идеи заключается в том, что обработка информации распределяется по «интеллектуальным» периферийным устройствам. Переход от ЭВМ четвертого поколения к ЭВМ пятого поколения намечалось осуществить не за счет существенного изменения элементной базы (как было ранее), а за счет резкого качественного изменения сложности и интеллектуальности различных компонент ЭВМ.

Конвейерная организация. Здесь обрабатываемое устройство разделяется на последовательно включенные операционные блоки, каждый из которых специализирован на выполнение строго определенной части операции. При этом работа осуществляется следующим образом: когда i -й операционный блок выполняет i -ую часть j -й операции, $(i-1)$ -й операционный блок выполняет $(i-1)$ -ую часть $(j+1)$ -й операции, а $(i+1)$ -й операционный блок выполняет $(i+1)$ -ую часть $(j-1)$ -й операции. В результате образуется своего рода конвейер обработки и за счет этого повышается производительность системы.

5.5. ОСНОВНЫЕ КОМАНДЫ ЭВМ

Большое изумление у человека, не знакомого с вычислительной техникой вызывает тот факт, что все разнообразие решаемых на ЭВМ задач реализуется с помощью небольшого набора очень простых команд. Система команд у типичной ЭВМ включает в себя всего 60—150 базовых команд. Все команды в основном служат для выполнения очень простых действий, таких, как прочитать, запомнить, сложить, сдвинуть, сравнить и т.д. Интеллектуальность ЭВМ достигается за счет того, что ЭВМ способна выполнять программы, состоящие из большого числа таких простых действий с огромной, не достижимой для человека скоростью. В данном разделе кратко рассмотрим набор команд, используемых в типичных ЭВМ, и действия, реализуемые этими командами.

При описании системы команд ЭВМ обычно принято классифицировать команды по функциональному назначению, длине, способу адресации и другим признакам. Классификации команд по различным признакам показаны на рис. 5.7. Рассмотрим основные группы команд, придерживаясь классификации команд на группы по функциональному признаку.

Команды передачи данных. Данная группа команд включает в себя подгруппы команд передачи кодов между регистрами внутри процессора, из регистров процессора в память, из памяти в регистры процессора, из одних ячеек памяти в другие и передачи данных между процессором и портами внешних устройств. Отдельную подгруппу составляют команды работы со стеком. Они позволяют включать данные в стек для временного хранения и извлекать данные из стека при необходимости их использования. Подробное назначение и принципы использования стеков в ЭВМ рассмотрим ниже при обсуждении работы с подпрограммами и прерываниями.

Команды обработки данных. Данную группу команд с точки зрения выполняемых над данными операций можно подразделить на арифметические (сложить, вычесть, умножить и т.д.), логические (операции И, ИЛИ, НЕ и т.д.) и команды сдвига. Команды этого типа могут иметь один или два операнда. Операнды могут храниться в регистрах центрального процессора, в памяти или в самой команде. Результат операции формируется в регистре-приемнике или в специализированном регистре-аккумуляторе. Команды данной группы формируют признаки результатов, устанавливаемые в регистре флагов процессора: перенос из старшего разряда, переполнение, нулевой результат и др. К арифметическим командам относят также и команды сравнения. Обычно для сравнения двух чисел процессор выполняет операцию вычитания. По результату вычитания устанавливаются флаги во флаговом регистре процессора. Очевидно, что если сравниваемые величины равны, результат вычитания будет нулевым и во флаговом регистре установится флаг



Рис. 5.7. Классификация команд ЭВМ

нулевого результата. Если первая из сравниваемых величин больше — результат вычитания будет отрицательным и установится флаг отрицательного результата и т.д. Результат вычитания не сохраняется в памяти, по состоянию флагового регистра можно судить о результатах сравнения двух величин. Многие процессоры имеют команды сравнения операнда с нулем. В некоторых процессорах имеются команды проверки или установки состояния отдельных битов в операнде.

Команды передачи управления. Они имеют важное значение, так как используются для изменения естественного порядка следования команд и организации циклических участков в программах.

Простейшей командой передачи управления является команда безусловного перехода $JMP <адрес>$, которая загружает адрес перехода, указанный в команде, в программный счетчик. Команды условного перехода проверяют указанное в команде условие и модифицируют программный счетчик, если условие истинно. Обычно команды условного перехода используются после команд, изменяющих состояние флагового регистра (например, команд сравнения). При проверке условия производится сравнение состояния одного или нескольких флагов из флагового регистра с комбинацией, указанной в коде команды условного перехода. Модификация программного счетчика может производиться либо за-

грузкой в него нового значения, либо сложением его со смещением, указанным в команде.

Команды для работы с подпрограммами. Стеки. В практике программирования широко используется такой прием, как организация подпрограмм. Подпрограмма описывается один раз, а использоваться (вызываться) может из различных мест программы. При этом, после того как подпрограмма закончила свою работу, управление должно быть передано туда, откуда подпрограмма была вызвана на команду, следующую в памяти сразу за командой обращения к подпрограмме. Адрес команды, на которую управление передается после окончания работы подпрограмм, называется *адресом возврата*. Очевидно, для того, чтобы начать выполнять подпрограмму, в программный счетчик необходимо загрузить адрес первой команды подпрограммы. Для осуществления возврата из подпрограммы необходимо запомнить в каком-то месте адрес возврата. Можно, например, сохранить адрес возврата в одном из регистров процессора. Такой способ сохранения адреса возврата очень прост и легко реализуется. Однако он обладает одним существенным недостатком. Достаточно часто встречаются подпрограммы, которые вызывают другие подпрограммы. Пусть основная программа вызвала подпрограмму А. Она в свою очередь обратилась к подпрограмме В. Если адрес возврата для подпрограммы А хранится в регистре процессора, то куда девать адрес возврата при вызове подпрограммы В?

Для организации подпрограмм большинство ЭВМ используют аппаратно поддерживаемую структуру данных, называемую стеком. *Стек* — это структура данных, организованная по принципу: последним вошел — первым вышел, т.е. последние записанные в стек данные извлекаются из него первыми. В переводе с англ. *stack* — стопка. Аналогом стека может служить стопка тарелок. Положить тарелку в стопку можно только сверху, извлечь опять-таки только верхнюю тарелку. В ЭВМ для организации стека выделяется область оперативной памяти, а для ее адресации и доступа к стеку используется упоминавшийся выше регистр — указатель стека. *Указатель стека* хранит адрес ячейки памяти, содержащей последнее помещенное в стек значение.

Для возврата из подпрограммы в основную программу служат *команды возврата* RETURN. Команды возврата из подпрограммы извлекают из стека верхний элемент и помещают его в программный счетчик. Если имели место несколько вложенных вызовов подпрограмм, то возврат произойдет по адресу возврата, сохраненному после последнего вызова, так как для хранения адресов возврата используется стек и последний сохраненный адрес возврата будет использован первым.

Прочие команды. В ЭВМ могут быть дополнительные (специальные) команды. К их числу можно отнести команды остановки центрального процессора, сброса внешних устройств, установки или сброса отдельных признаков и т.д.

5.6. ПОКОЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ СРЕДСТВ

Первые проекты электронных вычислительных машин (ЭВМ) появились в конце 30-х — начале 40-х годов XX в. Технические предпосылки для этого уже были созданы, развивалась электроника и счетно-аналитическая вычислительная техника. В 1904 г. был изобретен первый ламповый диод, а в 1906 г. — первый триод (соответственно двух- и трехэлектродная электронная лампа); в 1918 г. — электронное реле (ламповый триггер). Триггерные схемы стали широко применяться в электронике для переключения и релейной коммутации.

Другой технической предпосылкой создания ЭВМ стало развитие электромеханической счетно-аналитической техники. Благодаря накопленному опыту в области развития вычислительной техники в середине 30-х годов стало возможным создание программно-управляемых вычислительных машин, а построение ВМ на электронных схемах открывало широкие перспективы, связанные с увеличением надежности и быстродействия.

ЭВМ появились, когда возникла острая необходимость в проведении трудоемких и точных расчетов. Уровень прогресса в таких областях науки и техники, как, например, атомная энергетика, аэрокосмические исследования, во многом зависел от возможности выполнения сложных расчетов, которые нельзя было осуществить в рамках электромеханических счетных машин. Требовался переход к вычислительным машинам, работающим с большей производительностью.

В истории развития ЭВМ выделяют пять этапов, соответствующих пяти поколениям ЭВМ.

Период машин первого поколения начинается с переходом к серийному производству ЭВМ в начале 50-х годов XX в. В них были реализованы основные принципы, предложенные Джоном фон-Нейманом (см. § 6.1 — 6.2).

1. *Принцип хранимой программы.* Машина имеет память, в которой хранятся программа, данные и результаты промежуточных вычислений. Программа вводится в машину, так же как и данные, в виде двоичных кодов (а не штекерным методом, т.е. коммутацией проводов в определенной последовательности).

2. *Адресный принцип.* В команде указываются не сами числа, над которыми нужно выполнять арифметические действия, а адреса ячеек памяти, где эти числа находятся.

3. *Автоматизм.* После ввода программы и данных машина работает автоматически, выполняя предписания программы без вмешательства человека. Для этого машина запоминает адрес выполняемой команды, а каждая команда содержит указание об адресе следующей команды. Указание может быть одним из трех типов: неявным (перейти к команде, следующей по адресу за выполняемой), безусловным (перейти к команде по задан-

ному адресу), условным (проверить заданное условие и в зависимости от его выполнения перейти к команде по тому или иному адресу).

4. *Переадресация.* Адреса ячеек памяти, указанные в команде, можно вычислять и преобразовывать как числа.

Структура ЭВМ, в которой реализованы принципы фон-Неймана, впоследствии получила название структуры «фон-Неймана» (или классической). Все дальнейшее развитие ЭВМ шло двумя путями: совершенствование структуры фон-Неймана и поиск новых структур.

Технической основой элементной базы процессоров первых ЭВМ были электронные вакуумные лампы (ЭВЛ), а в качестве оперативных запоминающих устройств использовались электронно-лучевые трубки (ЭЛТ). Это были громоздкие по габаритам машины, занимающие много места и потребляющие много электроэнергии. Они делали несколько тысяч операций в секунду и обладали памятью в несколько тысяч машинных слов. Эти машины предполагали монопольный режим использования, т.е. в распоряжении пользователя были все ресурсы машины и ее управление. Программист писал свою программу в машинных кодах и отлаживал ее за пультом машины, которая на время отладки была полностью в его распоряжении. При этом 90% времени машина простаивала в ожидании команд, т.е. использование машинных ресурсов было малоэффективным из-за отсутствия развитой операционной системы. Использовались ЭВМ первого поколения в основном для научных расчетов. Первой отечественной ЭВМ была МЭСМ (малая электронная счетная машина), разработанная в 1947 — 1951 гг. под руководством акад. С.А. Лебедева. В 1952 г. была введена в эксплуатацию БЭСМ (большая электронная счетная машина), созданная под руководством С.А. Лебедева. В 1955 г. начался выпуск малой ЭВМ «Урал-1» (руководитель проекта Б.И. Рамеев). Примером зарубежной серийной модели ЭВМ является IBM-701 (США).

Второе поколение ЭВМ (конец 50-х — середина 60-х годов) называют транзисторно-ферритовым, так как транзисторы (твердые диоды и триоды) заменили электронные лампы в процессорах, а ферритовые (намагничиваемые) сердечники — электронно-лучевые трубки в оперативных запоминающих устройствах.

Применение транзисторов существенно повлияло на характеристики и структуру машин. Транзисторные схемы позволили повысить плотность монтажа электронной аппаратуры на порядок и существенно (на несколько порядков) снизить потребляемую электроэнергию. Срок службы транзисторов на два-три порядка превосходил срок службы электронных ламп. Скорость ЭВМ возросла до сотен тысяч операций в секунду, а память — до десятков тысяч машинных слов.

Создание долговременной памяти на магнитных дисках и лентах, а также возможность подключения к ЭВМ изменяемого состава внешних

устройств существенно расширили функциональные возможности вычислительных машин.

В организации вычислительного процесса крупным достижением было совмещение во времени вычислений и ввода-вывода информации, переход от монопольного режима использования ресурсов машины к пакетной обработке. Задания для ЭВМ (на перфокартах, магнитных лентах или дисках) собирались в пакет, который обрабатывался без перерыва между заданиями. Это позволило более экономно использовать ресурсы машины.

В программировании были разработаны методы программирования в символических обозначениях, созданы первые алгоритмические языки и трансляторы с этих языков, созданы библиотеки стандартных программ.

Наиболее широкое применение нашли отечественные ЭВМ, такие, как БЭСМ-4, М-220, «Минск-32». Типичным представителем зарубежной ЭВМ второго поколения является IBM-7090.

Третье поколение ЭВМ (конец 60-х — начало 70-х годов) характеризуется появлением в качестве элементной базы процессора интегральных полупроводниковых схем (вместо отдельных транзисторов), что привело к дальнейшему увеличению скорости до миллиона операций в секунду и памяти до сотен тысяч слов.

ЭВМ третьего поколения также характеризуются крупнейшими сдвигами в архитектуре ЭВМ, их программном обеспечении, организации взаимодействия человека с машиной. Это прежде всего наличие развитой конфигурации внешних устройств (алфавитно-цифровые терминалы, графопостроители и т.п.) с использованием стандартных средств сопряжения, развитая операционная система, обеспечивающая работу в мультипрограммном режиме (несколько одновременно размещаемых в оперативной памяти программ совместно используют ресурсы процессора). Метод использования ресурсов ЭВМ — режим разделения времени совместно с пакетной обработкой. Высокое быстродействие позволяет время обслуживания пользователей разбить на кванты, обрабатывая в течение кванта задание каждого, возвращаясь к пользователю за такое малое время, что у него за дисплеем создается иллюзия, что он один пользуется ресурсами машины.

Решающее значение в развитии вычислительной техники во всем мире сыграло создание семейства вычислительных машин на интегральных схемах с широким диапазоном вычислительной мощности и совместимых снизу вверх на уровне машинных языков, внешних устройств, модулей конструкции и системы элементов. Программная совместимость снизу вверх машин одного и того же семейства предполагает, что любая программа, выполнявшаяся на младшей машине, должна без всяких переделок выполняться на старшей.

Широкое распространение получили также семейства мини-ЭВМ. Сущность их конструкторского решения состояла в такой минимизации

аппаратуры центрального процессора, которая позволяла на уровне технологии того времени создать универсальные ЭВМ, способные осуществлять управление в реальном масштабе времени, при котором темп выдачи управляющих воздействий на объект управления согласован со скоростью протекания процессов в этом объекте.

В нашей стране в период машин третьего поколения была создана Единая Система ЭВМ (ЕС ЭВМ), в основных чертах копирующая IBM-360 и IBM-370, а также серия мини-ЭВМ СМ ЭВМ, ориентированная на зарубежные модели. Вклад отечественной науки в мировое развитие электронной вычислительной техники в этот период связан с промышленным внедрением многопроцессорной ЭВМ М-10.

В период машин третьего поколения произошел крупный сдвиг в области применения ЭВМ. Если раньше ЭВМ использовались в основном для научно-технических расчетов, то в 60 — 70-е годы первое место стала занимать обработка символьной информации, в основном экономической.

Машины серии ЕС ЭВМ имеют универсальное назначение, а основной областью применения СМ ЭВМ является автоматизация технологических процессов, научных экспериментов и испытательных установок, проектно-конструкторских работ.

Переход к машинам четвертого поколения — ЭВМ на больших интегральных схемах (БИС) — происходил во второй половине 70-х годов и завершился приблизительно к 1980 г. Теперь на одном кристалле размером 1 см² стали размещаться сотни тысяч электронных элементов. Скорость и объем памяти возросли в десятки тысяч раз по сравнению с машинами первого поколения и составили примерно 10⁹ оп/с и 10⁷ слов соответственно.

Характерными особенностями машин четвертого поколения являются тесная связь аппаратурной и программной реализаций в структуре машины, отход от принципа минимизации аппаратуры и поручение ей функций программы, что стало возможным благодаря относительно низкой стоимости БИС.

Развитие архитектуры ЭВМ в период машин четвертого поколения привело к появлению структур, в которых вычислительный процесс может протекать по нескольким ветвям параллельно, что приводит к увеличению производительности вычислительных машин. Идея параллелизма была технически реализована в многопроцессорных системах, состоящих из двух или более взаимосвязанных процессоров, работающих с общей памятью и управляемых общей операционной системой.

В результате возросшего быстродействия ЭВМ стало возможным расширить оперативную память за счет введения виртуальной памяти, основанной на страничном обмене информации между внешней и основной памятью.

Наиболее крупным достижением, связанным с применением БИС, стало создание микропроцессоров, а затем на их основе микроЭВМ. Ес-

ли прежние поколения ЭВМ требовали для своего расположения специальных помещений, системы вентиляции, специального оборудования для электропитания, то требования, предъявляемые к эксплуатации микроЭВМ, ничем не отличаются от условий эксплуатации бытовых электроприборов. При этом они имеют достаточно высокую производительность, экономичны в эксплуатации и дешевы. МикроЭВМ используются в измерительных комплексах, системах числового программного управления, в управляющих системах различного назначения.

Дальнейшее развитие микроЭВМ привело к созданию персональных компьютеров (ПК), широкое распространение которых началось с 1975 г., когда фирма IBM выпустила свой первый персональный компьютер IBM PC. Сейчас такие компьютеры (совместимые с IBM PC) составляют около 90% всех производимых в мире ПК. В ПК реализован принцип открытой архитектуры, который означает, что по мере улучшения характеристик основных блоков ПК возможна легкая замена устаревших частей, а модернизированный блок будет совместим с ранее используемым оборудованием. Другими преимуществами ПК являются развитые средства диалога, высокая надежность, удобство эксплуатации, наличие программного обеспечения, охватывающего практически все сферы человеческой деятельности.

В период машин четвертого поколения стали также серийно производиться и суперЭВМ. Рост степени интеграции БИС стал технологической основой производительности ЭВМ. В нескольких серийных моделях была достигнута производительность свыше 1 млрд. операций в секунду. К числу наиболее значительных разработок машин четвертого поколения относится ЭВМ «Крей-3», спроектированная на основе принципиально новой технологии — замены кремниевого кристалла арсенидом галлия, имеющая производительность до 16 млрд. операций в секунду. Примером отечественной суперЭВМ является многопроцессорный вычислительный комплекс «Эльбрус» с быстродействием до 1,2·10⁸ оп/с.

С конца 80-х годов в истории развития вычислительной техники наступила пора пятого поколения ЭВМ. Технологические, конструкторские, структурные и архитектурные идеи машин пятого поколения принципиально отличаются от машин предшествующих поколений. Прежде всего их структура и архитектура отличаются от фон-неймановской (классической). Высокая скорость выполнения арифметических вычислений дополняется высокими скоростями логического вывода. Даже скорость предполагается выражать в единицах логического вывода. Машина состоит из нескольких блоков. Блок общения обеспечивает интерфейс между пользователем и ЭВМ на естественном языке и дисциплина программирования как наука для пользователя перестает в будущем быть актуальной. Важное место в структуре ЭВМ занимает блок, представляющий базу знаний, в котором хранятся знания, накопленные человечеством в различных предметных областях, которые постоянно

расширяются и пополняются. Следующий блок, называемый решателем, организует подготовку программы решения задачи на основании знаний, получаемых из базы знаний и исходных данных, полученных из блока общения. Ядро вычислительной системы составляет ЭВМ высокой производительности.

В связи с появлением новой базовой структуры ЭВМ в машинах пятого поколения широко используются модели и средства, разработанные в области искусственного интеллекта

Контрольные вопросы

1. Дайте определение ЭВМ.
2. Сформулируйте принципы построения ЭВМ фон-Неймана.
3. Что такое архитектура и конфигурация ЭВМ?
4. Раскройте шинную организацию ЭВМ.
5. Назовите состав и функции блоков центрального процессора ЭВМ.
6. Для чего предназначены регистры?
7. Какие функции выполняет устройство управления ЭВМ?
8. Расскажите о составе устройств памяти ЭВМ.
9. Объясните систему шин ЭВМ.
10. Расскажите последовательность работы элементов ЭВМ при обработке информации.
11. Из каких элементов состоит ЭВМ с канальной организацией?
12. В чем особенности информационной модели ЭВМ?
13. Какие Вы знаете классы команд ЭВМ?
14. По каким характеристикам осуществляется классификация ЭВМ по поколениям?
15. Дайте характеристику ЭВМ четвертого поколения.
16. Расскажите об особенностях ЭВМ пятого поколения.

ГЛАВА 6

ПЕРСОНАЛЬНЫЕ ЭВМ

С 1975 г. в США было начато серийное производство персонального компьютера (ПК), или персональной ЭВМ (ПЭВМ). Это событие сегодня в мире часто называют второй информационной революцией (первой информационной революцией считается появление печатного станка и книгопечатания — 1445 г.). ПК появился на базе мини- и микроЭВМ и обеспечивает персональные вычисления, т.е. режим работы специалиста в данной предметной области непосредственно с ПК на своем рабочем месте. За дисплей ЭВМ может пользоваться непрофессионал в программировании. С 1981 г. стали выпускаться ПЭВМ, имеющие блочно-модульную конструкцию. Эти машины, простые в эксплуатации и сравнительно дешевые, предназначались для потребителей, не обладающих знаниями в области вычислительной техники и программирования. Широкое распространение мини-ЭВМ в начале 70-х годов определялось необходимостью приблизить компьютер к пользователю. Мини-ЭВМ устанавливались непосредственно на предприятиях и в организациях, где использование больших ЭВМ было экономически невыгодным.

Методические указания

Материал шестой главы составляет основу применения предыдущих знаний к решению практических задач на ЭВМ. Студенты должны усвоить, что ЭВМ наряду с программными средствами являются инструментом информатики.

В результате изучения этой главы студенты должны знать:

- принципы строения ПК;
- назначение основных устройств ПК;
- правила техники безопасности при работе на ПК;
- клавиатуру ПК;
- общие принципы размещения информации в электронной таблице и основные способы получения результатов с ее использованием;

важнейшие операции с текстовым и графическим редакторами.

Студенты должны уметь:

пользоваться клавиатурой ПК;

набирать текст на экране;

использовать простейшие приемы редактирования текста;

пользоваться графическим редактором при построении простейших изображений;

размещать информацию в электронной таблице;

решать простейшие вычислительные задачи с помощью электронных таблиц;

пользоваться внешними устройствами хранения информации и печатающими устройствами.

Студенты должны иметь представление о:

классификации ПК;

операционной системе ПК;

интерфейсе пользователя;

перспективах развития ПК.

6.1. КРАТКАЯ ХАРАКТЕРИСТИКА И КЛАССИФИКАЦИЯ ПК

Так как технической основой ПК служит микропроцессор (МП), то поэтому развитие технологии МП определило смену поколений ПЭВМ:

8-разрядный МП (1975 — 1980 гг.) — первое поколение;

16-разрядный МП (1981 — 1985 гг.) — второе поколение;

32-разрядный МП (1986 — 1992 гг.) — третье поколение;

64-разрядный МП (1993 г. — по настоящее время) — четвертое поколение ПЭВМ.

Важное значение в развитии ПЭВМ сыграло появление компьютера IBM PC, произведенного корпорацией IBM (США) на базе МП intel-8086 в 1981 г. Это персональный компьютер занял ведущее место на рынке ПЭВМ. Его основное преимущество — так называемая «открытая архитектура», благодаря которой пользователи могут расширять возможности приобретенной ПЭВМ, добавляя различные периферийные устройства и модернизируя компьютер. В дальнейшем другие фирмы начали создавать свои ПЭВМ, но компьютер IBM PC стал как бы стандартом класса ПЭВМ. В наши дни около 85% всех продаваемых ПЭВМ базируется на архитектуре IBM PC.

ПЭВМ по назначению классифицируются на: бытовые, общего назначения и профессиональные.

Бытовые ПЭВМ предназначены для массового потребителя, поэтому они должны быть достаточно дешевыми, надежными и иметь, как правило, простейшую базовую конфигурацию. Бытовые ПЭВМ используются в домашних условиях для развлечений (видеоигры), для обучения и тренировки, управления бытовой техникой. Однако архитектура этих

машин позволяет подключать их к каналам связи, расширять набор периферийного оборудования. При некоторой модернизации эти модели могут использоваться для индивидуальной обработки текста, решения небольших научных и инженерных задач (например, отечественная ПЭВМ «Амата»). Бытовые ПЭВМ снабжаются пакетом игр, программным обеспечением локальной сети и др. Фирмы предлагают за дополнительную плату нарастить комплектность компьютера НЖМД типа «винчестер», музыкальной картой, монитором и т.д. Модель «Амата» легко превращается в ПЭВМ общего назначения.

Персональные ЭВМ общего назначения применяются для решения задач научно-технического и экономического характера, а также для обучения и тренировки. Они размещаются на рабочих местах потребителей: на предприятиях, в учреждениях, в магазинах, на складах и т.п.

Машины этого класса обладают достаточно большой емкостью оперативной памяти, имеют внешнюю память на гибких и жестких магнитных дисках, собственный дисплей. Интерфейсы позволяют подключать большое количество периферийных устройств, средства для работы в составе вычислительных сетей.

ПЭВМ общего назначения используются прежде всего потребителями-непрофессионалами. Поэтому они снабжаются развитым программным обеспечением, включающим операционные системы, трансляторы с алгоритмических языков, пакеты прикладных программ. В состав аппаратуры входят устройства для вывода как текстового, так и графического материала, принтеры с высоким качеством печати. Этот класс ПЭВМ получил наибольшее распространение на мировом рынке.

Профессиональные ПЭВМ используются в научной сфере, для решения сложных информационных и производственных задач, где требуются высокое быстродействие, эффективная передача больших массивов информации, достаточно большая емкость оперативной памяти. Потребителями профессиональных ПЭВМ, как правило, являются профессионалы-программисты, поэтому программное обеспечение должно быть достаточно богатым, гибким, включать инструментальные программные средства.

Благодаря подключению широкой номенклатуры периферийных устройств функциональные возможности ПЭВМ значительно расширяются. Они могут работать в многозадачном режиме, с алгоритмическими языками высокого уровня, в составе вычислительных сетей. По своим функциональным возможностям многопроцессорные профессиональные ПЭВМ не только приближаются, но и вполне могут конкурировать с большими ЭВМ предыдущего поколения.

В настоящее время появился новый признак классификации ПЭВМ по конструктивному исполнению, связанному с микроминиатюризацией изделий. Снижение веса и уменьшение габаритов привело к выпуску компьютеров, называемых LAPTOP («нако-

ленные» компьютеры), NOTEBOOK (компьютеры-блокноты) и HANDHELD (ручной компьютер).

В LAPTOP-компьютере клавиатура и системный блок выполнены в одном корпусе, который сверху, как крышкой, закрывается жидкокристаллическим дисплеем, неразъемно соединенным со своим электронным основанием. Соединительные провода между дисплеем и ЭВМ скрыты в корпусе. Компьютер можно легко переносить и держать на коленях пользователя. Эти модели немного уступают по своим техническим параметрам настольным ПЭВМ. Они построены на МП 180386, имеют встроенные НГМД и НЖМД. В большинстве моделей LAPTOP используются монокромные дисплеи, так как применение цветных дисплеев приводит к резкому удорожанию компьютера. Компьютеры класса LAPTOP не должны весить более 3,5 кг.

NOTEBOOK (компьютеры-блокноты) имеют размеры одного листа бумаги стандарта А4 (297×210), обладают неполной клавиатурой (около 80 клавиш). В них используются НЖМД (например, дисковод емкостью 120 Мбайт, диаметром 2,5 дюйма) и НГМД. В комплекте с NOTEBOOK можно применять модем или факс-модем, выполненные в виде отдельного настольного блока, присоединенного кабелем к компьютеру и телефонной сети. Однако существуют блоки модемов и факс-модемов, вставляемые в корпус NOTEBOOK и работающие, как правило, только на передачу сообщений. Компьютеры NOTEBOOK могут использоваться в деловых поездках, не требуют места на рабочем столе, могут храниться в ящике для бумаг, в портфеле.

ПЭВМ HANDHELD — ПЭВМ, размер которой меньше одного листа бумаги стандарта А4 (например, модель Hewlett Packard 95 LX имеет размеры 160×86×25 мм), поэтому они всегда под рукой (в кармане) в готовом к работе состоянии. Эти модели могут работать независимо от электросети. Программы при автономной работе вводятся с помощью твердой карточки (ROM CARD), на которых записаны программы емкостью 32, 64 или 128 Кбайт. Карточки можно перепрограммировать. Для хранения результатов расчетов, введенного текста, составленных электронных таблиц и других результатов работы пользователь применяет ROM CARD со встроенной в них батарейкой. Это карточки очень небольшого размера (2×5мм), что позволяет вставлять их в специальные отверстия в корпусе персонального компьютера для чтения с них программ, данных или записи результатов работы пользователя. По мере надобности результаты работы могут быть по кабелю перенесены на настольный компьютер. В конструкциях этих моделей ПЭВМ предусматривается гораздо больший объем постоянной памяти, чем в конструкциях настольных ПЭВМ.

По прогнозам специалистов, миниатюрные компьютеры в ближайшем будущем смогут включаться в вычислительные сети без проводов (с помощью радио), что потребует минимальных затрат. Такая технология

получила название «полевая компьютеризация» (Field Computing). Ее внедрение вызовет новый революционный скачок в информатизации общества.

6.2. ХАРАКТЕРИСТИКА ОСНОВНЫХ БЛОКОВ ПК

ПЭВМ включает три основных устройства: системный блок, клавиатуру и дисплей (монитор). Однако для расширения функциональных возможностей ПЭВМ можно подключить различные дополнительные периферийные устройства, в частности: печатающие устройства (принтеры), накопители на магнитной ленте (стримеры), различные манипуляторы (мышь, джойстик, трекбол, световое перо), устройства оптического считывания изображений (сканеры), графопостроители (плоттеры) и др.

Эти устройства подсоединяются к системному блоку с помощью кабелей через специальные гнезда (разъемы), которые размещаются обычно на задней стенке системного блока.

В некоторых моделях ПЭВМ при наличии свободных гнезд дополнительные устройства вставляются непосредственно в системный блок, например, модем для обмена информацией с другими ПЭВМ через телефонную связь или стример для хранения больших массивов информации на МЛ. ПЭВМ, как правило, имеет модульную структуру (рис. 6.1). Все модули связаны с системной магистралью (шиной).

Системный блок. Главный блок ПЭВМ, он включает в свой состав центральный микропроцессор, сопроцессор, модули оперативной и постоянной памяти, контроллеры, накопители на магнитных дисках и другие функциональные модули. Набор модулей определяется типом ПЭВМ. Пользователи по своему желанию могут изменять конфигурацию ПЭВМ, подключая дополнительные периферийные устройства.

В системный блок может быть встроено звуковое устройство, с помощью которого пользователю удобно следить за работой машины, вовремя обращать внимание на возникшие сбои в отдельных устройствах или на возникновение необычной ситуации при решении задачи на ПЭВМ. Со звуковым устройством часто связан таймер, позволяющий вести отсчет времени работы машины, фиксировать календарное время, указывать на окончание заданного промежутка времени при выполнении той или иной задачи.

Микропроцессор. Центральный микропроцессор является ядром любой ПЭВМ. Он выполняет функции обработки информации и управления работой всех блоков ПЭВМ.

Конструктивно МП, как правило, выполнен на одном кристалле (на одной СБИС). В состав МП входят:

арифметико-логическое устройство,

центральное устройство управления,
внутренняя регистровая память,
КЭШ-память,
схема обращения к оперативной памяти,
схемы управления системной шиной и др.

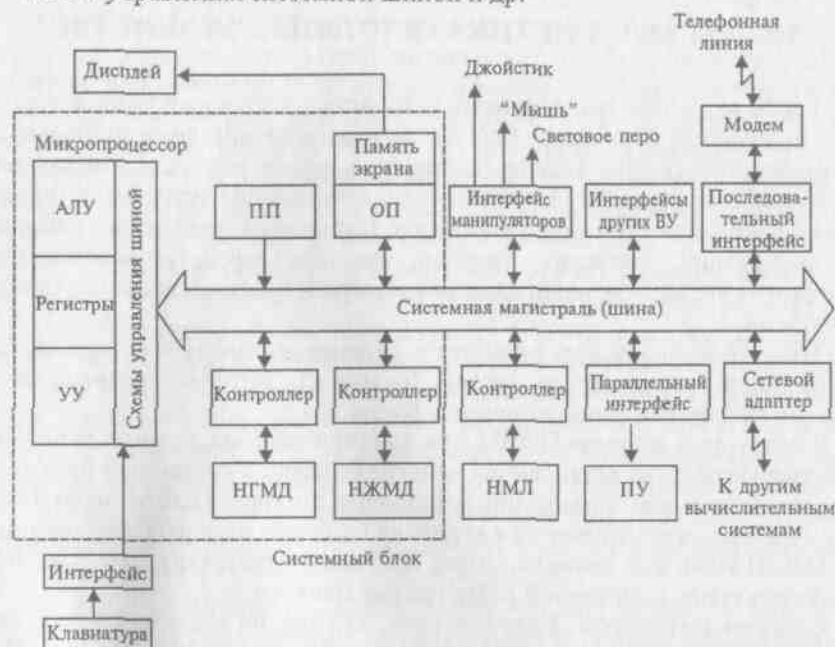


Рис. 6.1. Структурная схема ПЭВМ с периферийными устройствами:

АЛУ — арифметико-логическое устройство; УУ — устройство управления;
ПП — постоянная память; ОП — оперативная память; ВУ — внешнее устройство;
НГМД — накопитель на гибких магнитных дисках; НЖМД — накопитель на жестких магнитных дисках; НМЛ — накопитель на магнитной ленте; ПУ — печатающее устройство

Рассмотрим структуру и функционирование микропроцессора на примере разработанной фирмой Intel модели i486.

АЛУ выполняет логические операции, а также арифметические операции в двоичной системе счисления и в двоично-десятичном коде, причем арифметические операции над числами, представленными в форме с плавающей точкой, реализуются в специальном блоке. В некоторых конфигурациях с этой целью используется *арифметический сопроцессор* (например, i80387). Он имеет собственные регистры данных и управления, работает параллельно с центральным МП, обрабатывает данные с плавающей точкой.

Память МП состоит из функциональных регистров: регистры общего назначения, указатель команд, регистр флагов и регистры сегментов.

Восемь 32-разрядных регистров общего назначения используются для хранения данных и адресов. Они обеспечивают работу с данными разрядностью 1, 8, 16, 32 и 64 бита и адресами размером 16 и 32 бита. Каждый из таких регистров имеет свое имя, например EAX или ESP.

32-разрядный указатель команд содержит смещение при определении адреса следующей команды. 32-разрядный регистр флагов указывает признаки результата выполнения команды.

Регистры сегментов содержат значения селекторов сегментов, определяющих текущие адресуемые сегменты памяти.

Кроме вышеуказанных регистровая память МП содержит регистры процессора обработки чисел с плавающей точкой, системные и некоторые другие регистры.

Производительность микропроцессора значительно повышается за счет буферизации часто используемых команд и данных во внутренней КЭШ-памяти размером (в данном случае) 8 Кбайт. При этом сокращается число обращений к внешней памяти. Внутренняя КЭШ-память имеет несколько режимов работы, что обеспечивает гибкость отладки и выполнения рабочих программ.

Устройство управления микропроцессорного типа обеспечивает конвейерную обработку данных с помощью блока предварительной выборки (очереди команд).

Устройство управления МП обеспечивает многозадачность. Многозадачность — способ организации работы ПЭВМ, при котором в ее памяти одновременно содержатся программы и данные для выполнения нескольких задач. В составе МП i486 имеются аппаратно-программные средства, позволяющие эффективно организовать многозадачный режим, в том числе системы прерывания и защиты памяти.

Система прерываний обрабатывает запросы на прерывание как от внешних устройств, так и от внутренних блоков МП. Поступление запроса на прерывание от внутреннего блока МП свидетельствует о возникновении исключительной ситуации, например о переполнении разрядной сетки. Внешнее прерывание может быть связано с обслуживанием запросов от периферийных устройств. Требуя своевременного обслуживания, внешнее устройство посылает запрос прерывания МП. МП в ответ приостанавливает нормальное выполнение текущей программы и переходит на обработку этого запроса, чтобы в дальнейшем выполнить определенные действия по вводу-выводу данных. После совершения таких действий происходит возврат к прерванной программе. МП i486 способен обрабатывать до 256 различных типов прерываний, причем первые 32 типа отведены для внутрисистемных целей и недоступны пользователю.

Защита памяти от несанкционированного доступа в многозадачном режиме осуществляется с помощью системы привилегий, регулирующих

доступ к тому или иному сегменту памяти в зависимости от уровня его защищенности и степени важности.

Обмен информацией между блоками МП происходит через *магистраль микропроцессора*, включающую 32-разрядную шину адреса, 32-разрядную двунаправленную шину данных и шину управления.

Шина адреса используется для передачи адресов ячеек памяти и регистров для обмена информацией с внешними устройствами.

Шина данных обеспечивает передачу информации между МП, памятью и периферийными устройствами. По этой шине возможна пересылка 32-, 16- и 8-разрядных данных. Шина двунаправленная, т.е. позволяет осуществлять пересылку данных как в прямом, так и в обратном направлении.

Шина управления предназначена для передачи управляющих сигналов — управления памятью, управления обменом данных, запросов на прерывание и т.д.

Системная магистраль выполняется в виде совокупности шин (кабелей), используемых для передачи данных, адресов и управляющих сигналов. Количество линий в адресно-информационной шине определяется разрядностью кодов адреса и данных, а количество линий в шине управления — числом управляющих сигналов, используемых в ПЭВМ.

Внутренняя память ПЭВМ состоит из оперативной памяти и постоянной памяти (ПП).

Оперативная память (ОП) ПЭВМ. Она построена на БИС или СБИС и является энергозависимой: при отключении питания информация в ОП теряется. В оперативной памяти хранятся исполняемые машинные программы, исходные и промежуточные данные и результаты. Емкость ОП в ПЭВМ измеряется в Кбайтах и Мбайтах. Иногда адресное пространство увеличивается до Гбайта. В наиболее распространенных конфигурациях ПЭВМ емкость ОП составляет 1 — 16 и более Мбайт.

В ОП обычно выделяется область, называемая стеком. Обращение к стековой памяти возможно только в той ячейке, которая адресуется указателем стека. Стек удобен при организации прерываний и обращении к подпрограммам.

Постоянная память (ПП). Она является энергозависимой, используется для хранения системных программ, в частности так называемой базовой системы ввода-вывода (BIOS — Basic Input and Output System), вспомогательных программ и т.п. Программы, хранящиеся в ПП, предназначены для постоянного использования МП.

Контроллеры (К) служат для управления внешними устройствами (ВУ). Каждому ВУ соответствует свой контроллер. Электронные модули-контроллеры реализуются на отдельных печатных платах, вставляемых внутрь системного блока. Такие платы часто называют адаптерами ВУ (от адаптировать — приспособливать). После получения команды от

МП контроллер функционирует автономно, освобождая МП от выполнения специфических функций, требуемых для того или другого конкретного ВУ.

Контроллер содержит регистры двух типов — регистр состояния (управления) и регистр данных. Эти регистры часто называют портами ввода-вывода. За каждым портом закреплен определенный номер — адрес порта. Через порты пользователь может управлять ВУ, используя команды ввода-вывода. Программа, выполняющая по обращению из основной выполняемой программы операции ввода-вывода для конкретного устройства или группы устройств ПЭВМ, входит в состав ядра операционной системы ПЭВМ.

Для ускорения обмена информацией между МП и внешними устройствами в ПЭВМ используется прямой доступ к памяти (ПДП). Контроллер ПДП, получив сигнал запроса от внешнего устройства, принимает управление обменом на себя и обеспечивает обмен данными с ОП, минуя центральный МП. В это время микропроцессор продолжает без прерывания выполнять текущую программу. Прямой доступ к памяти, с одной стороны, освобождает МП от непосредственного обмена между памятью и внешними устройствами, а с другой стороны, позволяет значительно быстрее по сравнению с режимом прерываний удовлетворять запросы на обмен.

6.3. ВНЕШНИЕ ЗАПОМИНАЮЩИЕ УСТРОЙСТВА ПК

В качестве ВЗУ в ПЭВМ в основном используются НГМД и НЖМД типа винчестер.

Накопители на гибких магнитных дисках служат для хранения программ и данных небольшого объема и удобны для перенесения информации с одной ПЭВМ на другую.

На рабочей поверхности диска (дискеты) по концентрическим окружностям, размещенным на определенном расстоянии от центрального отверстия, записываются данные. Стандартный формат дискеты для IBM PC и совместимых с ней ПЭВМ имеет 40(80) дорожек. Каждая дорожка разделена на части, называемые секторами или записями. Секторы представляют собой основную единицу хранения информации на дискете. При чтении или записи устройство всегда считывает или записывает целое число секторов независимо от объема запрашиваемой информации.

Емкость сектора (число байтов или слов) — основная характеристика формата данных на носителе. Она определяется наименьшим количеством данных, которое может быть считано или записано на дискету за одну операцию ввода-вывода.

Данные на дискете могут размещаться как на одной стороне, так и на двух ее сторонах.

Важной характеристикой дискеты является плотность записи. Дискеты могут быть с одинарной, двойной и повышенной (учетверенной) плотностью записи. При одинарной плотности записи на двусторонней дискете диаметром 5,25 дюйма сохраняется до 780 Кбайт, а при повышенной плотности записи емкость НГМД составляет до 1,2 Мбайт.

Существует два способа разбивки (разметки) дорожек на секторы: фиксированный (или аппаратный) и программный. Если размер сектора задан жестко и определяется механическими характеристиками устройства, такая разметка называется *фиксированной*. При фиксированной разметке индексные отверстия, расположенные по кругу, обозначают начало каждого сектора и, следовательно, его положение на дискете точно определено.

Для стандартных дискет ПЭВМ размером 133 мм (5,25 дюйма) расположение дорожек на дискете и число сторон неизменны: они определяются характеристиками самих дискет. Однако количество секторов на дорожке и их размер могут определяться программно в процессе разметки (форматирования). Именно поэтому гибкие диски называют также дисками с программной разметкой секторов (*soft-sector*). Форматирование выполняется либо программами операционной системы, либо программами BIOS.

Размер сектора 5,25-дюймового диска, поддерживаемого системой BIOS, может составлять 128, 256, 512 и 1024 байт.

В последние годы широкое распространение получили НГМД диаметром 3 дюйма. Их емкость достигает 1,44 Мбайт. Достоинством этих НГМД по сравнению с 5,25-дюймовыми дискетами является не только большая компактность, но и наличие жесткого пластикового корпуса со специальной металлической сдвигающейся крышкой, защищающего рабочие поверхности дискеты от загрязнения и механических повреждений. Специальные сдвигающиеся рычажки на корпусе дискеты обеспечивают ее механическую защиту от записи.

Как правило, современные настольные ПЭВМ имеют ВЗУ для обоих типов дискет, а портативные — лишь для 3-дюймовых.

Накопители на жестких магнитных дисках (винчестер) содержат несколько дисков, объединенных в пакет. Чаще всего такой пакет включает 4 — 6 дисков диаметром 5,25 или (в портативных ПЭВМ) 3 дюйма. НЖМД является несменяемым, располагается внутри системного блока.

В НЖМД магнитные головки, объединенные в блок, перемещаются одновременно в радиальном направлении по отношению к дискам. Дорожки с одинаковыми номерами на разных поверхностях дисков образуют цилиндр. Цилиндр имеет тот же номер, что и объединенные им дорожки.

Любой диск имеет физический и логический формат.

Физический формат диска определяет размер сектора (в байтах), число секторов на дорожке (или — для жестких дисков — в цилиндре), число дорожек (цилиндров) и число сторон.

Логический формат диска задает способ организации информации на диске и фиксирует размещение информации различных типов.

В настоящее время наиболее распространены НЖМД емкостью от 80 до 240 Мбайт. Вместе с тем нередкими стали конфигурации ПЭВМ, включающие НЖМД типа винчестер емкостью в 500 Мбайт и даже в 1 Гбайт.

Важным параметром для пользователя является *время доступа*, характеризующее скорость чтения и записи информации на диски. Для наиболее распространенных НЖМД оно колеблется от 14 до 70 мкс. Реальная скорость работы НЖМД в большой степени зависит от типа используемой программы. Так, обработка больших массивов информации, требующая многократного поиска одиночных сведений, может неожиданно для пользователя занять весьма значительное время. Еще более продолжительной может оказаться обработка сложных изображений.

Расширение внешней памяти достигается подключением к системному блоку стримера. *Стример* — это устройство для быстрой перезаписи данных с жесткого диска на магнитную ленту. Обычно емкость стримера составляет до 60 Гбайт.

В последние годы появились устройства для хранения информации на оптических (лазерных) дисках. Их емкость измеряется гигабайтами и даже десятками гигабайт, однако в большинстве случаев такие диски не допускают перезаписывания, поэтому используются для хранения постоянной информации (например, сложных компьютерных игр с высококачественной графикой).

6.4. ВНЕШНИЕ УСТРОЙСТВА ПК

Эффективность использования ПЭВМ в большой степени определяется количеством и типами внешних устройств, которые могут применяться в ее составе. Внешние устройства обеспечивают взаимодействие пользователя с ПЭВМ. Широкая номенклатура внешних устройств, разнообразие их *технико-эксплуатационных и экономических характеристик* дают возможность пользователю выбрать такие конфигурации ПЭВМ, которые в наибольшей мере соответствуют его потребностям и обеспечивают рациональное решение его задач.

Внешние устройства составляют до 80% стоимости ПЭВМ и оказывают значительное (иногда даже решающее) влияние на характеристики машины в целом.

Конструктивно каждая модель ПЭВМ имеет так называемый базовый набор внешних устройств — клавиатуру, дисплей, НЖМД и один

или два НГМД, составляющий вместе с системным блоком «базовую конфигурацию» этой модели. Пользователь, как правило, сам подбирает желательное ему печатающее устройство. В случае необходимости к ПЭВМ могут подключаться также дополнительные внешние устройства, например сканеры, стримеры, плоттеры или диджитайзеры. В последние годы многие фирмы прилагают значительные усилия для разработки совершенно новых видов внешних устройств, ориентированных на стремительно растущие запросы пользователей, в частности для приложений в области мультимедиа.

Клавиатура (клавишное устройство) реализует диалоговое общение пользователя с ПЭВМ:

ввод команд пользователя, обеспечивающих доступ к ресурсам ПЭВМ;

запись, корректировку и отладку программ;

ввод данных и команд в процессе решения задач.

Центральную часть клавиатуры обычно занимают клавиши букв латинского и русского алфавита, служебных знаков (!, », :, %, ... и др.), а также цифровые клавиши. В большинстве случаев одна клавиша используется для ввода нескольких разных знаков, причем переход между ними производится за счет одновременного нажатия соответствующей клавиши и одной и/или двух служебных функциональных клавиш (обычно — клавиш Shift, Alt, Ctrl). В большинстве моделей клавиатуры (за исключением клавиатуры ПЭВМ классов LAPTOP, NOTEBOOK, HANDHELD) с правой стороны размещается дополнительная цифровая клавиатура, что создает удобства при необходимости частого ввода чисел. По периферии клавиатуры размещаются служебные функциональные клавиши: Enter, Esc, Delete, Insert, Tab и др., а также «программируемые» функциональные клавиши (F1 — F12). Функциональные клавиши в программах выполняют в основном специальные операции. К примеру, клавиша Esc обычно означает «отмену» или «возврат», клавиша Insert — «вставку» и т.п. Назначение программируемых функциональных клавиш F1 — F12 более гибко: как правило, определяется в соответствующих программах и приводится в их документации. Служебные клавиши (Shift, Alt, Ctrl) и индикаторы режимов (Print screen, Caps Lock, Break) служат для переключения назначения алфавитно-цифровых клавиш, вывода «образа экрана дисплея» на принтер, изменения режима работы и прерывания программ. Клавиши управления (←↑→ и ↓) необходимы для позиционирования курсора на экране дисплея. Ряд клавиш обеспечивают перемещение курсора в начальную или конечную позицию на строке экрана дисплея (Home, End), а также на страницу вперед или назад (PgUp и PgDn).

Типовые размеры клавиатуры 40×450×180 мм. При разработке клавиатуры учитывается возможность предельного сокращения нажатий на

клавиши пользователем. Это достигается изменением значений отдельных клавиш программным путем. Клавиатура ПЭВМ передает МП не код символа, а порядковый номер нажатой клавиши и продолжительность времени каждого нажатия. Интерпретация смысла нажатой клавиши выполняется программным путем. Таким образом, кодировка клавиши оказывается независимой от кодировки символов, что значительно упрощает работу с клавиатурой.

Дисплей (монитор) — основное устройство для отображения информации, выводимой во время работы программ на ПЭВМ. Дисплеи могут существенно различаться, от их характеристик зависят возможности машин и используемого программного обеспечения. Различают дисплеи, пригодные для вывода лишь алфавитно-цифровой информации, и графические дисплеи. Другой важный признак — возможность поддержки цветного или только монохромного изображения. Важными техническими параметрами являются текстовый формат и разрешающая способность изображения. Текстовый формат (в текстовом режиме) характеризуется числом символов в строке и числом текстовых строк на экране. В графическом режиме разрешающая способность задается числом точек по горизонтали и числом точечных строк по вертикали. Другой характерный параметр — количество поддерживаемых уровней яркости в монохромном режиме и соответственно количество цветов при цветном изображении. Не менее важным параметром является и размер экрана: он определяет различимость изображения в целом и четкость его отдельных элементов, в том числе букв и цифр.

Указанные параметры зависят как от конструкции экрана, так и от схемы управления, сосредоточенной в системном блоке. В настоящее время в большинстве случаев применяется схема формирования изображения на основе растровой памяти (*bit mapping*). Каждый элемент изображения — одна точка на экране дисплея формируется из фрагмента растровой памяти, состоящего из 1, 2 или 4 бит. Информация, записанная в указанных битах, управляет яркостью (или цветом) точки на экране, а также ее миганием и другими возможными атрибутами.

Объем растровой памяти прямо связан с разрешающей способностью дисплея. Дисплею, к примеру, с двумя уровнями яркости и разрешающей способностью 640×480 точек требуется 26 Кбайт растровой памяти. Если же при этом необходимо управлять 16 цветами для каждой точки, требуемый объем растровой памяти составит не менее 64 Кбайт, а при двухцветном экране с разрешающей способностью 1024×768 потребуется уже 132 Кбайт растровой памяти. При таком методе управления изображением знаки выводятся на экран при помощи специальных знакогенераторов — особых электронных схем, управляемых точечными матрицами, на которых формируется изображение каждого символа.

Большинство профессиональных ПЭВМ использует дисплеи, основанные на монохромных или цветных ЭЛТ. Наиболее часто в IBM-

совместимых ПЭВМ используются мониторы типа VGA или SVGA, а в более ранних моделях — CGA, EGA, Hercules.

В профессиональных ПЭВМ широко применяются цветные мониторы с очень высоким разрешением (1024×768 и 1600×1200 точек) и возможностью получения изображений из 4096 базовых цветов, что обеспечивает до 16 млн. оттенков.

Пользователи ПЭВМ проводят в непосредственной близости от работающих дисплеев многие часы подряд. В связи с этим фирмы-производители дисплеев усилили внимание к оснащению экранов дисплеев специальными средствами защиты от всех видов воздействий, которые негативно сказываются на здоровье пользователя. Так, фирма Samsung выпускает дисплеи «Low Radiation» с нанесенным на экран специальным покрытием, снижающим уровень жесткого излучения. Используются и другие методы, повышающие комфортность работы с дисплеями.

Общение пользователя с ПЭВМ облегчается с помощью различных *манипуляторов*. Наиболее распространенным из них является так называемая мышь. Мышь представляет собой небольшую коробочку с двумя или тремя клавишами и утопленным свободно вращающимся в любом направлении шариком на нижней поверхности. Коробочка подключается к компьютеру при помощи специального кабеля. Пользователь, перемещая мышь по поверхности стола (обычно для этого используются специальные резиновые коврики), позиционирует указатель мыши (стрелку, прямоугольник) на экране дисплея, а нажатием клавиш выполняет определенное действие, связанное с соответствующей клавишей (например, выполняет определенный пункт меню). Мышь требует специальной программной поддержки.

В портативных ПЭВМ мышь обычно заменяется особым встроенным в клавиатуру шариком на подставке с двумя клавишами по бокам, называемым *трекбол*. Позиционирование указателя трекбола на экране дисплея производится вращением этого шарика. Клавиши трекбола имеют то же значение, что и клавиши мыши. Несмотря на наличие трекбола, пользователь портативной ПЭВМ может использовать и обычную мышь, подключив ее к соответствующему порту.

Для непосредственного считывания графической информации с бумажного или иного носителя в ПЭВМ применяются оптические *сканеры*. Сканеры бывают настольные, позволяющие обрабатывать весь лист бумаги или пленки целиком, а также ручные. Ручные сканеры проводят над нужными рисунками или текстом, обеспечивая их считывание. Введенный при помощи сканера рисунок распознается ПЭВМ с помощью специального программного обеспечения. Рисунок может быть не только сохранен, но и откорректирован по желанию пользователя соответствующими графическими пакетами программ. В настоящее время выпускаются черно-белые и цветные сканеры с точностью разрешения до 8000

точек на дюйм (более 300 точек на 1 мм), однако эти устройства весьма дороги. Использование сканеров для непосредственного ввода в ПЭВМ текстовой информации с ее последующим редактированием затруднено также значительной сложностью программного обеспечения, необходимого для правильного распознавания и интерпретации отдельных символов.

Для той же цели, т.е. для ввода рисунков в ПЭВМ, может использоваться также так называемое световое перо и различные диджитайзеры.

К ручным манипуляторам относится и *джойстик* (*joystick*), представляющий собой подвижную рукоять с одной или двумя кнопками, при помощи которой можно позиционировать указатель на экране дисплея. Кнопки имеют то же назначение, что и клавиши мыши. Джойстик чаще используется в бытовых ПЭВМ, в первую очередь для игровых применений.

6.5. ПЕЧАТАЮЩИЕ УСТРОЙСТВА ПЭВМ

В ПЭВМ используются матричные, лепестковые, струйные и лазерные принтеры.

Матричные принтеры наиболее распространены. Печатаемые знаки синтезируются в матричных принтерах при помощи игольчатой матрицы (головки),двигающейся вдоль каждой печатаемой строки по специальной направляющей и ударяющей по красящей ленте. Чаще всего применяются принтеры с 9- и 24-игольчатыми головками. Эти принтеры позволяют получить вполне приемлемое для большинства приложений качество печати, в том числе за счет многократных проходов при печати одной строки с небольшими смещениями. Вместе с тем это снижает и без того невысокую скорость печати. Недостатком матричных принтеров следует считать и довольно значительный уровень производимого при печати шума.

При выборе матричного принтера следует обратить внимание на максимальную ширину применяемой бумаги (узкий или широкий принтер), возможность использования рулонной бумаги, количество игл в матрице (головке), скорость печати, указываемые в паспортных данных. Важной характеристикой матричного принтера, также указываемой в его паспорте, являются количество и виды встроенных шрифтов и возможность печати кириллицы. Вместе с тем большинство современных программных систем обработки текстов (Word, Word for Windows, Word Perfect, Lexicon и др.) включают специальные «загружаемые» шрифты (*soft fonts*).

Качество печати, обеспечиваемое матричными принтерами, практически не уступает качеству, обеспечиваемому пишущей машинкой, однако оно совершенно недостаточно при работе с графикой, а также

для изготовления оригинал-макетов, которые можно было бы использовать в полиграфии.

Лазерные принтеры обладают многообразными возможностями печати, обеспечивают ее высокое качество при значительной скорости.

Лазерные принтеры имеют собственный расширяемый блок памяти. Они позволяют масштабировать шрифты, широко использовать «загружаемые» шрифты. «Паспортная» скорость печати у различных моделей лазерных принтеров, как правило, колеблется от 4 до 64 страниц в минуту. Вместе с тем эта скорость зависит от объема собственной памяти принтера (и может заметно сократиться при ее недостатке для конкретной печатаемой информации) и сложности выводимого изображения.

Лазерные принтеры используют исключительно листовую бумагу (форматов А4, А3 и др.), в связи с чем существенное значение приобретает емкость подающего бумагу лотка, так как от нее зависит скорость работы принтера: бумагу необходимо периодически подкладывать в лоток вручную. Недостатком лазерных принтеров являются довольно жесткие требования к качеству бумаги — она должна быть достаточно плотной и не должна быть рыхлой, недопустима печать на бумаге с пластиковым покрытием и т.д.

Особенно эффективны лазерные принтеры при изготовлении оригинал-макетов книг и брошюр, рекламных проспектов, деловых писем и иных материалов, требующих высокого качества. Они позволяют с большой скоростью печатать графики, рисунки.

В последние годы появилась целая гамма лазерных принтеров, обеспечивающих не только черно-белую, но и многокрасочную цветную печать.

Даже самые простые модели лазерных принтеров в пять — десять раз дороже средних моделей матричных принтеров, а цена цветных лазерных принтеров более чем стократно превосходит цену матричных. Весьма дороги и сменные картриджи, содержащие красящий порошок. Все это делает лазерные принтеры малопригодными для изготовления значительных тиражей, поскольку печать одного листа обходится существенно дороже ксерокопии.

Струйные принтеры в последние годы получают все более широкое распространение среди пользователей ПЭВМ. Этот тип принтера занимает промежуточное положение между матричными и лазерными принтерами. Струйные принтеры, являясь, как и матричные, построчно печатающими, обеспечивают качество печати, приближающееся к качеству лазерных принтеров. Они просты в эксплуатации и работают практически бесшумно. При работе под управлением соответствующих программных средств струйные принтеры позволяют печатать вполне удовлетворительные по качеству гра-

фические материалы. Вместе с тем скорость печати, обеспечиваемая струйными принтерами, ненамного превосходит скорость печати матричными принтерами, а их стоимость — в два-три раза выше. Струйные принтеры вполне успешно применяются во всех случаях, когда скорость печати и качество не являются критическими факторами. Красящая жидкость («чернила») для струйных принтеров помещается в специальных компактных картриджах. Она производится нескольких цветов, так что простой заменой картриджа можно обеспечить печать многоцветных изображений. Ряд моделей струйных принтеров допускает одновременную многоцветную печать.

Графопостроители (плоттеры) применяются для вывода графической информации в ПЭВМ. Плоттеры значительно дешевле, чем лазерные принтеры, хотя скорость вывода изображений у них значительно ниже. Достоинством плоттеров по сравнению с лазерными принтерами является также возможность использования для печати крупноформатной бумаги и пленки (вплоть до формата А0). Плоттеры выпускают двух типов — рулонные и планшетные. В рулонных плоттерах бумажный лист перемещается транспортирующим валиком в вертикальном направлении, а пишущий узел — в горизонтальном. Рулонные принтеры позволяют получать полноцветные изображения хорошего качества. В планшетных плоттерах лист бумаги фиксируется горизонтально на плоском столе, а пишущий узел (одно или несколько разноцветных перьев) перемещается по направляющим в двух направлениях — по осям X и Y. Планшетные плоттеры обеспечивают более высокую по сравнению с рулонными точность печати рисунков и графиков.

6.6. ОПЕРАЦИОННАЯ СИСТЕМА И ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ ПК

Операционная система (ОС) ПЭВМ представляет собой программу, которая автоматически загружается при включении ПК и представляет собой базовый набор команд, с помощью которых пользователь может производить общение с ЭВМ и выполнять ряд действий, например: запуск программ, форматирование дискеты, копирование файлов и т.д. ОС обеспечивает интерфейс пользователя с ПК и обработку данных.

Операционные системы делятся на однопрограммные, многозадачные и многопользовательские (рис. 6.2). К однопрограммным ОС относятся SCP, MS-DOS и др. Многозадачные ОС (Unix, Windows начиная с версии 3.1, DOS 7.0, OS/2 и др.) предусматривают одновременное выполнение нескольких приложений.

Если однопрограммные системы работают или в пакетном режиме, или в диалоговом, то многозадачные ОС могут совмещать эти режимы. Таким образом, эти ОС обеспечивают пакетную и диалоговую технологии обработки данных.

Многопользовательские ОС отвечают требованиям пользователей различных категорий (неквалифицированных пользователей, прикладных и системных программистов) и профессий. Эти ОС реализуются также сетевыми ОС Novell Netware, обеспечивают сетевые технологии, а также пакетные и диалоговые технологии.



Рис. 6.2. Классификация пользовательского интерфейса

Разнообразие ПЭВМ и ОС привели к новому понятию — платформа. Платформа определяет тип компьютера и ОС, на которых устанавливается та или иная информационная технология. Она имеет сложную структуру. Главным компонентом является тип компьютера, определяемый типом процессора: Intel, Motorola, Atary, Sincler и др. Следующим компонентом является ОС, работающая на том или ином процессоре. Например, Windows NT работает на многих типах процессоров: Intel, MIPS, ALPHA, Power PC.

Технология общения с ПК зависит от интерфейса. Современные ОС поддерживают командный, WIMP- и SILK-интерфейсы (рис. 6.2.). Ставится вопрос о создании так называемого «общественного» интерфейса (social interface). Командный интерфейс означает выдачу на экран системного приглашения для ввода команды. Например, в MS DOS это приглашение выглядит как C:\>, в Unix — \$.

WIMP-интерфейс является графическим и расшифровывается как Windows (окна), Image (образ), Menu (меню), Pointer (указатель), т.е. на экране высвечивается окно, содержащее образы программ и меню действий. Для выбора одного из них используется указатель. SILK-интерфейс означает Speech (речь), Image (образ), Language (язык) и Knowledge (знание), т.е. на экране по речевой команде происходит перемещение от

одних поисковых образов к другим. Предполагается, что при использовании общественного интерфейса не нужно будет разбираться в меню. Экранные образы однозначно укажут дальнейший путь перемещения от одних поисковых образов к другим по смысловым семантическим связям.

Пользовательским интерфейсом называют набор приемов взаимодействия пользователя с приложением. Пользовательский интерфейс включает три понятия: общение пользователя с приложением и язык общения.

Рассмотрим кратко основные положения стандарта пользовательского интерфейса. Пользовательский интерфейс зависит от интерфейса, обеспечиваемого ОС. Свойствами интерфейса являются конкретность и наглядность. Графическая система Windows является очень удобной, а богатство ее возможностей сделали ее оптимальной системой для повседневной работы. Приложения, написанные по Windows, используют тот же интерфейс, поэтому его единообразие сводит к минимуму процесс обучения работе с любым приложением Windows.

Разработка пользовательского интерфейса состоит из проектирования панелей и диалога.

Панель приложения разделена на три части: меню действий, тело панели и область функциональных клавиш.

Меню действий (и выпадающее меню) заключается в том, что действия наглядны и могут быть запрошены пользователем установкой курсора, функциональной клавишей, вводом команды или каким-то другим простым способом. На цветном экране меню действий обычно имеет другой цвет по отношению к цвету панели. На монохромном экране используется сплошная линия для его отделения. Меню действий содержит объекты, состоящие из одного или нескольких слов. Два последних из них резервируются для действий «выход» и «справка». Размещаются объекты слева направо по мере убывания частоты их использования.

Тело панели содержит следующие элементы: разделители областей, идентификатор панели, заголовок панели, инструкцию, заголовок столбца и группы, заголовок поля, указатель протяжки, область сообщений, область команд, поле ввода, поле выбора.

Область функциональных клавиш — необязательная часть, показывающая соответствие клавиш и действий, которые выполняются при нажатии.

Для указаний текущей позиции на области функциональных клавиш принят принцип «объект — действие». Этот принцип разрешает пользователю сначала выбрать объект, а затем произвести действие с этим объектом, что минимизирует число режимов, упрощает и ускоряет обучение работе с приложениями. Если панель располагается в отдельной ограниченной части экрана, то она называется окном, которое может быть первичным или вторичным. В первичном окне диалог начинается, и если в приложении не нужно создавать другие окна, то окном считается весь эк-

ран. Первичное окно может содержать столько панелей, сколько нужно для ведения диалога. *Вторичные окна* вызываются из первичных. В них пользователь ведет диалог параллельно с первичным окном. Часто ведется диалог параллельно с первичным окном. Часто вторичные окна используются для подсказки. Первичные и вторичные окна имеют заголовок в верхней части окна. Пользователь может переключаться из первичного окна во вторичное и наоборот. Существует также понятие *всплывающих окон*, которые позволяют расширить диалог пользователя с приложением, ведущийся из первичного или вторичного окна. В основном всплывающие окна используются для передачи сообщений или подсказки.

Когда пользователь и ЭВМ обмениваются сообщениями, диалог движется по одному из путей приложения, т.е. пользователь движется по приложению, выполняя конкретные действия. При этом действие обязательно требует от приложения обработки информации. Диалоговые действия также контролируют информацию, которую набирает пользователь. Если пользователь перешел к другой панели и его действия могут привести к потере информации, рекомендуется требовать подтверждения о том, следует ли ее сохранить. При этом пользователю предоставляется шанс сохранить информацию, отменить последний запрос, вернуться на один шаг назад.

Путь, по которому движется диалог, называется *навигацией*. Он может быть изображен в виде сети или графа, где узлы — действия, а дуги — переходы. Диалог состоит из двух частей: *запросов на обработку информации и навигации по приложению*. Часть запросов на обработку и навигацию является унифицированной. Унифицированные действия диалога — это действия, имеющие одинаковый смысл во всех приложениях. Некоторые унифицированные действия могут быть запрошены из выпадающего меню, посредством действия «команда», функциональной клавишей. К унифицированным действиям диалога относятся следующие: «отказ», «команда», «ввод», «выход», «подсказка», «регенерация», «извлечение», «идентификатор», «клавиши», «справка».

6.7. ТЕХНОЛОГИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ В ПК

Технология информационных процессов в любом типе ЭВМ имеет сходную структуру и состоит из операций и этапов. *Операция* — это совокупность элементарных действий, выполняемых на одном рабочем месте, которая приводит к реализации определенной обработки данных. Под операцией понимается любой процесс, связанный с обработкой данных. *Этап* — это совокупность взаимосвязанных операций, которые реализуют определенную законченную функцию обработки данных.

В технологии информационных процессов выделяют следующие этапы: первичный, предварительный, основной и заключительный.

Первичный этап технологии информационных процессов состоит из сбора, регистрации и передачи информации на обработку.

На *предварительном этапе* осуществляется прием, визуальный контроль данных, регистрация, кодирование, комплектование, перенос на машинный носитель, заполнение и формирование первичного документа.

При *визуальном контроле* проверяется четкость заполнения, отсутствия пропусков реквизитов, исправление ошибок. Для сокращения объема вводимой в ЭВМ информации применяется *операция кодирования*, т.е. присвоения кодов одному или нескольким реквизитам. Обычно кодируются наименования, для чего разработаны специальные справочники и классификаторы.

Комплектование данных — операция вынужденная. При вводе больших объемов данных их разбивают на комплекты (пачки). Каждой пачке присваивается свой номер, который также вводится в ЭВМ. Комплектование облегчает поиск информации и исправление ошибок, обеспечивает контроль полноты вводимых данных, позволяет прерывать процесс ввода или подготовки данных на машинном носителе.

Операция переноса на машинный носитель выполнялась на больших ЭВМ. Основными носителями были перфокарты, перфоленты и магнитные ленты. В настоящее время эта операция часто совмещается с непосредственным вводом информации в ЭВМ с клавиатуры, специальных устройств, считывающих данные с документов, штрих-коды, а также с получением данных по сети или по запросу из БД.

Первичный и предварительный этапы технологии информационных процессов были выделены при обработке данных на больших ЭВМ, так как они выполнялись на разных рабочих местах в условиях пооперационной технологии. При обработке информации на ПК эти два этапа чаще всего объединяются в один *домашний этап*, на котором все операции практически выполняются вручную.

Основной этап технологии информационных процессов содержит операции: ввод данных в ЭВМ; контроль безопасности данных и систем; сортировка; корректировка; группировка; анализ; расчет; формирование отчета и вывод данных. Так как все эти операции выполняются ПК, то этот этап называют *внутримашинным*.

Операция ввода данных — это одна из основных и сложных операций технологии информационных процессов. Данные могут быть представлены в виде бумажного документа, в образе электронного документа, электронной таблицы, штрих-кодов, могут быть запрошены из БД, получены по сети, вводиться с клавиатуры, а в перспективе будет осуществляться речевой ввод. Поэтому ввод данных в ЭВМ обязательно сопровождается операцией контроля, так как неверные данные нет смысла обрабатывать.

Сами данные могут быть любого типа: текстовые, табличные, графические, в виде знаний, объектов реального мира и т.д. При этом подсистемы ИС обычно имеют дело с разнородными данными, приходящими из различных источников. После ввода и контроля данные могут быть записаны в файл, показаны на дисплее, переданы в БД в режиме ее актуализации, переданы по сети. Чаще всего данные записываются в файл или БД.

Контроль безопасности данных и систем подразделяется на контроль достоверности данных, безопасности данных и компьютерных систем. *Контроль достоверности данных* выполняется программно во время их ввода и обработки. Средства безопасности данных и программ защищают их от копирования, искажения несанкционированного доступа. Средства безопасности компьютерных систем обеспечивают защиту от кражи, вирусов, неправильной работы пользователей, несанкционированного доступа.

Сортировка данных используется для упорядочения записей файла по одному или нескольким ключам. *Запись* — это минимальная единица обмена между программой и внешней памятью. *Файл* — это совокупность записей. Обычно одна запись содержит информацию одного документа или его законченной части. Структура записи и файла определяется пользователем при проектировании. *Ключ* — это реквизит или группа реквизитов, служащих для идентификации записей. Сортировка упрощает дальнейшую обработку данных. Она присутствует во всех файловых системах.

Корректировка — это операция актуализации файла или БД. При этом обычно выполняются операции просмотра, замены, удаления, добавления нового. Эти операции применяются к отдельным реквизитам, записи, группе записей, файлу, БД. *Группировка* состоит в операции соединения записей, сходных по одному или нескольким ключам, в относительно самостоятельные новые объекты — группы.

Анализ — это операция, реализующая метод научного исследования, основанный на расчленении целого на составные части, разбор, рассмотрение чего-либо. Для проведения анализа используются экономико-математические и статистические методы, методы выявления тенденций, прогнозирования, моделирования, построение графиков, диаграмм, экспертные методы.

Расчет — это операция, позволяющая выполнить требуемые вычисления для получения результатов или промежуточных данных.

Операция *формирования отчетов* состоит в оформлении результатов расчета для вывода и передачи данных потребителю в привычном для него виде.

Последняя операция — вывод — служит для вывода результатов обработки данных на печать, в БД, файл, дисплей, по сети ЭВМ.

Заключительный этап технологии информационного процесса содержит следующие операции: визуальный контроль результа-

тов, размножение и передачу потребителю. Этот этап также называют *послемашинным*. При установке ПК на рабочем месте пользователя этот этап может содержать только операции контроля: четкость вывода, непротиворечивость результатов и т.д. Все остальные операции могут выполняться на машинном этапе. Так как существует система электронной подписи, а потребителем является сам пользователь, то результаты обработки данных передаются либо по сети либо записываются в БД.

6.8. РАБОТА С ТЕКСТОМ, ГРАФИКОЙ И ТАБЛИЦАМИ

Работа с текстом. Одной из самых распространенных работ на ПК является редактирование текстовых данных. Для работы с текстом используются текстовые процессоры (или редакторы). ПЭВМ всегда оснащена одним или несколькими текстовыми процессорами.

К настоящему времени разработано много текстовых процессоров. В целом назначения всех текстовых редакторов одно, но предоставляемые возможности и средства их реализации — разные. То же относится к графическим процессорам и электронным таблицам. Среди текстовых процессоров Windows (как наиболее распространенной среды) можно выделить Write и Word. Технология их использования основана на интерфейсе WIMP, но возможности Word значительно расширены и в какой-то мере его можно рассматривать как настольную типографию.

Текстовые процессоры обеспечивают следующие функции: набор текста, хранение его на магнитных носителях, просмотр и печать. В большинстве текстовых процессоров реализованы функции проверки орфографии, выбора шрифтов и кеглей, центровки заголовков, разбиения на страницы, печати в одну или несколько колонок, вставки в текст таблиц и рисунков, использования шаблонов, подстраничных ссылок, перемещения кусков текста, изменения структуры документа.

С помощью средств форматирования можно создать внешний вид документа, изменить стиль, подчеркнуть, выделить курсивом, изменить размеры символов, выделить абзацы, выровнять их влево, вправо, к центру, выделить рамкой. Перед печатью документ можно просмотреть, проверить текст, выбрать размер бумаги, задать число копий для вывода. Повторяющиеся участки текста можно обозначить как автотекст и присвоить ему имя. В дальнейшем вместо данного текста достаточно указать его имя, а текстовый процессор автоматически заменит его.

Работа с графическим объектом. Потребность ввода графиков, схем, диаграмм, рисунков в текст или документ вызвала необходимость создания графических процессоров. Графические процессоры представляют собой инструментальные средства, позволяющие создавать и модифицировать графические образы с использованием иллюстративной, коммерческой, научной и когнитивной графики.

ИТ иллюстративной графики позволяют создавать иллюстрации для различных текстовых документов в виде регулярных структур — различные геометрические фигуры (так называемая «векторная графика»). Процессоры, реализующие ИТ иллюстративной растровой графики, дают возможность пользователю выбрать толщину и цвет линий, палитру заливки, шрифт для записи и наложения текста, создавать разные графические образы. Кроме этого, пользователь может стереть, разрезать рисунок и перемещать его части. Эти средства реализованы в пакете Paint Brush. Существуют также ИТ, позволяющие просматривать изображения в режиме слайдов, спецэффектов и вживлять их (Corell Draw, Storyboard, Animator, 3dStudio).

ИТ коммерческой, или деловой, графики обеспечивают отображение информации, хранящейся в табличных процессорах, БД и отдельных локальных файлах в виде двух- или трехмерных графиков, круговой диаграммы столбиковой гистограммы, линейных графиков и др. (Excel, FoxPro и т.д.).

ИТ научной графики предназначены для обслуживания задач оформления научных расчетов, содержащих математические, химические и прочие формулы, задач картографии и др.

Когнитивные компьютерные средства представляют собой комплекс виртуальных устройств, программ и систем, реализующих комплексную обработку зрительной информации в виде образов, процессов, структур и позволяющих средствами диалога реализовать методы и приемы представления условий задачи или подсказки в виде зрительных образов. К о г н и т и в н а я г р а ф и к а позволяет образно представить различные математические формулы и закономерности для доказательства сложных теорем, открывает новые возможности для познания законов функционирования сознания — этой наиболее сложной и сокровенной тайны мироздания. Средства когнитивной графики связаны со многими новейшими ИТ, включая гипертекст и мультимедиа.

Большинство графических процессоров удовлетворяют стандарту пользовательского интерфейса WIMP. Панель содержит меню действий и линейки инструментов и цветов. Линейка инструментов состоит из набора графических символов, требующихся для построения практически любого рисунка. Линейка из цветов содержит цветовую гамму монитора ПК.

Работа с таблицами. Документы табличного вида составляют большую часть документооборота предприятия любого типа. Поэтому табличные ИТ представляют особую важность. Комплекс программных средств, реализующих создание, регистрацию, редактирование, хранение и обработку электронных таблиц и выдачу их на печать, называется *табличным процессором*. *Электронная таблица* — это двумерный массив строк и столбцов, размещенный в памяти ЭВМ. Широкое распространение получили такие табличные процессоры, как SuperCalc, Visicalc,

Lotus 1-2-3, Quattro Pro. Для Windows был создан процессор Excel, технология работы с которым аналогична работе с любым приложением Windows интерфейса WIMP.

Табличный процессор позволяет решать большинство статистических, финансовых и административных задач, например таких, как расчет заработной платы; прогнозирование продаж, роста рынка, доходов, анализ процентных ставок и налогов; подготовка финансовых деклараций и балансовых таблиц; ведение бухгалтерских книг для учета платежей; сметные калькуляции; учет денежных чеков; бюджетного расчета и многое другое.

Основное единицей электронной таблицы является рабочий лист, имеющий имя, где он располагается. Место пересечения строки со столбцом называется *ячейкой* или *полем*. Существуют два варианта адресации ячеек: абсолютная и относительная. Абсолютная адресация наиболее употребительная. *Адресом ячейки (идентификатором)* служит буква, указывающая столбец, и цифра, указывающая номер строки. И то, и другое видно на рабочем листе. При относительной адресации в верхней строке состояния указывается приращение со знаком от начала искомой клетки. В нижней строке рабочего листа дается расшифровка выбранного действия меню. В верхней части располагается меню действий, панель инструментов и строка сумматора, где отражаются все воспроизводимые действия.

Ширина столбца и высота строки даются по умолчанию. Однако имеется возможность форматирования ячейки, столбца, строки, листа. При этом можно изменить стиль текста, что позволяет улучшить внешний вид документа без применения текстового редактора.

Данные в виде чисел, текста или формул вводятся в ту ячейку, которая отмечена текстовым курсором. Для указания блока ячеек достаточно указать адрес левой верхней ячейки диагонали блока, поставить точку или двоеточие, адрес нижней правой ячейки диагонали или наоборот. Блок можно задать выделением.

Редактирование таблиц позволяет копировать, удалять, очищать ячейку, блок, лист и выполнять многие другие функции, перечисленные в меню действий «правка» и «вставка». Посредством OLE-технологии (Objekt Linked and Embedding) в таблицу можно вставить рисунок, график, диаграмму, любой объект, представленный другой программой.

Большинство электронных таблиц имеют средства создания графиков и диаграмм, средства их редактирования и включения в нужное место листа. Кроме того, в них имеется большое число встроенных функций: математических, статистических и других. Это существенно облегчает процесс вычислений и расширяет диапазон применений. Пользователю представляется возможность переопределить панель инструментов, вид рабочего листа, изменить масштабирование, включить полосы прокрутки, переключатели, меню. Сервисные функции табличного

процессора excel позволяют проверить орфографию текста, защитить данные от чтения или записи. Возможно создание диалоговых окон или обращение к динамическим библиотекам Windows.

В табличном процессоре Excel есть средство создания макросов — Visual Basic. Он является объектно-ориентированным языком программирования. Отличие его, например, от Pascal или C++ в том, что в Visual Basic нет возможности создавать новые типы объектов или порождать потомки уже существующих. Однако пользователь получает большой набор готовых объектов: рабочие книги, листы, ячейки, диаграммы и т.д.

Все табличные процессоры позволяют создавать БД и представляют учебные средства работы с ними.

В Microsoft Excel 5.0 имеется один тип файла — *рабочая книга*, состоящая из рабочих листов, листов диаграмм и макросов, но при этом все листы подшиты в рабочую книгу. Такой подход упрощает работу с несколькими документами за счет быстрого доступа к каждому листу через ярлычки в нижней части листа, позволяет работать с листами, объединенными в группу, например, группу учетных карточек на товар. При этом если производится группа действий на одном листе, то эти действия автоматически повторяются на всех листах группы, что упрощает оформление однотипных по структуре листов. Объемные ссылки позволяют создавать сводные документы на основе данных из нескольких листов без ввода громоздких формул с внешними ссылками. Микротехнология «*мастер сводных таблиц*» позволяет выбрать нужные данные из документа, представить их сводной таблицей, изменяя структуру, внешний вид, добавляя итоговые строки, а также группируя и сортируя эти данные. В рабочую книгу можно включать информацию о теме, авторе, ключевых словах. Ее можно использовать при поиске файла на диске или при выяснении его назначения.

При выполнении всех функций в процессоре Excel можно использовать многооконную систему, позволяющую выполнять параллельные действия. Все объекты, созданные пользователем (сформированные таблицы, макросы, выборки из БД, диаграммы и графики), можно сохранить на диске, в виде файла или распечатать.

6.9. ПЕРСПЕКТИВЫ РАЗВИТИЯ ПК

Перспективы развития ПЭВМ во многом определяются функциональными возможностями, технико-эксплуатационными характеристиками и архитектурным построением МП.

В настоящее время крупнейшим производителем МП в мире является фирма Intel. Последняя модель фирмы Intel — МП Pentium используется в мощных настольных ПЭВМ, рабочих станциях и многопроцессорных серверах.

МП Pentium имеет суперскалярную архитектуру, два конвейера с отдельными исполнительными устройствами, встроенный сопроцессор с плавающей точкой. Локальная шина PCI (Peripheral Component Interconnect) предназначена для передачи данных между процессором и высокоскоростной периферийной ПЭВМ. Пропускная способность PCI достигает 132 Мбайт/с, причем возможно дальнейшее наращивание пропускной способности до 264 Мбайт/с. Шина PCI позволяет использовать одни и те же высокоскоростные периферийные устройства в ПЭВМ с процессорами Intel, ALPHA или RISC. Использование шины PCI с процессором Pentium обеспечивает максимальную пропускную способность при работе с сетями, графикой, дисками и т.п. Фирма Intel рекомендует использовать МП Pentium при решении задач моделирования, трехмерного проектирования, для создания серверов и многопроцессорных систем.

Прогресс в развитии МП обеспечивается использованием новых архитектурных решений, в частности транспьютерной и RISC-архитектуры, конвейерного выполнения команд, применения сопроцессоров, параллельной обработки данных и т.п.

Рассмотрим особенности МП с архитектурой RISC (Reduced Instruction Set Computer — «компьютер с сокращенной системой команд»). В этих МП применяется сравнительно небольшой (сокращенный) набор наиболее часто употребляемых команд, определенный в результате статистического анализа большого числа программ. Для RISC-архитектуры характерны следующие факторы: все команды имеют одинаковый формат; большинство команд — трехадресные; большое количество внутренних регистров МП, позволяющее резко сократить число обращений к ОП, а следовательно, уменьшить время машинного цикла; конвейеризация выполнения команд; наличие КЭШ-памяти. Ограниченный набор команд сравнительно простой структуры дает возможность уменьшить количество аппаратуры.

При одной и той же тактовой частоте ПЭВМ RISC-архитектуры имеют производительность в 2 — 4 раза выше, чем ПЭВМ обычной архитектуры. В настоящее время ПЭВМ с RISC-архитектурой применяются в качестве графических рабочих станций, серверов локальных сетей, являются основой для создания современных управляющих, телекоммуникационных и банковских сетей. На отечественном рынке предлагаются модели новых систем RISC/6000 — семи моделей серверов и четырех моделей рабочих станций на базе МП Power PC (фирма IBM). МП Power PC 601 работает с тактовой частотой от 50 до 200 МГц. Фирма Apple несколько лет назад выпустила в продажу модель Power Macintosh 8100/80 на базе процессора Power PC 601 RISC 80 МГц со встроенным сопроцессором и кэш-памятью 32 Мбайт, ОП емкостью 16 Мбайт (с возможностью расширения до 264 Мбайт) и винчестером емкостью 1000 Мбайт.

Транспьютеры, как правило, используются в качестве сопроцессоров. Они рассчитаны на работу в параллельных системах с однотипными процессорными элементами и аппаратной поддержкой вычислительных процессоров. В состав системы команд транспьютеров входят команды управления процессами, поддержки инструкций, языков высокого уровня. Транспьютеры используют коммуникационные быстрые каналы, которые позволяют передавать по одной магистрали данные в процессор, а по другой (одновременно) — данные из него. Высокая производительность обеспечивается прежде всего за счет высокой скорости работы АЛУ и передачи операндов.

В современных МП широко применяются КЭШ-память и виртуальная память, что приближает ПЭВМ по функциональным возможностям к большим ЭВМ. В ПЭВМ стали использовать многозадачный режим работы, динамическое распределение памяти, системы защиты памяти.

На отечественном компьютерном рынке появилось большое количество разнообразных моделей ПЭВМ многих фирм. Большинство из них, однако, построены на МП фирмы Intel. Менее распространены модели ПЭВМ, созданные на базе МП фирмы Motorola.

В настоящее время тенденции в развитии производства ПЭВМ сводятся к насыщению рынка ПЭВМ следующих трех классов: профессиональные многопроцессорные ПЭВМ, приблизившиеся по своим параметрам к большим ЭВМ; сравнительно недорогие ПЭВМ для массового потребителя; микроминиатюрные ПЭВМ (типа NOTEBOOK и HANDHELD). При выборе ПЭВМ следует обращать внимание прежде всего на скорость работы МП (на его тактовую частоту), которая во многом определяет диапазон применения компьютера. Важным фактором эффективности использования ПЭВМ является емкость ОП, обеспечивающая возможность хранения набора программ, которые планируется выполнять на выбранной ПЭВМ. Емкость ОП для работы в среде Windows, например, должна быть 16 Мбайт.

Периферийные устройства необходимо выбирать по их технико-эксплуатационным параметрам, ориентируясь на классы задач, которые должны решаться на ПЭВМ, условия эксплуатации ПЭВМ, а также удобства, предоставляемые пользователю. При этом следует учитывать возможность развития потребностей пользователя. Это особенно важно при выборе НЖМД, поскольку многие современные программные средства требуют для своего хранения десятков мегабайт памяти на НЖМД типа винчестер.

В настоящее время появились мощные переносные ПК и серверы, TV PC — компьютерные телеприемники, Wallet PC — компьютерные бумажники, помещающиеся в кармане, Kiosk PC — устройства, заменяющие телефонный аппарат и предоставляющие широкий спектр услуг от видеоконференции до пересылки денежных сумм. Ключом такого рода технологических достижений остается микропроцессор.

1. Что такое ПЭВМ?
2. В чем отличие ПЭВМ от ЭВМ других классов?
3. Из каких устройств состоит системный блок? Каково их назначение?
4. Какие периферийные устройства входят в состав ПЭВМ?
5. Назовите назначение и основные характеристики периферийных устройств.
6. Расскажите об известных Вам внешних запоминающих устройствах ПК.
7. Какие виды печатающих устройств ПК Вы знаете?
8. Что такое операционная система ПК?
9. Дайте определение понятию «пользовательский интерфейс».
10. Из каких этапов состоит технология информационных процессов в ПК?
11. Назовите функции наиболее распространенных текстовых процессоров.
12. Расскажите о работе на ПК с графическими объектами.
13. Дайте краткую характеристику табличному процессору.
14. Как Вы понимаете «диалоговый режим работы ПК»?
15. Назовите основные направления развития ПЭВМ.

ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ГЛАВА 7

КЛАССИФИКАЦИЯ И ХАРАКТЕРИСТИКА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Современные информационные системы организационного управления предназначены оказывать помощь специалистам, руководителям, принимающим решения, в получении им своевременной, достоверной, в необходимом количестве информации, созданий условий для организации автоматизированных офисов, проведения с применением компьютеров и средств связи оперативных совещаний, сопровождаемых звуковым и видеорядом. Достигается это переходом на новую информационную технологию.

Новая (современная) информационная технология — это технология, основанная на:

*повсеместном применении ЭВМ и оргтехники;
активном участии пользователей (непрофессионалов в области вычислительной техники и программировании) в информационном процессе;*

*высоком уровне дружественного пользовательского интерфейса;
широком использовании пакетов прикладных программ (ППП) общего и проблемного назначения;*

возможности для пользователя доступа к базам данных и программ, в том числе и удаленным, благодаря локальным и глобальным сетям ЭВМ;

анализе ситуаций при выработке и принятии управленческих решений с помощью автоматизированных рабочих мест специалистов;

применении систем искусственного интеллекта;

*внедрении экспертных систем;
использовании телекоммуникации;
создании геоинформационных систем и других технологий.*

Методические указания

Преподаватель должен помочь студентам уяснить две очень важные мысли: первая — информационная технология — это умение правильно работать с информацией, которое обеспечивается совокупностью программно-технических средств и приемов; вторая — в настоящее время разработано довольно большое число информационных технологий и в будущем их возможности будут очень быстро расширяться.

В результате изучения содержания седьмой главы студенты должны знать: определение и назначение баз данных; понятия признака и запроса простого и сложного на поиск информации в информационно-поисковых системах (ИПС); основные операции с данными, допускаемые ИПС; назначение систем прикладного программного обеспечения; основные классы задач, решаемые с помощью ППП; назначение и функциональные возможности АРМ; назначение и функциональные возможности систем искусственного интеллекта и экспертных систем.

Студенты должны понимать, что:

ППП делают применение ЭВМ массовым, освобождая пользователя от самостоятельного составления сложных программ; программным средством можно пользоваться даже не зная его структуры и методов, положенных в его основу; с помощью одного и того же ППП можно решать различные классы задач (иногда не очень похожих друг на друга);

ИТ существенно облегчают хранение и поиск нужной информации; для различных жизненных задач нужны разные ИТ; ресурсы ЭВМ (в первую очередь объем памяти и быстродействие) и их эксплуатационные характеристики влияют на возможности, предоставляемые ИТ.

Студенты должны уметь: использовать основные возможности ППП: текстового и графического редакторов, электронных таблиц, математических и экономических ППП; пользоваться простейшими ИТ: изменять и добавлять данные, искать информацию, составляя простые и сложные запросы, сортировать данные, хранящиеся в БД.

Студенты должны иметь представление о: классификации ИТ; структуре и функциях элементов БД; применении АРМ; видах систем искусственного интеллекта; системах представления знаний; видах ЭС и типах решаемых ими задач; назначении и структуре геоинформационных систем.

7.1. КЛАССИФИКАЦИЯ И ХАРАКТЕРИСТИКА СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Стремительное развитие и совершенствование эксплуатационных возможностей ЭВМ создало предпосылки для автоматизации умственного труда и формирования рынка информационных продуктов и услуг. Развитие ИТ шло параллельно с появлением новых видов технических средств обработки и передачи информации, новыми средствами коммуникаций.

В настоящее время классификация ИТ проводится по следующим признакам: способу реализации в автоматизированных информационных системах (АИС), степени охвата задач управления, классам реализуемых технологических операций, типу пользовательского интерфейса, вариантам использования сети ЭВМ, обслуживаемой предметной области и др.

По способу реализации ИТ делятся на традиционные и современные ИТ. *Традиционные ИТ* существовали в условиях централизованной обработки данных, до периода массового использования ПЭВМ. Они были ориентированы главным образом на снижение трудоемкости пользователя (например, инженерные и научные расчеты, формирование регулярной отчетности на предприятиях и др.). Новые (современные) ИТ связаны в первую очередь с информационным обеспечением процесса управления в режиме реального времени.

По степени охвата ИТ задач управления выделяют: электронную обработку данных, автоматизацию функций управления, поддержку принятия решений, электронный офис, экспертную поддержку. В первом случае *электронная обработка данных* выполняется с использованием ЭВМ без пересмотра методологии и организации процессов управления при решении локальных математических и экономических задач. Во втором случае при *автоматизации управленческой деятельности* вычислительные средства, включая суперЭВМ и ПЭВМ, используются для комплексного решения функциональных задач, формирования регулярной отчетности и работы в информационно-справочном режиме для подготовки управленческих решений. К этой же группе относятся *ИТ поддержки принятия решений*, которые предусматривают широкое использование экономико-математических методов и моделей, ППП для аналитической работы и формирования прогнозов, составления бизнес-планов, обоснованных оценок и выводов по процессам и явлениям производственно-хозяйственной деятельности. К названной группе относятся и широко внедряемые в настоящее время ИТ, получившие название электронного офиса и экспертной поддержки

решений. Эти два варианта ИТ ориентированы на использование достижений в области новейших подходов к автоматизации работы специалистов и руководителей, создание для них наиболее благоприятных условий выполнения профессиональных функций, качественного и современного информационного обслуживания за счет автоматизированного набора управленческих процедур, реализуемых в условиях конкретного рабочего места и офиса в целом.

Электронный офис предусматривает наличие интегрированных ППП, которые обеспечивают комплексную реализацию задач предметной области. В настоящее время все большее распространение приобретают электронные офисы, сотрудники и оборудование которых могут находиться в разных помещениях. Необходимость работы с документами, материалами и БД конкретного предприятия или учреждения в гостинице, транспорте, дома привела к появлению электронных офисов, включенных в соответствующие сети ЭВМ.

ИТ экспертной поддержки составляют основу автоматизации труда специалистов-аналитиков. Эти работники кроме аналитических методов и моделей для исследования складывающихся ситуаций вынуждены использовать накопленный опыт в оценке ситуаций, т.е. сведения, составляющие базу знаний в конкретной предметной области.

По классу реализуемых технологических операций ИТ подразделяются: на работу с текстовым и табличным процессорами, графическими объектами, системы управления БД, гипертекстовые и мультимедийные системы. Работа с *текстовым и табличным процессорами*, а также с *графическими объектами* детально рассмотрена в §6.8. Здесь лишь подчеркнем, что создание программных средств для вывода высококачественного звука и видеоизображения является перспективным направлением развития компьютерной технологии. Технология формирования видеоизображения получила название компьютерной графики. *Компьютерная графика* — это создание, хранение и обработка моделей объектов и их изображений с помощью ЭВМ. Эта технология проникла в область моделирования различных конструкций (машиностроение, авиационная техника, автомобилестроение, строительная техника и др.), экономического анализа, проникает в рекламную деятельность, делает занимательным досуг. Формируемые и обрабатываемые с помощью цифрового процессора изображения могут быть демонстрационными и анимационными. К *демонстрационным изображениям* относят, как правило, коммерческую (деловую) и иллюстрационную графику. Ко второй группе — *анимационной графике* — принадлежит инженерная и научная графика, а также графика, связанная с рекламой, искусством, играми, когда на экран выводятся не только одиночные изображения, но и последовательность кадров в виде фильма (интерактивный вариант). *Интерактивная графика* является одним из наиболее прогрессивных направлений среди современных ИТ. Это на-

правление переживает бурное развитие в области появления новых графических станций и в области специализированных программных средств, позволяющих создавать реалистические объемные движущиеся изображения, сравнимые по качеству с кадрами видеофильма.

В классическом понимании *система управления БД (СУБД)* представляет собой набор программ, позволяющих создавать и поддерживать БД в актуальном состоянии.

Обычно любой текст представляется как одна длинная строка символов, которая читается в одном направлении. *Гипертекстовая технология* заключается в том, что текст представляется как многомерный, т.е. иерархической структурой. Материал текста делится на фрагменты. Каждый видимый на экране ЭВМ фрагмент, дополненный многочисленными связями с другими фрагментами, позволяет уточнить информацию об изучаемом объекте и двигаться в одном или нескольких направлениях по выбранной связи.

Программно-техническая организация обмена с компьютером текстовой, графической, аудио и видеоинформацией получила название *мультимедиа-технологии*.

По типу пользовательского интерфейса можно рассматривать ИТ с точки зрения возможностей доступа пользователя к информационным и вычислительным ресурсам. Так, *пакетная ИТ* исключает возможность пользователя влиять на обработку информации, пока она проводится в автоматическом режиме. Это объясняется организацией обработки информации, которая основана на выполнении программно-заданной последовательности операций над заранее накопленными в системе и объединенными в пакет данными. В отличие от пакетной *диалоговая ИТ* предоставляет пользователю неограниченную возможность взаимодействовать с хранящимися в системе информационными ресурсами в реальном масштабе времени, получая при этом всю необходимую информацию для решения функциональных задач и принятия решений. Интерфейс *сетевой ИТ* предоставляет пользователю средства теледоступа к территориально распределенным информационным и вычислительным ресурсам благодаря развитым средствам связи.

В настоящее время наблюдается тенденция к объединению различных типов ИТ в единый компьютерно-технологический комплекс, который носит название *интегрированного*. Особое место в нем принадлежит средствам коммуникации, обеспечивающим не только чрезвычайно широкие технологические возможности автоматизации управленческой деятельности, но и являющимся основой создания самых разнообразных сетевых вариантов ИТ: *локальных, многоуровневых, распределенных и глобальных информационно-вычислительных сетей*.

По обслуживаемым предметным областям ИТ подразделяются чрезвычайно разнообразно. Например, только в экономике

ими являются: бухгалтерский учет, банковская, налоговая и страховая деятельность и др.

7.2. АВТОМАТИЗИРОВАННЫЕ БАНКИ ДАННЫХ

В настоящее время в нашей стране накоплен большой опыт разработки автоматизированных систем управления. Этот опыт говорит о том, что центральным техническим вопросом разработки АСУ и любых других информационных систем является организация, хранение и комплексное использование данных. В конечном счете это привело к созданию развитых средств управления данными, которые являются основой любой информационной системы, построенной на базе использования средств вычислительной техники.

Автоматизированные системы управления, спроектированные на основе концепции банков данных, обладают целым рядом характерных свойств, которые выгодно отличают их от предшествующих разработок, основой которых была система массивов данных, ориентированная на решении комплекса установившихся задач. Использование автоматизированных банков данных позволяет обеспечить многоаспектный доступ к совокупности взаимосвязанных данных, достаточно высокую степень независимости прикладных программ от изменений логической и физической организации данных, интеграции и централизации управления данными, устранение излишней избыточности данных, возможность совмещения пакетов и телепроцессорной обработки данных.

Поэтому разработки АСУ для любой сферы применения связаны прежде всего с созданием автоматизированных банков данных.

Так как основой любого управления является информация о состоянии объекта, то именно поэтому данные в автоматизированных системах, их организация, тщательное ведение, хранение, использование являются центром системы. Меняются техника, программное хозяйство, но данные остаются, работа с ними оказывается делом достаточно дорогим и именно поэтому задумались над системными принципами их организации, положенными в основу создания банков данных.

Под *автоматизированным банком данных* понимается организационно-техническая система, представляющая собой совокупность баз данных пользователей, технических и программных средств формирования и ведения этих баз и коллектива специалистов, обеспечивающих функционирование системы.

В самом общем виде основные функции банка данных можно сформулировать следующим образом:

адекватное информационное отображение предметной области;
обеспечение хранения, обновления и выдачи необходимых данных пользователям.

Составными частями любого банка данных являются база данных, система управления базой данных (СУБД), администратор базы данных, прикладное программное обеспечение (рис. 7.1).



Рис. 7.1. Составные части банка данных

Функционирование системы управления базой данных основано на введении двух уровней организации базы данных — логического и физического. Эти два уровня соответствуют двум аспектам организации данных: физическому с точки зрения хранения данных в памяти ЭВМ и логическому с точки зрения использования данных в прикладных приложениях.

Описание логических организаций баз данных определяет взгляд пользователей на организацию данных в системе, которые отображают состояние некоторой предметной области.

Необходимо отметить, что в общем случае структуры физической и логической организации данных могут не совпадать. Формальное описание логической организации данных иногда называют моделью данных или схемой.

Говоря о физической организации, необходимо отметить, что существует много различных способов организации данных в запоминающей среде, с помощью которых можно обеспечить соответствие некоторой модели.

Наиболее общее представление о базе данных заключается в следующем: база данных (*data base*) — это совокупность хранимых во внешней памяти ЭВМ большого объема данных; база данных является «интегрированной», т.е. представляет собой комплекс взаимосвязанных данных, предназначенный для обеспечения информационных нужд различных пользователей, каждый из которых имеет отношение к отдельным, возможно, совместно используемым частям данных; работа с базой данных может осуществляться либо в пакетном режиме, либо с удаленных терминалов в режиме реального времени. Структура системы баз данных представлена на рис. 7.2.

Таким образом, база данных — это совокупность хранимых в памяти ЭВМ и специальным образом организованных взаимосвязанных данных, отображающих состояние предметной области. База данных также предназначена для обеспечения информационных нужд определенных пользователей.

Создание единой базы данных о предметной области сложно и в настоящее время практически нереализуемо, хотя бы из-за недостаточного объема памяти современных ЭВМ. На практике большинство баз дан-

ных проектируется для ограниченного числа приложений. На одной ЭВМ, как правило, создается несколько различных баз данных. Со временем некоторые базы данных, предназначенные для выполнения родственных функций, могут объединиться, если это будет способствовать повышению производительности всего вычислительного комплекса.



Рис. 7.2. База данных

Создание баз данных обеспечивает интеграцию данных и возможность централизованного управления данными.

Объекты и отношения объектов. Любая информационная система должна отображать те или иные стороны окружающего нас реального мира или, как иногда говорят, проблемной или предметной области. Мы воспринимаем окружающий мир состоящим из объектов, которые человек, по совокупности определенных достаточно устойчивых свойств, группирует в наборы (классы) объектов, которым он присваивает имя. Например, в реальном мире есть конкретные собаки, но нет собаки «вообще». Понятие «собаки» описывает целый класс в каком-то смысле однородных реальных объектов.

Проблемная среда изменяется со временем, что выражается в изменении свойств объектов, возникновении новых и исчезновении старых объектов. Эти изменения происходят в результате событий. Временная последовательность событий образует процесс.

Всякая информационная система имеет дело не с самими объектами, как реальными сущностями, а с их знаковыми отображениями-идентификаторами. Главная функция знака-идентификатора — отличить объект в группе однородных объектов. Идентификатор объекта, вообще говоря, может не нести никакой информации о свойствах объекта или, что то же самое, об его принадлежности к тому или иному классу.

Например, 11591 — табельный номер служащего — является числовым идентификатором. Этот идентификатор не описывает свойства, их приходится задавать дополнительно.

Более полно объект описывается записью об объекте, которая обычно состоит из идентификатора объекта-знака, позволяющего отличить один объект от другого среди однородных объектов, и идентификаторов (значений) свойств (атрибутов). Например, запись о служащем некоторой организации имеет табельный номер служащего в качестве идентификатора и такие элементы данных, как должность, заработная плата, льготы и т.д., рассматриваемые как идентификаторы (значения) свойств служащего.

Следует подчеркнуть, что понятие объекта и свойства относительны. Если речь идет о служащем, то естественно понимать должность как свойство служащего. Но если речь идет о должности, например, в смысле должностных инструкций, то уже сама должность выступает в качестве объекта, который может иметь свойства. В частности, в определенном контексте табельный номер служащего может рассматриваться как свойство должности.

Поэтому при информационном отображении предметных сред (рис. 7.3) можно (а иногда и нужно) говорить не об объектах и их свойствах, а об отношениях объектов, ибо в этом случае все идентификаторы в записи можно рассматривать симметрично, а не в ориентации на один специально выделенный объект. Как видно из рис. 7.3 это соответствует так называемой реляционной точке зрения на базу данных.

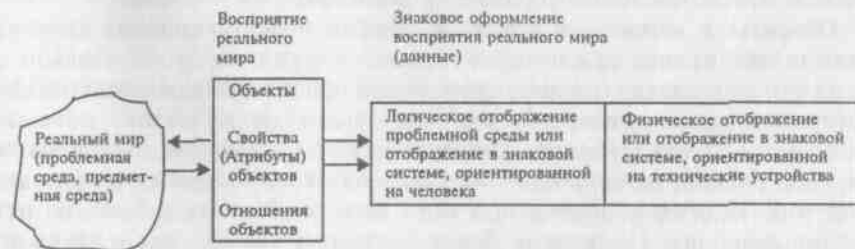


Рис. 7.3. Отображение предметных сред в банках данных

При информационном отображении реального мира весьма важно, в каких количественных пропорциях могут осуществляться отношения объектов. Четкое понимание того, к какой категории относится отношение объектов, позволяет сделать заключение о возможном характере связи между соответствующими данными. Фиксация этой стороны при информационном отображении предметной области определяет одну из сторон модели данных. Важно подчеркнуть, что характер отношений одних и тех же объектов не есть нечто застывшее. Оно может измениться и тогда изменится характер связей между элементами данных, который

может оказать существенное влияние на структуру банка данных, как логическую, так и физическую. Усложнение характера связей между данными делает более сложными программы их обработки.

Данные. Информация об объекте или отношениях объектов, выраженная в знаковой форме, образует данные. Эти данные могут быть восприняты человеком или каким-либо техническим устройством и соответствующим образом интерпретированы.

Характерной особенностью данных является то, что их можно переводить из одной знаковой системы в другую (перекодировка) без потери информации. Это существенное свойство знакового отображения позволяет описывать реальную предметную ситуацию в различных системах знаков, ориентированных на воспринимающего. При построении банков данных стало уже традиционным говорить о логическом отображении, ориентированном на человека, и о физическом отображении, ориентированном на устройства долговременной памяти.

На рис. 7.3 воспроизведены основные понятия, связанные с отображением некоторой предметной области в банке данных.

Следует отметить, что знаки сами по себе не образуют данных, несущих информацию о предметной области. В простейшем случае знаки должны быть структурно оформлены в виде фиксированной последовательности-записи, а каждое поле записи (в которое помещается знак) должно иметь интерпретацию с точки зрения предметной области, для которой создается банк данных. Например, если знаки образуются из букв русского алфавита по правилам образования слов русского языка, то, будучи взяты сами по себе, они несут только информацию о правилах образования знаков — синтаксическая информация. И в этом смысле их можно рассматривать как данные.

С точки зрения отображения предметной области для нас представляет интерес семантическая информация, а именно, как определенные знаки связаны с объектами предметной области и их отношениями. Самый простейший способ реализовать эту связь — это придать определенное содержание (смысл) полю записи. Например, если поле интерпретируется как «должность служащего», то любые знаки, помещенные в это поле, будут пониматься как конкретные идентификаторы различных должностей, и в этом смысле мы их рассматриваем как данные о предметной области.

Целесообразно кратко рассмотреть прагматический аспект знакового отображения. Если определены структуры записей обо всех объектах предметной области и их отношениях, то будем говорить, что задана модель данных предметной области.

Если предметная область обширна, например производственная деятельность предприятия, то и ее модель данных будет достаточно велика. На предприятии практически нет человека, который представляет производственную деятельность во всей ее детализации, а это значит, что

модель данных в целом никому из управляющего состава предприятия не нужна. Сама эта проблема (построение общей модели) возникла только в связи с разработкой и эксплуатацией автоматизированных информационных систем. Для каждого конкретного отдела или звена управления производственной деятельности характерна своя сфера информационных интересов, которая тоже может быть описана своей моделью данных.

По отношению к общей модели данных рассмотрим ее как подмодель. В общем случае подмодель данных — это не простое механическое усечение модели. Подмодель данных конкретного пользователя может быть связана со всей моделью данных весьма сложными структурными преобразованиями.

Понятие подмодели характеризует прагматический аспект знакового отображения.

7.2.1. МОДЕЛИ ДАННЫХ

Модель и подмодель данных. Банк данных хранит информацию об объектах реального мира и отношения между этими объектами через данные и связи между этими данными. Прежде чем говорить о размещении данных и связей между ними на устройствах памяти, необходимо представить взаимосвязь данных на логическом уровне, создав своеобразную модель данных.

Эта модель данных должна быть четко определена, для чего в системе управления базами необходимо представить средства (язык) для ее описания. Основное назначение модели данных состоит в том, чтобы дать возможность представить в целом информационную картину без отвлекающих деталей, связанных с особенностями хранения. Она является инструментом, с помощью которого разрабатывается стратегия получения любых данных, хранящихся в банке.

Термином *подмодель* определяют описание данных, используемое при прикладном программировании. На основе единой модели можно составить множество различных подмоделей. Теоретически они могут сильно отличаться от общей модели данных. Принципиально здесь то, что можно указать алгоритм, с помощью которого данные со структурой подмодели получают из базы данных, построенной в соответствии с моделью.

Рассмотрим иерархическую, сетевую и реляционную модели данных. Можно считать, что в некотором смысле эта последовательность отражает прогресс в понимании проблем обработки данных. Однако начнем рассмотрение их все-таки с сетевой модели, так как она наиболее наглядно отражает отношения объектов проблемной области.

Сетевая модель данных. Отношения объектов реального мира всегда могут быть представлены в виде некоторой сети. Это представление рисует довольно наглядную картину реальной действительности и, кроме того, претендует на то, что может быть естественным образом отражено в долговременной памяти вычислительной (информационной) системы. Пример сетевой модели данных приведен на рис. 7.4.



Рис. 7.4. Пример сетевой модели данных

Каждый узел сети соответствует элементу данных, отображающему группу однородных объектов реального мира. Тогда в реальной сети или, как говорят, экземпляре сети в каждом узле будет находиться идентификатор соответствующего объекта, например шифр детали.

На самом деле представлять и рисовать сеть на таком уровне детализации достаточно трудно. Поэтому в вершинах сети обычно стоят целые записи, состоящие из совокупности идентификаторов. Например, в вершине сети, имеющей имя деталь, фактически стоит запись (сегмент), состоящая из полей шифр детали, наименование детали и масса детали. На каком основании объединять различные элементы данных в записи (сегменты) — это предмет особого разговора.

Иерархическая модель данных. Более удобного представления данных можно достигнуть за счет увеличения информационной избыточности на уровне модели данных. Шагом на пути к увеличению информационной избыточности является переход от сетевого к иерархическому представлению данных. Такая возможность опирается на тот факт, что сеть можно представить в виде совокупности деревьев. Для этого в иерархических структурах требуется повторить и несколько преобразовать некоторые вершины сети.

Если на уровне сети, не вдаваясь в подробности физического хранения данных, база данных представляется в виде сложной объемной паутины, то на уровне иерархической модели база данных представляется в виде совокупности отдельных древовидных структур, в корнях которых стоят идентификаторы объектов, а на последующих ярусах раскрываются свойства этих объектов.

Таким образом, по крайней мере на уровне логического представления некоторые отношения и связи реального мира имитируются с помощью информационной избыточности.

Это еще не значит, что информационная избыточность будет реальной, т.е. будет иметь место и на уровне физического отображения на долговременную память. Такое видение мира более удобно с точки зрения обработки данных.

Следовательно, сетевые модели данных могут быть разложены на иерархические структуры. Однако следует отметить, что модель данных, представленная как совокупность нескольких деревьев, не обладает наглядностью, так как не создается впечатления некоторой взаимосвязанной системы данных. Вообще говоря, одновременно с использованием деревьев для описания модели данных неплохо представлять себе модель данных в виде сети, с тем чтобы ясно понимать, как с ней взаимодействуют иерархические структуры.

Сетевая модель, если она не очень громоздка, позволяет графически представить взаимодействие объектов, отображенных в базе данных. Это всегда полезно, даже если затем для описания используются иные структуры.

Реляционная модель данных. Наиболее абстрактной моделью является реляционная модель данных. Абстрактна она в том смысле, что в значительной степени ориентирована на интересы пользователя (программиста) и совершенно не несет в себе черт реального отображения на физическую память. Эта модель исторически возникла позже других, и ее появление оправдывается тем, что по мере усложнения информационных систем и прогресса в устройствах долговременной памяти поддержание математического обеспечения, несущего в себе черты «суэты представления», обходится очень дорого.

Реляционная модель получается путем дальнейшей формализации иерархической модели. В этой модели все связи между объектами задаются путем явной фиксации идентификаторов объектов в записях.

На первый взгляд реляционная модель может быть представлена в виде однородных таблиц (отношений), которые напоминают стандартные последовательные файлы. Однако это очень упрощенная точка зрения. Дело в том, что последовательный файл предполагает определенную упорядоченность и обработку в соответствии с этой упорядоченностью, или, иными словами, предполагается вход со стороны одного поля (ключа сортировки).

Существенное отличие реляционной модели от обыкновенного последовательного файла заключается в том, что все столбцы в таблице с точки зрения входа предполагаются эквивалентными. Именно это свойство делает эту модель весьма мощной и делает невозможным отображение ее на память в виде последовательного массива данных.

Информационная избыточность, по крайней мере на логическом уровне, максимальная. Однако повторим еще раз: это не предполагает

реальной информационной избыточности при отображении данных на физическую память.

На физическом уровне, например, может быть абсолютно избыточной сеть или сеть с перекрестными связями. Задача системы управления базой данных заключается в том, чтобы перевести запрос, сформулированный в терминах реляционной модели, в последовательность команд, осуществляющих поиск по сети.

Таким образом, реляционная модель предполагает очень высокую степень «интеллектуальности» систем управления базой данных.

Система управления базами данных. Она является составной частью автоматизированного банка данных и обеспечивает работу прикладных программ с базой данных.

Одним из важнейших назначений СУБД является обеспечение независимости данных. Под этим термином понимается независимость данных и использующих их прикладных программ друг от друга в том смысле, что изменение одних не приводит к изменению других. Необходимо также отметить такие возможности СУБД, как обеспечение защиты и секретности данных, восстановление баз данных после сбоев, ведение учета работы с базами данных. Однако это является неполным перечнем того, что должна осуществлять СУБД для обеспечения интерфейса пользователей с базами данных и жизнеспособности всего автоматизированного банка данных.

Система управления данными имеет набор средств, которые обеспечивают определенные способы доступа к данным. Наиболее общими операциями, которые выполняются средствами СУБД, являются операции поиска, исправления, добавления и удаления данных. Необходимо отметить, что операция поиска является главной среди указанных.

Степень реализации принципа независимости данных определяет гибкость системы управления базами данных. Учет особенностей обработки данных в какой-либо предметной области позволяет спроектировать специализированные СУБД, ориентированные на применение в АСУ предприятий с дискретным характером производства.

Существуют и универсальные системы управления базами данных, используемые для различных приложений. При настройке универсальных СУБД для конкретных приложений они должны обладать соответствующими средствами. Процесс настройки СУБД на конкретную область применения называется *генерацией системы*. К универсальным системам управления базами данных относятся, например, системы dBase, Paradox, Microsoft Access, MS DOS, UNIX, Windows, Oracle.

7.2.2. ОРГАНИЗАЦИЯ ПОИСКА ДАННЫХ

Записи логического файла идентифицируются с помощью уникальной группы символов — к л ю ч а. Обычно ключом является поле или сово-

купность полей фиксированной длины. В общем случае в качестве ключа может выступать любое поле записи. Каждому значению ключа может соответствовать одна или несколько записей файла. Ключ, каждому значению которого соответствует одна и только одна запись файла, называется первичным, или основным, ключом. Логические записи файла могут иметь несколько первичных ключей.

Каждой записи при ее хранении в памяти соответствует вполне определенное место, задаваемое адресом. Способ, задающий соответствие между основным ключом записи и ее адресом памяти, называется *способом адресации*. Основную проблему при адресации файла можно сформулировать следующим образом: как по основному ключу определить местоположение записи с данным ключом? Существует много различных способов адресации файлов, которые можно отнести к следующим трем основным группам: способы последовательной, индексной и прямой адресации. Используются также различные комбинации этих основных способов адресации.

Способы последовательной адресации характеризуются тем, что логические записи занимают непрерывный участок памяти и порядок их расположения определяется не значением ключа, а номером следования этого ключа в заданной последовательности ключей логической записи, т.е. не существует однозначного соответствия между значением одного ключа и адресов памяти.

При индексной адресации соответствие между основным ключом записи и ее адресом в памяти задается с помощью таблицы или иерархии таблиц, т.е. индекса. Способы прямой адресации характеризуются наличием некоторого алгоритма преобразования значения ключа записи непосредственно в адрес ее расположения во внешней памяти.

В зависимости от способов адресации файла могут использоваться различные способы локализации (поиска) записи с заданным значением ключа. При последовательной адресации поиск записи может осуществляться только путем просмотра (сканирования) файла с проверкой ключей записи.

Существует много различных способов поиска записей, например блочный и дихотомический поиск. Наиболее простым способом является последовательное сканирование с проверкой ключа каждой записи.

При использовании способов индексной адресации поиск записи осуществляется путем предварительного просмотра индекса. В результате поиска в индексе сокращается объем просматриваемой памяти, занимаемой непосредственно записями файла. Сам индекс и его просмотр могут быть организованы различными способами. Поиск записей файлов, использующих прямую адресацию, осуществляется в соответствии с выбранным алгоритмом преобразования значения ключа записи в ее адрес.

Администратора базы данных можно рассматривать как необходимый структурный элемент автоматизированного банка данных, т.е. банк данных включает в себя не только данные, программы и оборудование, но еще и персонал. Администратору базы данных отводится важная роль — ответственность за общее управление системой баз данных.

В обязанности администратора БД входит следующее.

Определение информационного содержания базы данных. Администратор базы данных с учетом запросов пользователей к базе принимает решение о том, какая информация должна содержаться в базе данных, т.е. определяет информационное содержание базы данных и задает их общую логическую организацию, так называемую модель данных.

Определение структуры памяти и стратегии доступа. Администратор базы должен решить, каким образом данные представляются в базе данных, т.е. разработать физическую организацию данных.

Взаимодействие с пользователем. Администратор базы данных — это лицо (или группа лиц), которое имеет глобальное представление об организации данных в системе и несет ответственность за их сохранность. Пользователи при участии администратора имеют возможность корректно определить собственный «взгляд» на базу данных, что выражается в задании подмодели данных.

Определение стратегии отказа и восстановления. Работа автоматизированной системы, использующей банк данных, существенно зависит от его успешного функционирования. В случае повреждения по какой-либо причине всей базы данных или ее некоторой части необходимо предусмотреть возможность восстановления данных с минимальной задержкой и влиянием на сохранившуюся часть базы данных. Администратор должен определить и реализовать соответствующую стратегию восстановления.

Модернизация и эффективность работы базы данных. Администратор базы данных ответствен за такую организацию системы, которая обеспечит максимальную эффективность ее функционирования, а также за выполнение всех модернизаций базы данных, направленных на более полное удовлетворение требований пользователей.

Для выполнения своих функций администратор базы данных использует набор вспомогательных программ. Эти программы составляют существенную часть системы управления базами данных. К ним относятся, например, программы ведения системного журнала, хранящего сведения о каждом обращении в базу данных, программы восстановления базы данных и программы анализа статистики использования данных.

На практике администратор БД — это чаще всего не один человек, а группа лиц, так как решаемый круг вопросов слишком широк для компетенции одного человека. Они несут ответственность за функциониро-

вание интегрированной БД, имеют полномочия по корректировке БД, отвечают как за целостность данных, так и за защиту их от несанкционированного доступа и надежность системы в целом.

Повышение требований к оперативности информационного обмена и управления, а следовательно, к срочности обработки информации привело к созданию многоуровневых систем организационного управления объектами, какими являются, например, банковские, налоговые, снабженческие, статистические и другие службы. Их информационное обеспечение поддерживают сети автоматизированных банков данных, которые строятся с учетом организационно-функциональной структуры соответствующего многоуровневого экономического объекта, машинного ведения информационных массивов. Эту проблему в новых информационных технологиях решают распределенные системы обработки данных с использованием каналов связи для обмена информацией между базами данных различных уровней. За счет усложнения программных средств управления базами данных повышаются скорости, обеспечиваются защита и достоверность информации при выполнении экономических расчетов и выработке управленческих решений.

7.3. ПАКЕТЫ ПРИКЛАДНЫХ ПРОГРАММ

Одним из условий эффективного внедрения вычислительной техники в практику является создание специализированных пакетов прикладных программ (ППП). Доступность и простота использования их создает предпосылки более широкого внедрения ЭВМ в инженерный труд, решение конкретных задач научной области, экономики, культуры, образования.

Пакеты прикладных программ обычно строятся на базе специальных систем и являются дальнейшим их развитием в конкретном направлении. Они поставляются отдельно от программного обеспечения вычислительных средств, имеют свою документацию и не входят в состав операционных систем. Многие пакеты имеют собственные средства генерации. Разработка пакета не должна требовать модификации операционных систем. Это относится к пакетам, влияющим на работу управляющих программ. Если пакет требует внесения изменений в управляющую программу, то это выполняется в процессе загрузки и инициализации пакета.

7.3.1. КЛАССИФИКАЦИЯ ППП

Все ППП могут быть разбиты на три группы: пакеты, расширяющие возможности операционных систем; пакеты общего назначения; пакеты, ориентированные на работу в АСУ.

ППП, реализующие возможности операционных систем, обеспечивают функционирование ЭВМ различных конфигураций. К ним относятся пакеты, обеспечивающие работу многомашиных комплексов типовых конфигураций, диалоговые системы, системы для работы в реальном масштабе времени, удаленную пакетную обработку.

ППП общего назначения включают в себя набор программ для широкого круга применений: для алфавитно-цифровых и графических дисплеев, графопостроителей, систем программирования для универсальных языков, систем программирования для специальных языков, а также для научно-технических расчетов, математического программирования, обработки матриц, различного вида моделирования, решения задач теории массового обслуживания и т.д.

Пакеты, ориентированные на работу в АСУ, включают в себя набор программ для общецелевых систем обработки банков данных; информационно-поисковых систем общего назначения, систем обработки документов.

Пакеты прикладных программ являются наиболее динамично развивающейся частью программного обеспечения: круг решаемых с помощью ППП задач постоянно расширяется. Во многом внедрение компьютеров практически во все сферы деятельности стало возможным благодаря появлению новых и совершенствованию существующих ППП.

Достижения в области микроэлектроники, приводящие к появлению более мощных по своим функциональным возможностям компьютеров, также являются причиной создания новых ППП. В свою очередь, необходимость улучшения характеристик использования пакета при решении конкретных задач пользователя стимулирует совершенствование архитектуры и элементной базы компьютеров и периферийных устройств.

Структура и принципы построения ППП зависят от класса ЭВМ и операционной системы, в рамках которой этот пакет будет функционировать. Наибольшее количество разнообразных ППП создано для IBM PC-совместимых компьютеров с операционными системами MS DOS и WINDOWS. Классификация этих пакетов программ по функционально-организационному признаку представлена на рис. 7.5.

Каждая группа пакетов имеет свои проблемы организации, трудности разработки и создания. Каждый пакет в зависимости от ЭВМ и его назначения реализуется на конкретном языке программирования в соответствии с требованиями, предъявленными к пакету, и возможностями языка.

В приведенной классификации не указаны игровые программы — они не являются инструментом для автоматизации, профессиональной деятельности и предназначены для доступа. Отсутствие программ-переводчиков, орфографии, электронных словарей связано с тем, что эти программы являются функциональным дополнением ППП типа редактора текста, презентации и т.п. Наблюдается тенденция включения этих программ в состав пакетов прикладных программ.



Рис. 7.5. Классификация ППП

Существующие ППП охватывают почти все сферы человеческой деятельности, связанной с обработкой информации. Развитие и совершенствование ППП — поступательный процесс, поэтому следует ожидать появления новых ППП, возможности которых превзойдут достижения настоящих пакетов.

7.3.2. ПРОБЛЕМНО-ОРИЕНТИРОВАННЫЕ ППП

Проблемно-ориентированные ППП — наиболее развитая в плане реализуемых функций и многочисленная по количеству созданных пакетов часть ППП. Она включает следующие проблемно-ориентированные программные продукты: текстовые процессоры, издательские системы, графические редакторы, демонстрационную графику, системы мультимедиа, ПО-САПР, организаторы работ, электронные таблицы (табличные процессоры), системы управления базами данных, програм-

мы распознавания символов, финансовые и аналитико-статистические и другие программы.

Текстовые процессоры — специальные программы, предназначенные для работы с документами (текстами), позволяющие компоновать, форматировать, редактировать тексты при создании пользователем документа. Обычно они включают в себя дополнительные функции по работе с блоками текста и объектами. Признанными лидерами в части текстовых процессоров для ПЭВМ являются MS WORD, WordPerfect, AmiPro.

Настольные издательские системы (НИС) — программы, предназначенные для профессиональной издательской деятельности и позволяющие осуществлять электронную верстку широкого спектра основных типов документов, типа информационного бюллетеня, краткой цветной брошюры и объемного каталога или торговой заявки, справочника. Предусмотренные в пакетах данного типа средства позволяют:

- компоновать (верстать) текст;
- использовать всевозможные шрифты и полиграфические изображения;
- осуществлять редактирование текста на уровне лучших текстовых процессоров;
- обрабатывать графические изображения;
- обеспечивать вывод документов высокого качества;
- работать в сетях и на разных платформах.

Наилучшими пакетами в этой области для ПЭВМ являются: Corel Ventura, PageMaker, QuarkXPress, FrameMaker, Microsoft Publisher, PagePlus, CompuWork Publisher.

Графические редакторы — пакеты, предназначенные для обработки графической информации. Они делятся на ППП обработки растровой графики и изображений и векторной графики.

ППП первого типа предназначены для работы с рисунками и фотографиями и включают в себя набор средств по кодированию изображений в цифровую форму. Признанный лидер среди пакетов данного класса — Adobe Photoshop. Известны также пакеты Aldus Photo Styler, Picture Publisher, Photo Works Plus. Все программы ориентированы на работу в среде Windows.

Пакеты для работы с векторной графикой предназначены для профессиональной работы, связанной с художественной и технической иллюстрацией с последующей цветной печатью (на рабочем месте дизайнеров, например), занимают промежуточное положение между пакетами для систем автоматизированного проектирования (САПР) и настольными издательскими системами.

Пакеты данного класса в настоящее время обладают достаточно широким набором функциональных средств для осуществления сложной точной обработки графических изображений и включают в себя:

- инструментарий для создания графических изображений;
- средства выравнивания (по базовой линии и странице, по сетке, пересечению, ближайшей точке и т.п.);
- средства манипулирования объектами;
- средства обработки текста в части оформления и работы с различными шрифтами;
- средства импорта (экспорта) графических объектов (файлов) различных форматов;
- средства вывода на печать с соответствующей настройкой экранного образа на полиграфическое исполнение;
- средства настройки цвета.

Своеобразным стандартом в этом классе является пакет CorelDraw. Можно также отметить такие пакеты, как Adobe Illustrator, Aldus Free Hand, Professional Draw.

Электронные таблицы (табличные процессоры) — пакеты программ, предназначенные для обработки табличным образом организованных данных. Пользователь имеет возможность с помощью средств пакета осуществлять разнообразные вычисления, строить графики, управлять форматом ввода-вывода данных, компоновать данные, проводить аналитические исследования и т.п.

В настоящее время наиболее популярными и эффективными пакетами данного класса являются Excel, Improv, Quattro Pro, Lotus 1-2-3.

Организаторы работ — это пакеты программ, предназначенные для автоматизации процедур планирования использования различных ресурсов (времени, денег, материалов) как отдельного человека, так и всей фирмы или ее структурных подразделений. Целесообразно выделить две разновидности пакетов данного класса: управления проектами и организации деятельности отдельного человека.

Пакеты первого типа предназначены для сетевого планирования и управления проектами. Достаточно простые и удобные в использовании, эти программные средства позволят быстро спланировать проект любой величины и сложности, эффективно распределить людские, финансовые и материальные ресурсы, составить оптимальный график работ и контролировать его исполнение.

К пакетам данного типа относятся: Time Line, MS Project, CA-Super Project.

Пакеты второго типа представляют собой своего рода электронных помощников делового человека. Такие пакеты, как Lotus Organizer, АСП, выполняют функции электронных секретарей и предназначены для эффективного управления деловыми контактами.

Системы управления базами данных (СУБД) — предназначены для автоматизации процедур создания, хранения и извлечения электронных данных. Многие существующие экономические, информационно-справочные, банковские, программные комплексы реализованы с использованием инструментальных средств СУБД.

Для различных классов компьютеров и операционных средств разработано множество СУБД, отличающихся по способу организации данных, формату данных, языку формирования запросов. Наиболее распространенными пакетами для ПЭВМ типа IBM PC являются dBase, Paradox, Microsoft Access, Oracle.

Пакеты демонстрационной графики являются конструкторами графических образов деловой информации, призванные в наглядной и динамической форме представлять результаты некоторого аналитического исследования.

Работа с пакетами этого типа строится по следующей схеме: разработка общего плана представления, выбор шаблона для оформления элементов, формирование и импорт элементов, таких, как текст, графики, таблицы, диаграммы, звуковые эффекты и видеоклипы. Программы просты в работе и снабжены интерфейсом, почти не требующим дополнительного изучения. К наиболее популярным пакетам данного типа относятся PowerPoint, Harvard Graphics, WordPerfect Presentations, Freelance Graphics.

Пакеты программ мультимедиа предназначены для использования ПЭВМ для отображения и обработки аудио- и видеoinформации. Помимо программных средств компьютер при этом должен быть оборудован дополнительными платами, позволяющими осуществлять ввод-вывод аналоговой информации, ее преобразование в цифровую форму.

Программы мультимедиа для ПЭВМ появились сравнительно недавно благодаря значительному росту вычислительных возможностей ПК и большим достижениям в области производства оптических дисков. Дело в том, что при представлении аналоговой информации в цифровом виде требуются огромные объемы памяти: несколько минут видеопленки занимают десятки мегабайт памяти. Естественно, что работа с таким большим файлом возможна лишь при наличии быстродействующего процессора (желательно использовать ПК с RISC-процессором и быстродействующей шиной данных). Кроме того, распространение таких мультимедиа-приложений невозможно на традиционных магнитных дискетах, для этого необходимо использовать оптические компакт-диски (CD-ROM).

Среди мультимедиа-программ можно выделить две небольшие группы. Первая включает пакеты для обучения и досуга. Поставляемые на CD-ROMах емкостью от 200 до 500 Мбайт каждый, они содержат аудиовизуальную информацию по определенной тематике. Разнообразие их огромно, и рынок этих программ постоянно расширяется при одновременном улучшении качества видеоматериалов. Так, созданы и продаются электронные энциклопедии по отраслям знаний; электронные учителя в области иностранных языков, бизнеса, политики; деловые и авантюрные игры.

Вторая группа включает программы для подготовки видеоматериалов для создания мультимедиа представлений, демонстрационных дисков и стендовых материалов.

К пакетам данного вида относятся Director for Windows, Multimedia Viewer Kit, NEC MultiSpin.

Системы автоматизации проектирования — другая разновидность пакетов программ, связанная с обработкой графических изображений. Они предназначены для автоматизации проектно-конструкторских работ в машиностроении, автомобилестроении, промышленном строительстве и т.п. Пакеты САПР обладают набором инструментальных средств, обеспечивающих реализацию следующих основных функций:

- коллективная работа в сети пользователей с пакетом;
- экспорт-импорт файлов всевозможных форматов;
- масштабирование объектов;
- управление объектами в части их группировки, передвижения с растяжкой, поворота, разрезание, изменение размеров, работа со слоями;
- перерисовка (фоновая, ручная, прерываемая);
- управление файлами в части библиотек и каталогов чертежей;
- использование разнообразных чертежных инструментов, позволяющих рисовать кривые, эллипсы, произвольной формы линии, многоугольники и т.п., использование библиотеки символов, выполнение надписей и т.д.;
- работа с цветом;
- автоматизация отдельных процедур с использованием встроенного макроязыка.

Своеобразным стандартом среди программ данного класса являются пакеты AutoCAD фирмы Autodesk. Также отметим программы DesignCAD, Grafic CAD Professional, Drawbase, Microstation, Ultimate CAD Base и Turbo CAD. Перечисленные пакеты отличаются богатством функциональных возможностей и предназначены для функционирования в среде Windows (Windows NT) или OS/2.

Программы распознавания символов предназначены для перевода графического изображения букв и цифр в ASCII-коды этих символов. Используются, как правило, совместно со сканерами.

Пакеты данного типа обычно включают разнообразные средства, облегчающие работу пользователя и повышающие вероятность правильного распознавания.

Скорость сканирования современных ППП составляет примерно 1,5 мин на страницу. К пакетам данного типа относятся FineReader, CunieForm, Tiger™, OmniPage.

Разнообразными пакетами представлена группа *финансовых программ*: для ведения деловых записей типа записной книжки и расчета финансовых операций (баланс денежных средств, определение процентных выплат по займам и кредитам, временная структура денежных вложений и т.п.).

Для расчета величины налогов можно использовать программы TurboTax for Windows, Personal Tax Edge.

С помощью программ Quicken, DacEasy Accounting, Peachtree for Windows можно автоматизировать бухгалтерский учет. Эту же функцию выполняет ряд отечественных программ: «Турбобухгалтер», «1С: Бухгалтерия», «Бухгалтер» фирмы «Атлант-Информ» и др.

Для аналитических исследований используются хорошо зарекомендовавшие себя зарубежные статистические пакеты, такие, как StatGraphics или Systas, или отечественная разработка «Статистик-Консультант». Однако в коммерческих фирмах внедрение этих пакетов сдерживается отсутствием соответствующим образом подготовленных специалистов, высокой ценой пакетов и широким внедрением табличных процессов, с помощью которых можно провести простейшие, но наиболее часто используемые статистические расчеты.

7.3.3. ИНТЕГРИРОВАННЫЕ ППП

Интегрированные пакеты программ — по количеству наименований продуктов немногочисленная, но в вычислительном плане довольно мощная и активно развивающаяся часть ПО.

Идея создания интегрированных программных комплексов не нова и в той или иной мере была реализована во всех поколениях ЭВМ.

Внимание к этой проблеме объясняется как расширением сферы применения вычислительной техники, так и стремлением фирм-разработчиков программного обеспечения не «потерять» своих клиентов с переходом на более совершенные системы обработки данных.

Традиционные, или *полносвязанные*, интегрированные комплексы представляют собой многофункциональный автономный пакет, в котором в одно целое соединены функции и возможности различных специализированных (проблемно-ориентированных) пакетов, родственных в смысле технологии обработки данных на отдельном рабочем месте. Типичными представителями таких программ являются пакеты Framework, Symphony, а также пакеты нового поколения Microsoft Word, Lotus Works. В этих программах происходит интеграция функций редактора текстов, системы управления базами данных и табличного процессора. В целом стоимость такого пакета гораздо ниже суммарной стоимости аналогичных специализированных пакетов.

В рамках интегрированного пакета обеспечивается связь между данными, однако при этом сужаются возможности каждой компоненты по сравнению с аналогичным специализированным пакетом. Интерфейс более ранних программ был перегружен различными средствами обмена данными и описаниями среды работы, что требовало от пользователя определенных навыков и знаний в части переключения режимов пакета, форматов данных, принципов хранения и манипулирования различными типами данных, что в конечном счете снижало привлекательность паке-

тов. В современных пакетах (например, Microsoft Works) этот недостаток изжит: простота интерфейса позволяет применять его без предварительного обучения персонала.

В настоящее время активно реализуется другой подход интеграции программных средств: объединение специализированных пакетов в рамках единой ресурсной базы, обеспечение взаимодействия приложений (программ пакета) на уровне объектов и единого упрощенного центра-переключения между приложениями. Интеграция в этом случае носит *объектно-связанный характер*.

Типичные и наиболее мощные пакеты данного типа: Borland Office for Windows, Lotus SmartSuite for Windows, Microsoft Office. В профессиональной редакции этих пакетов присутствуют четыре приложения: текстовый редактор, СУБД, табличный процессор, программы демонстрационной графики. Целесообразность создания таких пакетов, очевидно, связана с желанием получить дополнительный эффект от интеграции по отношению к простой сумме составляющих его компонент. Этот эффект должен достигаться за счет согласованного взаимодействия компонент в процессе работы пользователя. При традиционном подходе к интеграции программ этот выигрыш может быть легко сведен на нет отсутствием нужной пользователю функции, присутствующей в специализированном пакете, и необходимостью в пусть небольшом, но дополнительном обучении.

Особенностью нового типа интеграции пакетов является использование общих ресурсов. Здесь можно выделить четыре основных вида совместного доступа к ресурсам:

- использование утилит, общих для всех программ комплекса. Так, например, утилита проверки орфографии доступна из всех программ пакета;

- применение объектов, которые могут находиться в совместном использовании нескольких программ;

- реализация простого метода перехода (или запуска) из одного приложения к другому;

- реализация построенных на единых принципах средств автоматизации работы с приложением (макроязыка), что позволяет организовать комплексную обработку информации при минимальных затратах на программирование и обучение программированию на языке макроопределений.

Совместное использование объектов с несколькими приложениями — краеугольный камень современной технологии интеграции программ и манипулирования данными. Разработаны два основных стандарта в этой области:

- динамической компоновки и встраивания объектов Object Linking and Embedding (OLE) 2.0 фирмы Microsoft;

- OpenDoc (открытый документ) фирм Apple, Borland, IBM, Novell и WordPerfect.

Механизм динамической компоновки объектов дает возможность пользователю помещать информацию, созданную одной прикладной программой, в документ, формируемый в другой. При этом пользователь может редактировать информацию в новом документе средствами того продукта, с помощью которого этот объект был создан (при редактировании автоматически запускается соответствующее приложение). Запущенное приложение и программа обработки документа-контейнера выводит на экран «согласованные» меню, часть пунктов которого принадлежит одной программе, а другая часть — другой.

- Кроме того, данный механизм позволяет переносить OLE-объекты из окна одной прикладной программы в окно другой.

- В этой технологии предусмотрена также возможность общего использования функциональных ресурсов программ: например, модуль построения графиков табличного процессора может быть использован в текстовом редакторе.

- Недостатком данной технологии является ограничение на размер объекта размером одной страницы.

- OpenDoc представляет собой объектно-ориентированную систему, базирующуюся на открытых стандартах фирм-участников разработки. В качестве модели объекта используется распределенная модель системных объектов (DSOM-Distributed System Object Model), разработанная фирмой IBM для OS/2. Предполагается совместимость между OLE и OpenDoc.

7.4. АВТОМАТИЗИРОВАННЫЕ РАБОЧИЕ МЕСТА

Деятельность работников сферы управления в настоящее время ориентирована на использование развитых информационных технологий.

Организация и реализация управленческих функций требует радикального изменения как самой технологии управления, так и технических средств обработки информации, среди которых главное место занимают персональные компьютеры. Они все более превращаются из систем автоматической переработки входной информации в средства накопления опыта управленческих работников, анализа, оценки и выработки наиболее эффективных экономических решений.

Тенденция к усилению децентрализации управления влечет за собой распределенную обработку информации с децентрализацией применения средств вычислительной техники и совершенствованием организации непосредственно рабочих мест пользователей.

Автоматизированное рабочее место (АРМ) можно определить как совокупность информационно-программно-технических ресурсов, обеспечивающую конечному пользователю обработку данных и автоматизацию управленческих функций в конкретной предметной области.

Создание АРМ предполагает, что основные операции по накоплению, хранению и переработке информации возлагаются на вычислительную технику, а экономист выполняет часть ручных операций и операций, требующих творческого подхода при подготовке управленческих решений. Персональная техника применяется пользователем для контроля производственно-хозяйственной деятельности, изменения значений отдельных параметров в ходе решения задачи, а также ввода исходных данных в АИС для решения текущих задач и анализа функций управления.

АРМ как инструмент для рационализации и интенсификации управленческой деятельности создается для обеспечения выполнения некоторой группы функций. Наиболее простой функцией АРМ является информационно-справочное обслуживание. Хотя эта функция в той или иной степени присуща любому АРМ, особенности ее реализации существенно зависят от категории пользователя.

АРМ имеют проблемно-профессиональную ориентацию на конкретную предметную область. Профессиональные АРМ являются главным инструментом общения человека с вычислительными системами, играя роль автономных рабочих мест. АРМ выполняют децентрализованную одновременную обработку информации на рабочих местах исполнителей в составе распределенной БД. При этом они имеют выход через системное устройство и каналы связи в ПЭВМ и БД других пользователей, обеспечивая таким образом совместное функционирование ПЭВМ в процессе коллективной обработки.

АРМ, созданные на базе персональных компьютеров, — наиболее простой и распространенный вариант автоматизированного рабочего места для работников сферы организационного управления. Такое АРМ рассматривается как система, которая в интерактивном режиме работы предоставляет конкретному работнику (пользователю) все виды обеспечения монопольно на весь сеанс работы. Этому отвечает подход к проектированию такого компонента АРМ, как внутреннее информационное обеспечение, согласно которому информационный фонд на магнитных носителях конкретного АРМ должен находиться в монопольном распоряжении пользователя АРМ. Пользователь сам выполняет все функциональные обязанности по преобразованию информации.

Создание АРМ на базе ПК обеспечивает:

- простоту, удобство и дружелюбность по отношению к пользователю;
- простоту адаптации к конкретным функциям пользователя;
- компактность размещения и невысокие требования к условиям эксплуатации;
- высокую надежность и живучесть;
- сравнительно простую организацию технического обслуживания.

Эффективным режимом работы АРМ является его функционирование в рамках локальной вычислительной сети в качестве *рабочей*

станции. Особенно целесообразен такой вариант, когда требуется «распределять» информационно-вычислительные ресурсы между несколькими пользователями.

Более сложной формой является АРМ с использованием ПЭВМ в качестве интеллектуального терминала, а также с удаленным доступом к ресурсам центральной (главной) ЭВМ или внешней сети. В данном случае несколько ПЭВМ подключаются по каналам связи к главной ЭВМ, при этом каждая ПЭВМ может работать и как самостоятельное терминальное устройство.

В наиболее сложных системах АРМ могут через специальное оборудование подключаться не только к ресурсам главной ЭВМ сети, но и к различным информационным службам и системам общего назначения (службам новостей, национальным информационно-поисковым системам, базам данных и знаний, библиотечным системам и т.п.).

Возможности создаваемых АРМ в значительной степени зависят от технико-эксплуатационных характеристик ЭВМ, на которых они базируются. В связи с этим на стадии проектирования АРМ четко формулируются требования к базовым параметрам технических средств обработки и выдачи информации, набору комплектующих модулей, сетевым интерфейсам, эргономическим параметрам устройств и т.д.

Обязательным условием функционирования АРМ является *техническое обеспечение*. Это обоснованно выбранный комплекс технических средств для их оснащения.

Средства обработки информации — вычислительные машины разных мощностей и типов — составляют основу технического обеспечения вычислительных сетей. Характерной особенностью практического использования технических средств в организационно-экономическом управлении в настоящее время является переход к децентрализованной и сетевой обработке на базе ПЭВМ.

Если ПЭВМ используется в качестве АРМ небольшой локальной сети, на котором централизованно хранится вся информация, необходимая для работы, объем обрабатываемой информации невелик. Скорость работы при этом определяется не быстродействием компьютера, а скоростью диалога оператора и машины. Отсюда вытекает, что в данном случае вполне приемлема ПЭВМ с небольшим быстродействием и минимальным объемом ОЗУ.

В другом случае, если компьютер предназначается для регулярной подготовки объемных документов и использует для этого большие массивы информации, необходима установка мощных машин с большим объемом внешней и внутренней памяти.

Информационное наполнение АРМ при определении круга пользователей и выяснении сущности решаемых ими задач осуществляет *информационное обеспечение* АРМ. В сфере организационного управления пользователи могут быть условно разделены на три категории:

руководители, персонал руководителей и обслуживающий персонал. Разрабатываемые АРМ для разных категорий пользователей отличаются видами представления данных. К примеру, обслуживающий персонал обычно имеет дело с внутренними данными организации, решает повторяющиеся задачи, пользуется, как правило, структурированной информацией. Руководителям требуются как внутренние, так и внешние данные для реализации цели управления или принятия решения.

Применение АРМ не должно нарушать привычный пользователю ритм работы. АРМ концентрируют внимание пользователя на логической структуре решаемых задач, а не на характеристике реализующей их программной системы. Однако если заданное системе действие не производится, пользователь должен знать причину, и информация об этом должна выдаваться на экран.

Математическое обеспечение АРМ представляет собой совокупность алгоритмов, обеспечивающих формирование резульатной информации. Математическое обеспечение служит основой для разработки комплекса прикладных программ.

В составе программного обеспечения (ПО) АРМ можно выделить два основных вида обеспечения, различающихся по функциям: общее (системное) и специальное (прикладное). К общему программному обеспечению относится комплекс программ, обеспечивающий автоматизацию разработки программ и организацию экономического вычислительного процесса на ПЭВМ безотносительно к решаемым задачам. Специальное программное обеспечение представляет собой совокупность программ решения конкретных задач пользователя.

В качестве операционных систем АРМ, созданных на базе 16-разрядных ПЭВМ, обычно используется MS DOS, на базе 32-разрядных — OS/2 и UNIX.

Основными приложениями пакетов прикладных программ, входящих в состав специального ПО АРМ, являются обработка текстов, табличная обработка данных, управление базами данных, машинная и деловая графика, организация человеко-машинного диалога, поддержка коммуникаций и работа в сетях.

Эффективными в АРМ являются многофункциональные интегрированные пакеты, реализующие несколько функций переработки информации, например табличную, графическую, управление базами данных, текстовую обработку в рамках одной программной среды.

Интегрированные пакеты удобны для пользователей. Они имеют единый интерфейс, не требуют стыковки входящих в них программных средств, обладают достаточно высокой скоростью решения задач.

Лингвистическое обеспечение АРМ включает языки общения с пользователем, языки запросов, информационно-поисковые языки, языки-посредники в сетях. Языковые средства АРМ обеспечи-

вают однозначное смысловое соответствие действий пользователя и аппаратной части в виде ПЭВМ. Одновременно языки АРМ должны быть пользовательско-ориентированными, в том числе профессионально-ориентированными.

Основу языков АРМ составляют заранее определяемые термины, описания способов установления новых терминов, списки правил, на основе которых пользователь может строить формальные конструкции, соответствующие его информационной потребности. Например, в одних АРМ данные и конструкции представляются в виде таблиц, в других — в виде операторов специального вида.

Языковые средства АРМ можно разделить по видам диалога. Средства поддержки диалога определяют языковые конструкции, знание которых необходимо пользователю. В одном АРМ может быть реализовано несколько типов диалога: иницируемый ЭВМ, с помощью заполнения шаблонов, с использованием меню, гибридный.

Организационное обеспечение АРМ включает комплекс документов, регламентирующих деятельность специалистов при использовании компьютера или терминала другого вида на рабочем месте и определяющих функции и задачи каждого специалиста.

Специалистом выполняются на АРМ следующие операции:

- ввод информации с документов при помощи клавиатуры (с визуальным контролем по экрану дисплея);
- ввод данных в ПЭВМ с магнитных носителей с других АРМ;
- прием данных в виде сообщений по каналам связи с других АРМ в условиях функционирования локальных вычислительных сетей;
- редактирование данных и манипулирование ими;
- накопление и хранение данных;
- поиск, обновление и защита данных;
- вывод на экран, печать, магнитный носитель резульатной информации, а также различных справочных и инструктивных сообщений пользователю;
- формирование и передача данных на другие АРМ в виде файлов на магнитных носителях или по каналам связи в вычислительных сетях;
- получение оперативных справок по запросам.

Методическое обеспечение АРМ состоит из методических указаний, рекомендаций и положений по внедрению, эксплуатации и оценке эффективности их функционирования. Оно включает в себя также организованную машинным способом справочную информацию об АРМ в целом и отдельных его функциях, средства обучения работе на АРМ, демонстрационные примеры.

Эргономическое обеспечение АРМ представляет собой комплекс мероприятий, обеспечивающих максимально комфортные условия использования АРМ специалистами. Это предполагает выбор специальной мебели для размещения техники АРМ, организацию картотек для хранения документации и магнитных носителей.

Одна из важнейших функций эргономического обеспечения АРМ — уменьшение отрицательных воздействий на человека со стороны ПЭВМ.

Правовое обеспечение АРМ — это система нормативно-правовых документов, определяющих права и обязанности специалистов в условиях функционирования АРМ. Эти документы строго увязаны с комплексом разработок, регламентирующих порядок хранения и защиты информации, правила ревизии данных, обеспечение юридической подлинности совершаемых на АРМ операций и т.д.

Эффективное функционирование АИС и АРМ базируется на комплексном использовании современных технических и программных средств обработки информации в совокупности с современными организационными формами размещения техники.

Выбор организационных форм использования программно-технических средств целесообразно осуществлять с учетом их расщепления по уровням иерархии управления в соответствии с организационной структурой автоматизируемого объекта. При этом основным принципом выбора является коллективное обслуживание пользователей, отвечающее структуре экономического объекта.

С учетом современной функциональной структуры территориальных органов управления совокупность программно-технических средств должна образовывать по меньшей мере трехуровневую глобальную систему обработки данных с развитым набором периферийных средств каждого уровня (рис. 7.6).

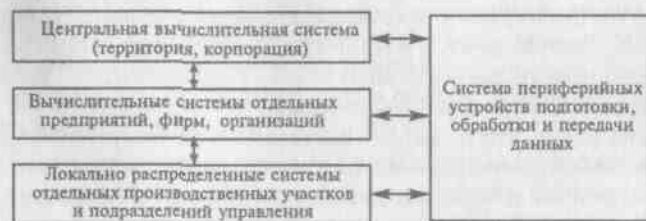


Рис. 7.6. Принципиальная схема многоуровневой организации программно-технических средств ЛВС

Первый уровень — центральная вычислительная система территориального или корпоративного органа, включающая одну или несколько мощных ЭВМ или мэйнфреймов. Ее главная функция — общий, экономический и финансовый контроль, информационное обслуживание работников управления.

Второй уровень — вычислительные системы предприятий (объединений), организаций и фирм, которые включают мэйнфреймы, мощные ПЭВМ, обеспечивают обработку данных и управление в рамках структурной единицы.

Третий уровень — локально распределенные вычислительные сети на базе ПЭВМ, обслуживающие производственные участки нижнего уровня. Каждый участок оснащен собственной ПЭВМ, которая обеспечивает комплекс работ по первичному учету, учету потребности и распределения ресурсов. В принципе это может быть АРМ, выполняющее функциональные вычислительные процедуры в рамках определенной предметной области.

В то же время на каждом уровне иерархии управления имеют место три способа организации технических средств: централизованный, децентрализованный и иерархически распределенный. Первый способ предполагает выполнение всех работ по обработке данных, начиная со сбора и регистрации данных, в одном центре обработки; второй предусматривает предварительную обработку информации, которая не требует создания очень крупных массивов данных, на периферийном оборудовании удаленного пользователя низовых звеньев экономического объекта; при третьем способе технология обработки оптимально распределена по уровням управления системы.

7.5. ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Искусственный интеллект (ИИ) как научное направление, связанное с попытками формализовать мышление человека, имеет длительную историю. Еще Платон, Аристотель, Р. Декарт, Г.В. Лейбниц, Дж. Буль и многие другие исследователи на уровне современных им знаний стремились описать мышление как совокупность некоторых элементарных операций, правил и процедур. Качественно новый период развития ИИ связан с появлением в научных лабораториях ЭВМ и публикацией книги Н. Винера «Кибернетика или управление и связь в животном и машине».

Первые шаги кибернетики были направлены на изучение и осмысление процессов, протекающих в сложных, прежде всего живых системах, включая и мыслящие. Исследования имели ярко выраженный познавательный характер. Но уже тогда стали появляться разработки, направленные на воспроизведение в ЭВМ определенных процессов и феноменов мышления. Позднее именно это направление работ и оформилось в самостоятельную область, разрабатывающую проблему ИИ.

В ходе последующего развития исследований по ИИ, в 60-х — 70-х годах, произошло их разделение на два самостоятельных направления. Это разделение сохраняется и до настоящего времени. Без четкого понимания различий и тенденций развития обоих направлений нельзя правильно разобраться и в современном состоянии проблемы.

7.5.1. НАПРАВЛЕНИЕ ИССЛЕДОВАНИЙ В ОБЛАСТИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Разделение работ по ИИ на два направления связано с существованием двух точек зрения на вопрос о том, каким именно образом строить системы ИИ. Сторонники одной точки зрения убеждены в том, что важнее всего результат, т.е. хорошее совпадение поведения искусственно созданных и естественных интеллектуальных систем, а что касается внутренних механизмов формирования поведения, то разработчик ИИ вовсе не должен копировать или даже принимать во внимание особенности естественных, живых аналогов.

Другая точка зрения заключается в том, что именно изучение механизмов естественного мышления и анализ данных о способах формирования разумного поведения человека могут создать основу для построения систем ИИ, причем построение это должно осуществляться прежде всего как моделирование, воспроизведение техническими средствами принципов и конкретных особенностей функционирования биологических объектов.

Таким образом, первое направление рассматривает продукт интеллектуальной деятельности человека, изучает его структуру (выделяя различные проявления интеллектуальной деятельности — решение задач, доказательство теорем, игры) и стремится воспроизвести этот продукт средствами современной техники, т.е. ЭВМ. Если удастся запрограммировать ЭВМ так, чтобы она успешно решала конкретную задачу, то считают, что соответствующий вид интеллектуальной деятельности автоматизирован. Ясно, что успехи этого направления ИИ тесно связаны с развитием ЭВМ и искусством программирования, т.е. с комплексом научно-технических исследований, называемым компьютерными науками. Это направление ИИ также часто называют машинным интеллектом.

Второе направление ИИ рассматривает данные о нейрофизиологических и психологических механизмах интеллектуальной деятельности, а в более широком плане — разумном поведении человека. Разработчики стремятся воспроизвести эти механизмы с помощью технических устройств, чтобы поведение их хорошо совпадало с поведением человека в определенных, заранее задаваемых пределах. При положительном решении этой проблемы считают, что соответствующий вид человеческой деятельности автоматизирован. Развитие этого направления, называемого искусственным разумом, тесно связано с успехами наук о человеке. Характерным в данном случае является стремление к воспроизведению более широкого, чем в машинном интеллекте, спектра проявлений разумной деятельности человека.

Оба основных направления ИИ связаны с моделированием. В первом случае с моделированием феноменологическим, имитационным, а во вто-

ром — со структурным. В упрощенном виде структура основных направлений, существующих в ИИ, изображена на рис. 7.7.



Рис. 7.7. Основные направления исследований в области искусственного интеллекта

7.5.2. МАШИННЫЙ ИНТЕЛЛЕКТ И РОБОТОТЕХНИКА

Основу всякого робота составляет его «тело» (рис. 7.8, блок 7), т.е. совокупность механических, электромеханических, пневматических и т.п. устройств, обеспечивающих его прочностные и несущие свойства (шасси, кожухи), способность перемещаться в пространстве (колеса, гусеницы, «ноги» и их приводы) и воздействовать на объекты внешнего мира (эффекторы — манипуляторы, толкатели, захваты). На шасси располагают устройства восприятия информации из среды (сенсоры), устройства управления и различные вспомогательные устройства, рассматриваемые как физические объекты. В зависимости от назначения робота состав его «тела» может изменяться.

Непременным компонентом робота является система управления, которую называют «жесткой» (блок 6). Она содержит неизменяемую программу или логические устройства, которые управляют работой приводов движителей и эффекторов, и может быть также представлена совокупностью выключателей, упоров и т.п. В тех случаях, когда робот

снабжен более развитой системой управления, «жесткая» система составляет ее нижний уровень.



Рис. 7.8. Структурная схема робота с элементами искусственного интеллекта

Система восприятия (блок 1) включает датчики, контролирующие состояние «тела» робота, и параметры внешней ситуации, в которой он находится. Сюда могут входить также развитые системы технического зрения, содержащие оптические, ультразвуковые, телевизионные и т.п. устройства. Все они составляют набор сенсоров робота. Система восприятия включает также средства предварительной обработки информации, поступающей от сенсоров, и средства интерпретации этой информации. Под интерпретацией понимают «перевод» языка выходных сигналов сенсоров на язык описания объектов и состояний внешней среды. Интерпретация осуществляется с помощью знаний о внешнем мире и «теле» робота, хранящихся в блоке 2. Интерпретированные сообщения предъявляются человеку-оператору, принимающему участие в управлении роботом (связь с блоком 3), или могут использоваться собственной системой управления (связи с блоками 4 и 5).

Устройство связи робота с человеком-оператором — диалоговый процессор — предназначен для выполнения ряда функций. Одна из них состоит в восприятии и предварительной обработке входных высказываний, поступающих от оператора. С помощью высказываний оператор может ставить перед роботом задачи, управлять процессом их решения,

корректировать действия робота и т.п. Специальные высказывания могут быть адресованы системе восприятия и управлять режимом ее работы (включение/выключение отдельных сенсоров, задачи на поиск нужных признаков, объектов). Высказывания могут быть речевыми или иметь текстовую форму. В последнем случае они представляются, например, в виде последовательности рукописных символов, пробивок на перфоленте или воздействий на специальную клавиатуру. Речевые высказывания могут производиться на некотором более или менее ограниченном подмножестве естественного языка.

Функция предварительной обработки состоит в осуществлении морфологического и синтаксического анализа входных высказываний, выполняемого на основе знаний об используемом языке (блок 2).

Следующая функция диалогового процессора связана с семантическим анализом высказываний. В ходе этого анализа высказывание интерпретируется в некий внутренний язык робота, понятный его системам управления, и на этом языке формулируется задача или сообщение, смысл которых должен полностью совпадать со смыслом воспринятого высказывания. Так осуществляется «понимание» роботом человека. Полнота этого понимания может быть проконтролирована лишь по конечным результатам действий робота.

Семантический анализ проводится на основе знаний, представленных в блоке 2. При этом должны быть использованы знания не только о внешнем (физическом) мире и о самом роботе, но и знания о мире задач человека, его целях, способах их выражения, характерных умолчаниях и т.п. Так, приказание «принеси круглый предмет» не должно приводить к попыткам робота рассмотреть в качестве круглого предмета голову оператора.

Полнота реализации описанных функций зависит от назначения робота и уровня используемого языка управления. Если, например, указания формируются в терминах команд для блока 6, диалоговый процессор может быть редуцирован до клавиатуры управления.

Еще одна функция диалогового процессора состоит в интерпретации и выдаче сообщений и запросов, формируемых системами восприятия, знаний и управления. Интерпретация заключается в переводе сообщений с внутреннего языка робота на язык систем вывода. В качестве последних могут использоваться синтезаторы речи, видеотерминалы, графопостроители и другие устройства.

Собственно система управления робота, точнее ее высшие уровни, представлены планировщиком (блок 5) и системой принятия решений, или решателем (блок 4). Задача планировщика заключается в том, чтобы на основе соответствующих данных осуществить автоматическое решение задачи, формальное описание которой поступает из диалогового процессора. В простых случаях планировщик выступает в роли трассировщика, определяющего оптимальную или близкую к ней траекто-

рию (программу) перемещения в пространстве рабочих органов робота или самого робота. В более сложных случаях планировщик выполняет сложную обработку информации, представленной не только в цифровой, но и в символической форме, и вырабатывает обобщенные планы достижения поставленной цели.

Решатель обычно выполняет те же функции, что и планировщик. Различие заключается в работе на разных уровнях детализации задач. Уровень планировщика можно назвать стратегическим, а уровень решателя — тактическим. При решении своих задач планировщик использует информацию об общем характере среды, примерном расположении препятствий и т.п. Такая информация хранится преимущественно в блоке 2 и играет роль карты рабочего пространства. Решатель же использует в основном данные о текущей ситуации, поступающие от системы восприятия, и формирует конкретные команды управления, последовательность которых должна обеспечить выполнение текущих этапов плана, синтезированного планировщиком.

Если выполнение плана приводит к неудаче, например, робот наталкивается на непреодолимое препятствие и очередная команда не может быть выполнена, решатель распознает это либо по данным системы восприятия, либо по сигналам с «жесткого» уровня управления и формирует соответствующее сообщение для планировщика. Последний либо решает задачу заново, синтезируя новый план, либо через диалоговый процессор передает сообщение оператору и ожидает дальнейших указаний.

Каждый из рассмотренных блоков реализует довольно сложный комплекс функций, решая при этом свою задачу. Проблемы, связанные с разработкой теоретических принципов и технических средств построения каждого из блоков, и являются основными проблемами робототехники. Все они тесно связаны с идеями и методами искусственного интеллекта.

Типология роботов. Многие из уже существующих и разрабатываемых в настоящее время роботов существенно различаются как по структуре, так и по функциональным возможностям. Различают три поколения роботов. Эта классификация основана на их структурных различиях и в то же время отражает хронологию развития робототехники.

Первыми были созданы роботы, включающие только блоки 6 и 7 (см. рис. 7.8) Они названы *роботами первого поколения*. К ним относятся практически все известные в настоящее время промышленные роботы. *Роботы второго поколения*, или *очувствленные роботы*, включают блоки 1, 4, 6 и 7. Например, система «глаз—рука». Отметим, что ранние представители этого поколения обладали довольно примитивными сенсорами, а их системы принятия решений реализовывали простые переключения «жестких» программ. В дальнейшем они значительно усложнились, однако наиболее совершенные роботы этого класса существуют пока лишь в лабораториях. *Роботы третьего поколения*, или *роботы с*

искусственным интеллектом (блоки 1 — 7), находятся в настоящее время в стадии разработки. Одной из основных проблем их создания является проблема знаний.

В ряде случаев оказывается удобным пользование не структурной, а функциональной классификацией, поскольку она указывает, что робот может делать и для чего применяться. Рассмотрим такую классификацию, используя понятие степени интеграции функций робота.

Развитые, функционально завершенные системы, являющиеся компонентами роботов разного типа, могут быть разделены на пять основных групп:

группа В — системы восприятия зрительной, слуховой, тактильной и других типов информации о внешнем мире;

группа М — системы воздействия на объекты внешнего мира — различные манипуляторы, толкатели и т. п.;

группа Т — системы, обеспечивающие перемещение робота (колесные, гусеничные, шагающие, плавающие, летающие платформы и аппараты);

группа П — системы целеполагания и планирования действий робота, системы решения задач;

группа Р — системы, обеспечивающие коммуникацию робота с человеком-оператором и другими роботами на языках общения различных уровней, вплоть до естественного.

7.5.3. ИНТЕЛЛЕКТУАЛЬНЫЕ РОБОТЫ

Успехи в развитии средств вычислительной техники обусловили преимущественное внимание к исследованиям в области машинного интеллекта. Результаты этих исследований широко освещены в научных и научно-популярных изданиях. Рассмотрим значение их для решения проблем, связанных с созданием и использованием интеллектуальных роботов.

Планирование. Практически одновременно с появлением ЭВМ первого поколения в ИИ начали разрабатывать программы, решающие головоломки, играющие в различные игры и доказывающие теоремы. Для робототехники особую роль сыграло развитие теории и техники автоматического доказательства теорем, в частности разработка машинно-ориентированной предикативной логики.

Наиболее известной системой планирования, использующей технику доказательств, является система STRIPS, разработанная для управления действиями самоходного аппарата-робота. Этот робот мог передвигаться в комнатах, подходить к имеющимся объектам, толкать их, проходить через двери и т.п. Составляемые системой планы состоят из шести действий. Созданная в 1971 г. система оказала значительное влияние на дальнейшее развитие работ в данной области.

Машинное зрение. Особое внимание в исследованиях по машинному интеллекту уделяется проблеме распознавания образов. Наиболее развитыми в робототехнике являются методы распознавания зрительных образов. Алгоритмы, реализующие эти методы, являются основной частью систем машинного, или технического, зрения (СТЗ). Источником информации для них являются различные оптические системы, видеокамеры и т. п. Основные задачи, решаемые СТЗ, можно разделить на два класса: инспекцию и идентификацию. Задачи инспекции заключаются в проверке наличия объектов, обнаружении дефектов и т.п. Типичными задачами идентификации являются определение позиций известных объектов, выделение отдельных объектов в случаях, когда они соприкасаются, перекрываются или лежат «навалом», определение похожести объектов и т. п.

В промышленности стоимость операций технического контроля в среднем составляет 10% от общей стоимости продукции, поэтому созданию инспекционных СТЗ уделяется значительное внимание. Примерно 30% всех СТЗ применяются для идентификации объектов. В тех случаях, когда СТЗ входят в состав роботов высокой степени интеграции, они используются как источник информации при управлении позиционированием деталей, сборкой, сваркой и т. п.

Речевое общение. В ИИ существует устойчивый интерес к изучению и воспроизведению различных аспектов речевого поведения. Работы в этой области группируются вокруг задач автоматического перевода, реферирования текстов, построения справочных и информационно-поисковых устройств и, наконец, удобных языков общения человека с машиной. Последняя задача в настоящее время привлекает наибольшее внимание, ибо ее решение позволило бы кардинальным образом изменить характер использования ЭВМ специалистами, работающими в предметных областях и не владеющими языками и навыками программирования. Необходимость в таком изменении остро назрела в связи с процессами компьютеризации науки, образования и народного хозяйства в целом.

Системы речевого общения используются при создании роботов класса Р, получивших название роботов связи. Такие роботы могут входить также в состав интегральных роботов, выполняя функции блока 3 (см. рис. 7.8.).

Функциональные возможности современных анализаторов и синтезаторов речи проиллюстрируем на примере разработанной в ФРГ диалоговой системы, предназначенной для приема по телефону номеров различных служб (погоды, медпомощи). По названному телефонному номеру обеспечиваются услуги выдачи адресов, маршрутов проезда и т.п. Используемый в системе анализатор выдает до трех гипотез каждого воспринятого слова и система просит утвердить одну из гипотез при помощи слов «да» и «нет». Если все три гипотезы отвергаются абонентом, диалог заканчивается сообщением «распознавание невозможно».

Для робототехники наибольший интерес представляют анализаторы речи, которые используются в составе диалогового процессора (см. рис. 7.8.) для ввода в систему управления робота устных команд и сообщений.

Программирование роботов. Параллельно с работами по созданию систем речевого общения разрабатываются языки управления роботами. Строятся они по тем же принципам, что и обычные алгоритмические языки высокого уровня, и отличаются лишь возможностью полнее учитывать специфику задач управления роботами, особенности их структуры и т. п. При использовании таких языков диалоговый процессор (блок 3 на рис. 7.8.) может быть представлен ЭВМ, осуществляющей интерпретацию поступающих на нее команд.

Языки программирования роботов разделяют на три основных уровня. *Языки первого уровня* содержат команды, явно задающие необходимые движения робота. Интерпретируя эти команды, диалоговый процессор адресует их непосредственно «жесткому» уровню управления (блок 6 на рис. 7.8.). При программировании робота на языках *второго уровня* пользователь определяет взаимосвязь того, что робот должен будет делать, с тем, что он в это время будет воспринимать. В данном случае диалоговый процессор передает команды некоему решателю (блок 4), который реализуется, как правило, в виде программной системы. Ко второму уровню относятся языки VAL-11 и AML. *Языки третьего уровня* дают возможность пользователю программировать действия робота путем указания желаемого эффекта их воздействия на объект. При этом оказывается необходимым наличие в системе управления планировщика (блок 5), работа которого может быть основана, например, на технике доказательства теорем. Одним из ранних языков программирования этого уровня является AUTOPASS, обеспечивающий нахождение безопасного пути среди многоугольных объектов. Язык RAPT, разработанный Р. Дж. Попплстоном, ориентирован на учет пространственных отношений объектов в рабочем пространстве. Предложены и другие языки третьего уровня.

7.5.4. МОДЕЛИРОВАНИЕ БИОЛОГИЧЕСКИХ СИСТЕМ

В большей части исследований в области ИИ непосредственным объектом моделирования являются структуры и процессы в нервной системе человека и животных. При модельном подходе к изучению нервной системы в поле зрения исследователя в первую очередь оказываются отдельные нервные клетки — нейроны и структуры из взаимосвязанных клеток — нейронные сети.

Нейроноподобные сети. Кора больших полушарий головного мозга человека содержит около 14 млрд. нейронов. Их короткие и длинные отростки — дендриты, по которым поступают входные воздействия, и аксоны, отводящие выходные реакции, образуют сложнейшее перепле-

тение связей. Устройство и законы функционирования самого нейрона также очень сложны. Поэтому при моделировании нейронов пользуются упрощенным описанием. Такие упрощенные модели нейронных сетей называют нейроноподобными сетями.

Различают два типа нейроноподобных сетей. В первом из них узлами сети являются формальные элементы, описывающие отдельные нейроны. К сетям такого типа относятся широко известные нейроноподобные сети, разработанные и исследованные У. Маккалоком и У. Питтсом. В сетях второго типа узлами являются формальные элементы, соответствующие не отдельным нейронам, а особым их совокупностям — нейронным ансамблям. Под нейронным ансамблем понимается такая совокупность взаимосвязанных нейронов, которая возбуждается полностью при возбуждении некоторой ее части. Многие исследователи полагают, что именно нейронный ансамбль, а не отдельный нейрон является функциональной единицей мозга как системы, обеспечивающей сложную приспособительную деятельность человека или животного.

Как объект модельного описания нейронный ансамбль отличается от отдельного нейрона двумя основными особенностями. Одна из них состоит в том, что выходное возбуждение ансамбля изменяется непрерывно, а не по закону «да — нет». Ансамбль соответственно может быть описан как нелинейный преобразователь аналоговой информации, задаваемый набором определенных статических и динамических характеристик. Другая особенность заключается в том, что ансамбль может быть поставлен в соответствие некоторой содержательной единице — понятию, образу и т.п. элементу, принимающему участие в процессе мыслительной деятельности. Таким образом, нейроноподобная сеть, узлы которой соответствуют ансамблям, становится сетью с семантикой, семантической сетью. Это обстоятельство во многом изменяет подход к проблемам синтеза сетей и содержательной интерпретации протекающих в них процессов.

Исследования нейроноподобных сетей, конструируемых как на уровне отдельных нейронов, так и их ансамблей, активно проводятся и при решении задач робототехники. Первые роботы, системы управления которых были построены на основе нейроноподобных сетей (на уровне отдельных нейронов), были разработаны в 60-х годах Л. Сутро, У. Килмером, Дж. Олбусом и др. Эти разработки имели бионическую направленность: при синтезе сетей использовались нейрофизиологические данные о взаимодействии различных отделов мозга позвоночных.

Результаты исследований в области моделирования нейронных сетей существенно расширили класс задач, решаемых при помощи нейроноподобных сетей. Теперь в этот класс включаются комбинаторные, оптимизационные и другие задачи. Успехи микроэлектроники подготовили технологическую базу для создания вычислительных устройств, способных осуществлять параллельную обработку информации. Два эти фак-

тора обусловили появление нейрокомпьютеров — ЭВМ, архитектура которых наилучшим образом приспособлена для решения задач моделирования нейронных сетей. В настоящее время нейрокомпьютеры создаются в виде компактных приставок к персональным ЭВМ, существенно увеличивая их функциональные возможности.

7.5.5. ЭВРИСТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ И МОДЕЛИРОВАНИЕ

Эвристическое программирование. В отличие от моделирования на уровне нейронных сетей, эвристическое программирование исследует другой уровень организации поведения, называемый *операционным*. На этом уровне поведение рассматривается как последовательность мыслительных, может быть не всегда осознаваемых человеком операций, выполнение которых приводит к успешному решению той или иной задачи.

Обычная процедура построения моделей методом эвристического программирования строится следующим образом. Испытуемому предлагается решать некоторую задачу, сопровождая свои размышления устными комментариями хода своих рассуждений. Все высказывания испытуемых тщательно протоколируются. Затем протоколы подвергают анализу с целью выявления хода решения, характера применяемых операций, догадок, приемов и т.п. Полученный в ходе анализа материал используется при составлении компьютерной программы — модели данного вида поведения. Таким образом, программа является моделью не испытуемого, а протокола. Такая модель должна выполнять то, что делает испытуемый, и так, как это делает он.

Следующий этап процедуры связан с исследованием работы модели при решении задач того же типа, которые предлагались испытуемому. Если процесс решения отклоняется от зафиксированного в протоколе, программа дорабатывается. Для этого могут быть поставлены новые эксперименты с испытуемыми, получены новые протоколы и т. д.

Использование описанной методики связано с рядом затруднений. При самоотчете испытуемый не в состоянии выделить и прокомментировать все без исключения шаги выполненного им решения, особенно если они связаны с догадками, неосознаваемыми заключениями и т.п. Соответственно такие шаги не попадают в протокол, и исследователь должен самостоятельно заполнять разрывы, опираясь на собственные знания и догадки. Поэтому правильнее будет говорить, что окончательная программа моделирует гипотезу исследователя об изучаемом процессе — гипотезу, основанную на протоколе и определенных данных из области психологии. Ряд трудностей заключается также в проведении анализа и интерпретации протокольных записей. Здесь необходима ти-

пизация приемов анализа и разработка мер, ограничивающих произвольность интерпретаций.

Развитие эвристического программирования связано с построением разнообразных моделей, таких, как модели поведения при выборе, при определении стратегии размещения ценных бумаг и др. Однако наибольшие успехи теоретического и прикладного характера достигнуты при создании систем ИИ, называемых решателями задач.

Разработка программ — решателей задач преследует две основные цели: во-первых, являясь моделью поведения человека в определенных условиях, решатель обладает некоторой объяснительной силой и может быть использован для предсказания действий человека в ходе решения; во-вторых, решатель может быть использован как составная часть в системах автоматизации управления сложными объектами, в частности — роботами.

Наряду с эвристическим программированием в области работ по созданию искусственного разума оформилось еще одно направление исследований, получившее название эвристического моделирования.

Эвристическое моделирование. Пусть имеется некий комплекс поведенческих актов, поведенческая функция, или поведение F , и для решения некоторой практической задачи требуется построить модель этого поведения. Исследователь, работающий на нейронном уровне, попытается найти нейронную структуру, ответственную за формирование F , изучить ее и построить модель этой структуры S . Функционирование модели S и должно воспроизводить поведение F . Недостатки схемы $S \rightarrow F$ связаны с тем, что нейронный уровень описания слишком далек от поведенческого и построить модель S удается, как правило, только в тех случаях, когда функция F достаточно проста.

Исследователь, пользующийся приемами эвристического программирования, подойдет к задаче иным образом. Он исследует протоколы экспериментов с испытуемыми, реализующими поведение F , построит гипотезу о структуре их мыслительных операций и воплотит эту гипотезу в модели P — модели процессов формирования поведения F . Уровень процессов ближе к уровню поведения, поэтому схема $P \rightarrow F$ позволяет строить модели сравнительно сложных форм поведения. Но и здесь эта сложность ограничена способностью испытуемого описывать ход своих рассуждений и способностью исследователя корректно интерпретировать протоколы.

Недостатки обоих подходов и пытаются преодолеть эвристическое моделирование. Пусть поведение F весьма сложно. Действия исследователя, работающего в области эвристического моделирования, будут строиться следующим образом. Прежде всего он изучит нейронные механизмы, ответственные за формирование F . Из-за сложности F ему скорее всего не удастся локализовать отдельные нейронные структуры, ответственные за F , и построенная структура будет описывать лишь неко-

торые наиболее существенные свойства нейронного аппарата. При построении такой гипотезы будут широко использованы данные нейрофизиологии и физиологии мозга. Затем будет построена модель гипотезы S . Однако модель еще не будет пригодна для воссоздания F , она окажется лишь аппаратом для решения этой задачи.

Следующий этап работы связан с изучением процессов формирования F . Здесь исследователь, опираясь на методики эвристического программирования, попытается построить конструктивную гипотезу о структуре процессов. Однако сложность поведения F сделает построение такой гипотезы невозможным. (Если этого не случится, то для решения задачи использовать эвристическое моделирование вообще не нужно: возможностей эвристического программирования вполне достаточно.) Гипотеза, следовательно, будет построена не столько на основе экспериментальных исследований, сколько на базе данных тех разделов психологии, в компетенцию которых входит изучение форм поведения, характерных для F . В настоящее время психология является преимущественно качественной наукой. Поэтому и модель P построенной гипотезы будет иметь качественный характер, т. е. модель P будет концептуальной моделью.

Итак, теперь имеем ситуацию, в которой, с одной стороны, известно, что именно вложить в разрабатываемую модель F (это «что» зафиксировано в модели P), а с другой — известно, как именно строить модель F (это «как» представлено моделью S , составляющей аппарат моделирования). Очередной этап работы и состоит в том, что исследователь реализует P с помощью S . Созданная модель представляет собой разновидность нейроноподобной сети и, как все такие сети, может быть реализована либо в виде компьютерной программы, либо в виде специализированного физического устройства. В последнем случае наиболее полно проявляются достоинства нейроноподобных сетей, связанные с их способностью осуществлять параллельные процессы переработки информации.

Созданная модель может быть подвергнута экспериментальному исследованию, в ходе которого она должна демонстрировать поведение, близкое к F . Однако, продвигаясь по цепочке $S \rightarrow P \rightarrow F$, исследователь вынужден сделать целый ряд эвристических предположений. Поэтому можно обнаружить, что между поведением модели и желаемым поведением F имеются существенные различия. Анализируя их, исследователь корректирует исходные гипотезы, вносит соответствующие изменения в модели S и P , вновь выделяет различия и в ходе таких последовательных приближений доводит модель до приемлемой по условиям задачи степени адекватности.

Таково довольно схематичное описание процедуры эвристического моделирования. Подводя итог, отметим следующее. Эвристическое моделирование ориентировано на построение моделей сложных форм поведения. Оно вооружило исследователя инструментом проверки развитых, больших гипотез, интегрирующих знания из разных областей наук

о человеке. До недавнего времени гипотезы такого рода могли существовать лишь в виде концептуальных конструкций и качественных описаний. В то же время процедура эвристического моделирования достаточно сложна и трудоемка; ее осуществление требует иногда проведения длительных циклов исследований. Поэтому на практике часто используются упрощенные, редуцированные схемы построения моделей.

7.5.6. СИСТЕМА ЗНАНИЙ

Одной из ключевых проблем создания ИИ является проблема представления и использования знаний. Ее разработка осуществляется различными направлениями ИИ.

Проблемы представления и использования знаний о мире, в котором функционирует ИИ, являются основными при построении его системы управления. Важность проблематики построения систем знаний была в должной мере осознана сравнительно недавно — в начале 70-х годов, когда получили широкое развитие исследования по проблемам знаний в ИИ.

В чем же сущность этих проблем и какова их специфика? В области ИИ понятие о знаниях сформировалось в ходе исследований по созданию принципов и техники работы с большими объемами данных и по построению баз данных (БД). Эффективность БД во многом зависит от того, каким именно способом организовываются, структурируются данные в памяти ЭВМ. До недавнего времени основную роль в этом играли формальные характеристики данных: принадлежность их некоторой табличной рубрике, вхождение в одну тематическую группу и т. д. Однако эффективность БД может быть существенно повышена, если связывать хранящуюся информацию не за счет форм тех или иных документов (таблиц, списков), а за счет тех отношений, которые существуют между фактами в объекте управления или в естественной среде. И отношения эти должны быть не случайными, ситуативными, а отражать существенные связи объекта, его природу, т. е. возникла необходимость отображения в БД знаний об объекте. Такие БД стали называть *интеллектуальными базами данных* или *базами (системами) знаний*. Идеология создания систем знаний (СЗ) в основном связана с формализацией семантической памяти, точнее — некоторых ее моделей, разработанных в когнитивной психологии. Таковы модели семантических сетей, фреймов и др. В то же время предпринимаются и попытки построения СЗ на основе средств логико-символических преобразований, изучаемых в ИИ длительное время и для которых развит хороший математический аппарат.

Итак, каждая СЗ является математической моделью некоторой области прикладного, неформализованного знания. Система понятий и отношений такой модели отображает систему понятий и отношений прикладного знания, а зависимости, существующие в модели, аппрокси-

мируют соответствующие зависимости в нем. Разработанные модели должны быть зафиксированы в памяти ЭВМ и использоваться для решения прикладных задач.

Создание СЗ предполагает решение следующих взаимосвязанных проблем. Прежде всего необходимо формализовать соответствующую область прикладного знания. Это трудная задача, поскольку решается она вручную и требует совместной работы специалистов-прикладников и математиков. Для проведения формализации требуется выбрать или построить концептуальную схему модели. Разработка методологии всех этих операций и составляет содержание первой проблемы — *проблемы формализации*. Вторая проблема — *проблема представления знаний* — связана с разработкой формального аппарата для описания способов их фиксации в памяти ЭВМ. Разработка теории вычислений и других преобразований, проводимых в построенных моделях, составляет третью проблему — *проблему использования знаний*. И наконец, четвертая, технологическая проблема, решением которой занимаются системные программисты, — это *проблема разработки средств программной поддержки моделей*, т. е. создания баз знаний и систем управления ими. Основное внимание в ИИ уделяется второй и третьей из перечисленных проблем, причем ведущая роль отводится проблеме представления. На практике ее разрабатывают совместно с вопросами построения концептуальных схем моделей знаний, и многие полагают, что именно эта проблема является основной для современного ИИ.

К настоящему времени в области разработки СЗ достигнуты значительные успехи. Полученные результаты послужили толчком к созданию полезных и интересных систем нового класса, имеющих широкое практическое применение, — экспертных систем, которые могут быть использованы в качестве советчиков и консультантов в самых разных сферах человеческой деятельности.

7.5.7. МОДЕЛИ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Любая предметная область характеризуется своим набором понятий и связей между ними, своими законами, связывающими между собой объекты данной предметной области, своими процессами, событиями. И конечно, каждая предметная область имеет свои, специфические методы решения задач. Знания о предметной области и способах решения в ней задач весьма разнообразны. Возможны различные классификации этих знаний. Наиболее часто знания подразделяются на декларативные и процедурные.

Процедурные знания описывают последовательности действий, которые могут использоваться при решении задач. Это, например, программы для ЭВМ, словесные записи алгоритмов, инструкция по сборке некоторого изделия.

Декларативные знания — это все знания, не являющиеся процедурными, например статьи в толковых словарях и энциклопедиях, формулировки законов в физике, химии и других науках и т.п. В отличие от процедурных знаний, отвечающих на вопрос: «Как сделать X?», декларативные знания отвечают, скорее, на вопросы: «Что есть X?» или «Какие связи имеются между X и Y?», «Почему X?» и т.д.

Интеллектуальные системы — это сложные программно-аппаратные комплексы, обязательно включающие в свой состав ЭВМ. Чтобы ввести знания о предметной области в ЭВМ, необходимо представить их в такой форме, которая была бы понятна машине. Иными словами, знания надо написать на языке, понятном ЭВМ, как понятны ей записи на языках программирования. Для это существуют специальные языки представления знаний. Их можно разделить на типы по тем формальным моделям представления знаний, которые лежат в их основе. Таких моделей четыре: логическая, сетевая, фреймовая и продукционная.

Логическая модель представления знаний. Она представляет собой формальную систему — некоторое логическое исчисление. Все знания о предметной области описываются в виде формул этого исчисления или правил вывода. Описание в виде формул дает возможность представить декларативные знания, а правила вывода — процедурные знания.

Сетевая модель представления знаний. Более наглядными являются языки, опирающиеся на сетевую модель представления знаний. В основе такой модели лежит идея о том, что любые знания можно представить в виде совокупности объектов (понятий) и связей (отношений) между ними.

Фреймовая модель представления знаний. Фреймы используются в системах искусственного интеллекта (например, в экспертных системах) как одна из распространенных форм представления знаний. Фрейм — это минимально возможное описание сущности какого-либо явления, события, ситуации, процесса или объекта. (Минимально возможное означает, что при дальнейшем упрощении описания теряется его полнота, оно перестает определять ту единицу знаний, для которой оно предназначено.) Фрейм имеет почти однородную структуру и состоит из стандартных единиц, называемых слотами. Каждая такая единица — слот — содержит название и свое значение. Изображается фрейм в виде цепочки:

Фрейм = <слот 1> <слот 2>... <слот N>.

Продукционная модель представления знаний. Продукция — один из распространенных в интеллектуальных системах способов представления знаний. Основу модели составляют системы продукций. Каждая продукция в наиболее общем виде записывается как стандартное выражение следующего вида:

«Имя продукции»:
Имя сферы;
Предусловие;
Условие ядра;
Если А, то В;
Постусловие.

Основная часть продукции — ее ядро имеет вид: «Если А, то В», где А и В могут иметь разные значения. Остальные элементы, образующие продукцию, носят вспомогательный характер. В наиболее простом виде продукция может состоять только из имени (например, ее порядкового номера в системе продукций) и ядра.

Рассмотренные модели представления знаний широко используются в современных интеллектуальных системах и прежде всего в экспертных системах. Каждая из форм представлений знаний может служить основой для создания языка программирования, ориентированного на работу со знаниями. Такими, например, языками являются язык FRL (Frame Representation Language), основанный на фреймовых представлениях, и язык Пролог, опирающийся на модель представления в виде продукций. Однако разные модели представления знаний имеют свои преимущества и недостатки. Поэтому в конце 80-х годов наметилась тенденция создавать комбинированные языки представления знаний. Чаще всего комбинируются фреймовые и продукционные модели.

Достаточно богатая предметная область содержит большое количество декларативных и процедурных знаний. Создание баз знаний большого размера — дело весьма сложное, ведь необходимо не только накапливать знания, представляя их выбранным способом, но и проверять полноту знаний и их непротиворечивость. Источниками знаний могут быть книги, документы, изобразительная продукция, устные тексты, получаемые от специалистов, и т.п. Эти различные источники знаний надо уметь объединять между собой, что приводит к сложным, интегрированным базам знаний. Отдельные базы знаний, территориально разнесенные между собой, могут совместно использоваться при решении задач. Так возникают распределенные базы знаний, образуются сложные по конфигурации сети баз знаний. Такие сети хранения и обработки знаний являются не только общенациональными, но и международными, доступными любому специалисту.

7.6. ЭКСПЕРТНЫЕ СИСТЕМЫ

В течение последнего десятилетия в рамках исследований по искусственному интеллекту сформировалось самостоятельное направление — экспертные системы (ЭС), или инженерия знаний. В задачу этого направления входят исследование и разработка программ (устройств), ис-

пользующих знания и процедуры вывода для решения задач, являющихся трудными для людей-экспертов. ЭС могут быть отнесены к системам ИИ общего назначения — системам, которые не только исполняют заданные процедуры, но на основе метапроцедур поиска генерируют и используют процедуры решения новых конкретных задач.

Огромный интерес к ЭС со стороны пользователей вызван, по крайней мере, тремя причинами. Во-первых, они ориентированы на решение широкого круга задач в неформализованных областях, на приложения, которые до недавнего времени считались малодоступными для вычислительной техники. Во-вторых, с помощью ЭС специалисты, не знающие программирования, могут самостоятельно разрабатывать интересующие их приложения, что позволяет резко расширить сферу использования вычислительной техники. В-третьих, ЭС при решении практических задач достигают результатов, не уступающих, а иногда и превосходящих возможности людей-экспертов, не оснащенных ЭС.

В настоящее время ЭС применяются в различных областях человеческой деятельности.

Общая характеристика, структура и режимы использования ЭС. *Экспертная система* — наиболее известный и распространенный вид интеллектуальных систем.

Первая особенность ЭС состоит в том, что они предназначены для пользователей, сфера деятельности которых далека от искусственного интеллекта, программирования, математики, логики. Для таких пользователей ЭС выступает как некая система, помогающая им в повседневной работе. Общения с ЭС должны быть так же просты, как просты, например, работа с телевизором, стиральной машиной или автомобилем.

Что дают пользователю экспертные системы? Во многих видах человеческой деятельности используемые знания далеко не всегда могут быть четко формализованы. Точнее, наряду со знаниями как бы отделенными, отобранными у специалистов (они зафиксированы в учебниках, инструкциях, учебных фильмах и т.п.), существуют так называемые навыки и умения, овладеть которыми можно только работая вместе с теми, кто уже овладел ими. Профессионал высокого уровня: врач, геолог, экономист, инженер, технолог и т.д. — отличается от новичка, овладевшего знаниями в объеме учебных программ института или университета, именно этими навыками и умениями. И повышение их профессионального мастерства — огромная социальная задача. Чтобы ее решить, надо научиться извлекать из профессионалов-экспертов те знания, которые не зафиксированы в форме, пригодной для массового распространения — в книгах, кинофильмах, наглядных пособиях. В этом и заключается вторая особенность экспертных систем. Само название этих систем указывает на то, что они должны хранить в себе знания профессионалов-экспертов в некоторой предметной области. И не просто хранить, но и передавать их тем, у кого таких знаний нет. Для этого

в экспертной системе предусмотрены не только простые средства общения между системой и специалистами, но и средства доведения хранимых в системе знаний до специалиста вместе с необходимыми пояснениями и разъяснениями.

На практике ЭС используются прежде всего как системы-советчики в тех ситуациях, где специалист сомневается в выборе правильного решения. Экспертные знания, хранящиеся в памяти системы, более глубокие и полные, чем соответствующие знания пользователя. Однако возможны и другие случаи применения экспертных систем.

Типичная ЭС состоит из следующих основных компонентов: решателя (интерпретатора), рабочей памяти (РП), называемой также базой данных (БД), базы знаний (БЗ), компонентов приобретения знаний, объяснительного и диалогового компонентов (рис. 7.9).



Рис. 7.9. Типовая структура экспертной системы

База данных предназначена для хранения исходных и промежуточных данных решаемой в текущий момент задачи. Этот термин совпадает по названию, но не по смыслу с термином, используемым в информационно-поисковых системах (ИПС) и системах управления базами данных (СУБД) для обозначения всех данных (и в первую очередь не текущих, а долгосрочных), хранимых в системе. База знаний в ЭС предназначена для хранения долгосрочных данных, описывающих рассматриваемую область (а не текущих данных), и правил, описывающих целесообразные преобразования данных этой области. Решатель, используя исходные данные из РП и знания из БЗ, формирует такую последовательность правил, которые, будучи примененными к исходным данным, приводят к решению задачи. Объяснительный компонент объясняет, как система получила решение задачи (или почему она не получила решения) и какие знания она при этом использовала, что облегчает эксперту тестирование системы и повышает доверие пользователя к полученному результату. Диалоговый компонент ориентирован на организацию дружелюбного общения со всеми категориями пользователей как в ходе решения задач, так и приобретения знаний, объяснения результатов работы.

В разработке ЭС участвуют представители следующих специальностей:

эксперт в той проблемной области, задачи которой будет решать ЭС;

инженер по знаниям — специалист по разработке ЭС;

программист — специалист по разработке инструментальных средств (ИС).

Необходимо отметить, что отсутствие среди участников разработки инженера по знаниям (т.е. его замена программистом) либо приводит к неудаче процесс создания ЭС, либо значительно удлиняет его. Эксперт определяет знания (данные и правила), характеризующие проблемную область, обеспечивает полноту и правильность введения в ЭС знаний. Инженер по знаниям помогает эксперту выявить и структурировать знания, необходимые для работы ЭС, осуществляет выбор того ИС, которое наиболее подходит для данной проблемной области, и определяет способ того представления знаний в этом ИС, выделяет и программирует (традиционными средствами) стандартные функции (типичные для данной проблемной области), которые будут использоваться в правилах, вводимых экспертом.

Программист разрабатывает ИС, содержащее в пределе все основные компоненты ЭС, осуществляет сопряжение ИС с той средой, в которой оно будет использоваться.

Экспертная система работает в двух режимах: приобретения знаний и решения задач (называемом также режимом консультации или режимом использования ЭС).

В режиме приобретения знаний общение с ЭС осуществляется через посредничество инженера по знаниям. Эксперт описывает проблемную область в виде совокупности данных и правил. Данные определяют объекты, их характеристики и значения, существующие в области экспертизы. Правила определяют способы манипулирования данными, характерные для рассматриваемой проблемной области. Эксперт, используя компонент приобретения знаний, наполняет систему знаниями, которые позволяют ЭС в режиме решения самостоятельно (без эксперта) решать задачи из проблемной области.

Важную роль в режиме приобретения знаний играет объяснительный компонент. Именно благодаря ему эксперт на этапе тестирования локализует причины неудачной работы ЭС, что позволяет эксперту целенаправленно модифицировать старые или вводить новые знания. Обычно объяснительный компонент сообщает следующее: как правильно используют информацию пользователя; почему использовались или не использовались данные или правила; какие были сделаны выводы и т.д. Все объяснения делаются, как правило, на ограниченном естественном языке или языке графики.

Отметим, что режиму приобретения знаний при традиционном подходе к разработке программ соответствуют этапы алгоритмизации, про-

граммирования и отладки, выполняемые программистом. В отличие от традиционного подхода разработку программ осуществляет эксперт (с помощью ЭС), не владеющий программированием, а не программист.

В режиме консультации общение с ЭС осуществляет конечный пользователь, которого интересует результат и (или) способ получения решения. Пользователь в зависимости от назначения ЭС может не быть специалистом в данной проблемной области, в этом случае он обращается к ЭС за советом, не умея получить ответ сам, или быть специалистом, в этом случае он обращается к ЭС, чтобы либо ускорить процесс получения результата, либо возложить на ЭС рутинную работу. Термин «пользователь» означает, что им является и эксперт, и инженер по знаниям, и программист. Поэтому, когда хотят подчеркнуть, что речь идет о том, для кого делалась ЭС, используют термин «конечный пользователь».

В режиме консультации данные о задаче пользователя обрабатываются диалоговым компонентом, который выполняет следующие действия:

распределяет роли участников (пользователя и ЭС) и организует их взаимодействие в процессе кооперативного решения задачи;

преобразует данные пользователя о задаче, представленные на привычном для пользователя языке, на внутренний язык системы;

преобразует сообщения системы, представленные на внутреннем языке, в сообщения на языке, привычном для пользователя (обычно это ограниченный естественный язык или язык графики).

После обработки данные поступают в РП. На основе входных данных в РП, общих данных о проблемной области и правил из БЗ решатель (интерпретатор) формирует решение задачи.

В отличие от традиционных программ ЭС в режиме решения задачи не только исполняет предписанную последовательность операций, но и предварительно формирует ее. Если ответ ЭС не понятен пользователю, то он может потребовать объяснения, как ответ получен.

7.6.1. КЛАССИФИКАЦИЯ ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ЭС

В широком толковании в инструментарий включают и аппаратуру, ориентированную на разработку ЭС (аппаратурный инструментарий). На проектирование и создание одной ЭС раньше требовалось 20 — 30 чел.-лет. В настоящее время имеется ряд средств, ускоряющих создание ЭС. Эти средства называются инструментальными (ИС) или инструментари-ем. Использование ИС сокращает время разработки ЭС в 3 — 5 раз.

Экспертные системы выполняются на ЭВМ следующих типов: ПЭВМ общего назначения; интеллектуальных рабочих станциях (т.е. рабочих станциях типа Sun, Apollo и др., снабженных эффективными ИС для создания ЭС); последовательных символьных ЭВМ типа

ЛИСП-машин (Symbolik-3670, Alpha, Explorer, Xerox 1100 и др.) и ПРОЛОГ-машин; параллельных символьных ЭВМ (Connection, Dado, Faun, Nuper Cube и др.).

Программные ИС определяются следующей совокупностью характеристик: назначение; стадия существования; тип ИС; тип используемых методов и знаний; универсальность; основные свойства; среда функционирования.

Назначение определяет, для работы в каких проблемных областях и для создания какой стадии ЭС предназначено ИС.

По степени отработанности ИС обычно выделяют три стадии существования: экспериментальная, исследовательская, коммерческая. Экспериментальные ИС создаются для решения узких специфических задач и редко проверяются на других задачах, обычно они работают медленно и неэффективно. Следующей стадией является исследовательская. Средства, достигшие этой стадии, обычно тщательно проверены, имеют документацию и поддерживаются разработчиком, однако они еще могут действовать медленно и неэффективно. Исследовательские ИС используются при разработке прототипов ЭС. Высшей стадией существования ИС является коммерческая. Этой стадии достигают те ИС, которые всесторонне и тщательно проверены, хорошо документированы, сопровождаются разработчиком, являются быстрыми и обладают удобным интерфейсом с пользователем.

По типу ИС классифицируются следующим образом:

- 1) символьные языки программирования, ориентированные на создание ЭС и систем ИИ (например, LISP, INTERLISP, SMALLTALK);
- 2) языки инженерии знаний, т.е. языки высокого уровня, ориентированные на построение ЭС (например, OPS-5, LOOPS, KES, ПРОЛОГ);
- 3) системы, автоматизирующие разработку (проектирование) ЭС (например, KEE, ART, TEIRESIAS, AGE, TMM), их часто называют окружением (environment) для разработки систем ИИ, ориентированных на знания;
- 4) оболочки ЭС (или пустые ЭС) — ЭС, не содержащие знаний ни о какой проблемной области (например, ЭКСПЕРТИЗА, ЕМУСИН, ЭКО, ЭКСПЕРТ).

В приведенной классификации ИС перечислены в порядке убывания трудозатрат, необходимых на создание с их помощью конкретной ЭС.

Инструментальные средства можно классифицировать и по классам ЭС на: ИС для создания простых ЭС, ИС для создания сложных ЭС. В настоящее время, как правило, ИС первого типа разрабатываются на ПЭВМ, а второго — на символьных ЭВМ, ЭВМ общего назначения и интеллектуальных рабочих станциях.

По типу используемых методов и знаний ИС делятся, так же как и ЭС, на традиционные, использующие только методы и способы представления инженерии знаний, и гибридные, сочетающие

подходы инженерии знаний с подходами, развитыми в традиционном программировании при представлении данных и использовании подпрограмм.

Универсальность задается совокупностью двух параметров: универсальностью представления знаний и универсальностью функционирования. Универсальность представления характеризует способ (модель) представления знаний в ИС и принимает следующие значения: единое представление — ИС использует одну модель; интегральное представление — ИС допускает интегральное использование нескольких моделей; универсальное — ИС допускает интегральное использование всех основных моделей представления. К основным моделям представления относятся: правила (продукции); фреймы или семантические сети; логические модели (исчисление предикатов). Примерами ИС, в которых используется единое представление, является ПРОЛОГ, интегральное представление — CENTAUR, а универсальное — KEE, ART.

Основные свойства определяют особенности ИС с точки зрения реализации компонентов ЭС.

Среда функционирования определяет тип ЭВМ, на которой реализовано ИС, тип операционной системы, в среде которой ИС работает, и используемый язык программирования.

7.6.2. ОРГАНИЗАЦИЯ ЗНАНИЙ В ЭС

Для специалистов в области ИИ термин *знания* означает информацию, которая необходима программе, чтобы она вела себя «интеллектуально». Эта информация принимает форму фактов или правил.

Факты и правила в ЭС не всегда либо истинны, либо ложны; иногда существует некоторая степень неуверенности в достоверности факта или точности правила. Если это сомнение выражено явно, то оно называется «коэффициентом уверенности».

Многие правила ЭС являются *эвристиками*, т.е. эмпирическими правилами или упрощениями, которые эффективно ограничивают поиск решения. Экспертная система использует эвристики, потому что задачи, которые она решает, будь то поиск новых месторождений или согласование исков, как правило, трудны и не до конца понятны. Эти задачи не поддаются строгому математическому анализу или алгоритмическому решению. Алгоритмический метод гарантирует корректное или оптимальное решение задачи, тогда как эвристический метод дает приемлемое решение в большинстве случаев.

Знания в ЭС организованы таким образом, чтобы знания о предметной области отделить от других типов знаний системы, таких, как общие знания о том, как решать задачи, или знания о том, как взаимодействовать с пользователем, например как печатать текст на терминале пользователя

или как изменить текст в соответствии с командами пользователя. Выделенные знания о предметной области называются *базой знаний*, тогда как общие знания о нахождении решений задач называются *механизмом вывода*. Программа, которая работает со знаниями, организованными подобным образом, называется *системой, основанной на знаниях*.

База знаний ЭС содержит факты (данные) и правила (или другие представления знаний), использующие эти факты как основу для принятия решений. Механизм вывода содержит интерпретатор, определяющий, каким образом применять правила для вывода новых знаний, и диспетчер, устанавливающий порядок применения этих правил. Такая структура ЭС показана на рис. 7.10.



Рис. 7.10. Состав экспертной системы

Выделение знаний о предметной области облегчает инженеру по знаниям разработку процедур для манипулирования ими. Каким образом система использует свои знания, имеет первостепенное значение, поскольку ЭС должна иметь и *адекватные знания*, и *средства эффективно использовать знания*, чтобы ее можно было считать *умелой* в каком-либо виде деятельности. Следовательно, для того, чтобы быть *умелой*, ЭС должна иметь базу знаний, содержащую высококачественные знания о предметной области, а ее механизм вывода должен содержать знания о том, как эффективно использовать знания о предметной области.

Концепция *механизма вывода* ЭС часто вызывает некоторое недоумение среди начинающих разработчиков. Обычно ясно, как знания предметной области могут быть записаны в виде фактов и правил, но далеко не ясно, каким образом конструировать и использовать так называемый «механизм вывода». Это недоумение происходит от отсутствия простого и общего метода организации логического вывода. Его структура зависит и от специфики предметной области и от того, как знания структурированы и организованы в ЭС. Многие языки высокого уровня, предназначенные для построения экспертных систем, например EMYCIN, имеют механизм вывода, в некотором смысле встроенный в язык как его часть.

Другим примером может служить язык программирования ПРОЛОГ со встроенным в него механизмом логического вывода, который может быть непосредственно использован при создании простых ЭС.

Языки более низкого уровня, например LISP, требуют, чтобы создатель ЭС спроектировал и реализовал механизм вывода.

Оба подхода имеют свои достоинства и недостатки. Язык высокого уровня со встроенным механизмом вывода облегчает работу создателя ЭС. В то же время у него, понятно, меньше возможностей определять способы организации знаний и доступа к ним, и ему следует очень внимательно рассмотреть вопрос о том, годится или нет на самом деле предлагаемая схема управления процессом поиска решения для данной предметной области. Использование языка более низкого уровня без механизма вывода требует больших усилий на разработку, но позволяет разработать нужные программные блоки, которые разработчик может встроить в схему управления процессом решения, который будет адекватен данной предметной области.

Что касается механизма вывода, то здесь дело не ограничивается выбором «все или ничего». Так, некоторые инструменты построения ЭС имеют набор встроенных механизмов вывода, но позволяют разработчику модифицировать или переопределять их для большего соответствия с предметной областью.

Рассмотрим, каким образом знания структурированы в программах, т.е. способы представления знаний. Существует много стандартных способов представления знаний, и при построении ЭС может быть использован любой из них, сам по себе или в сочетании с другими. Каждый способ позволяет получить программу с некоторыми преимуществами — делает ее более эффективной, облегчает ее понимание и модификацию. Широкий обзор наиболее важных способов можно найти в «Справочнике по искусственному интеллекту». В современных ЭС чаще всего используются три самых важных метода представления знаний: правила (самый популярный), семантические сети и фреймы.

Представление знаний, основанное на правилах, построено на использовании выражений вида ЕСЛИ (*условие*) — ТО (*действие*). Например:

[1] Если пациент был по профессии изолировщиком до 1988 г., то пациент непосредственно работал с асбестом.

[2] Если пациент непосредственно работал с асбестом и пациент находился при этом в закрытом помещении, то пациент получил большую дозу асбестовой пыли.

Когда текущая ситуация (факты) в задаче удовлетворяет или согласуется с частью правила ЕСЛИ, то выполняется действие, определяемое частью ТО. Это действие может оказаться воздействием на окружающий мир (например, вызовет распечатку текста на терминале пользователя), или же повлиять на управление программой (например, вызвать проверку и запуск некоторого набора правил), или может сводиться к указанию системе о получении определенного заключения (например, необходимо добавить новый факт или гипотезу в базу данных).

Сопоставление частей ЕСЛИ правил с фактами может породить так называемую *цепочку выводов*. Цепочка выводов, образованная последова-

тельным применением правил 1 и 2, изображена на рис. 7.11. Эта цепочка показывает, как система использует правила для вывода о том, насколько серьезную дозу канцерогенного вещества пациент получил при работе с асбестом.



Рис. 7.11. Цепочка вывода для получения заключения о накопленной дозе асбестовой пыли

Представление знаний, основанное на фреймах, использует сеть узлов, связанных отношениями и организованных иерархически. Каждый узел представляет собой концепцию, которая может быть описана атрибутами и значениями, связанными с этим узлом. Узлы, которые занимают более низкое положение в иерархии, автоматически наследуют свойства узлов, занимающих более высокое положение. Эти методы обеспечивают естественный и эффективный путь классификации и построения таксономии, например залежей руд или различных заболеваний.

Наиболее важные термины по ЭС собраны в табл. 7.1. Более полное описание этих понятий дано в словаре терминов ЭС в Приложении 3.

Таблица 7.1

Термин	Значение
Алгоритм	Формальная процедура, которая гарантирует получение оптимального или корректного решения
База знаний	Часть системы, основанной на знаниях, или экспертной системы, содержащей предметные знания
Диспетчер	Часть механизма вывода, которая решает, когда и в каком порядке применять различные «куски» предметных знаний
Знания	Информация, необходимая программе для того, чтобы эта программа вела себя интеллектуально
Интерпретатор	Часть механизма вывода, которая решает, каким образом применять предметные знания

Термин	Значение
Коэффициент уверенности	Число, которое означает вероятность или степень уверенности, с которой можно считать данный факт или правило достоверным или справедливым
Механизм вывода	Та часть ЭС, в которой содержатся общие знания о схеме управления решением задач
Правило	Предметные знания, знания о предметной области
Представление знаний	Процесс структурирования предметных знаний с целью облегчить поиск решения задачи
Семантическая сеть	Метод представления знаний посредством сети узлов, соответствующих концепциям или объектам, связанных дугами, которые описывают отношения между узлами
Система, основанная на знаниях	Программа, в которой предметные знания представлены в явном виде и отделены от прочих знаний программы
Фрейм	Метод представления знаний, когда свойства связываются с вершинами, представляющими концепции или объекты. Свойства описываются в терминах атрибутов (называемых слотами) и их значений
Эвристики	Правило, которое упрощает или ограничивает поиск решений в предметной области, которая является сложной или недостаточно изученной

7.6.3. ВИДЫ ЭС И ТИПЫ РЕШАЕМЫХ ИМИ ЗАДАЧ

Можно расширить понимание ЭС, рассмотрев некоторые наиболее характерные их применения. В настоящем параграфе опишем эти применения в двух аспектах — основные виды деятельности ЭС и области, в которых они решают проблемы. Взятые в совокупности, эти аспекты позволяют увидеть широту диапазона применения, многообразие информации и разнообразие форм представления знаний, присущие существующим ЭС.

Экспертные системы создаются для решения разного рода проблем, но основные типы их деятельности можно сгруппировать в категории, приведенные в табл. 7.2.

ЭС, выполняющие *интерпретацию*, как правило, используют информацию от датчиков для описания ситуации. В качестве примера приведем интерпретацию показаний измерительных приборов на химическом заводе для определения состояния процесса. Интерпретирующие системы имеют дело не с четкими символическими представлениями проблемной ситуации, а непосредственно с реальными данными. Они сталкиваются с затруднениями, которых нет у систем других типов, потому что им при-

ходится обрабатывать информацию зашумленную, недостаточную, неполную, ненадежную или ошибочную. Им необходимы специальные методы регистрации характеристик непрерывных потоков данных, сигналов или изображений и методы их символического представления.

Таблица 7.2

Категория	Решаемая проблема
Интерпретация	Описание ситуации по информации, поступающей от датчиков
Прогноз	Определение вероятных последствий заданных ситуаций
Диагностика	Выявление причин неправильного функционирования системы по результатам наблюдений
Проектирование	Построение конфигураций объектов при заданных ограничениях
Планирование	Определение последовательности действий
Наблюдение	Сравнение результатов наблюдений с ожидаемыми результатами
Отладка	Составление рецептов исправления неправильного функционирования системы
Ремонт	Выполнение последовательности предписанных исправлений
Обучение	Диагностика, отладка и исправление поведения обучаемого
Управление	Управление поведением системы как целого

Интерпретирующие ЭС могут обработать разнообразные виды данных. Например, системы анализа сцен и распознавания речи, используя естественную информацию, — в одном случае визуальные образы, в другом — звуковые сигналы — анализируют их характеристики и понимают их смысл. Интерпретация в области химии использует данные дифракции рентгеновских лучей, спектрального анализа или ядерно-магнитного резонанса для вывода химической структуры веществ. Интерпретирующая система в геологии использует каротажное зондирование — измерение проводимости горных пород в буровых скважинах и вокруг них — чтобы определить подповерхностные геологические структуры. Медицинские интерпретирующие системы используют показания следящих систем (например, значения пульса, кровяного давления), чтобы установить диагноз или тяжесть заболевания. Наконец, в военном деле интерпретирующие системы используют данные от радаров, радиосвязи и сонарных устройств, чтобы оценить ситуацию и идентифицировать цели.

ЭС, осуществляющие *прогноз*, определяют вероятные последствия заданных ситуаций. Примерами служат прогноз ущерба урожаю от некоторого вида вредных насекомых, оценивание спроса на нефть на мировом рынке в зависимости от складывающейся геополитической ситуации

и прогнозирование места возникновения следующего вооруженного конфликта на основании данных разведки. Системы прогнозирования иногда используют имитационное моделирование, т.е. программы, которые отражают причинно-следственные взаимосвязи в реальном мире, чтобы сгенерировать ситуации или сценарии, которые могут возникнуть при тех или иных входных данных. Эти возможные ситуации вместе со знаниями о процессах, порождающих эти ситуации, образуют предпосылки для прогноза. Специалисты ИИ пока что разработали сравнительно мало прогнозирующих систем, возможно потому, что очень трудно взаимодействовать с имитационными моделями и создавать их.

ЭС выполняют *диагностирование*, используя описания ситуаций, характеристики поведения или знания о конструкции компонент, чтобы установить вероятные причины неправильного функционирования диагностируемой системы. Примерами служат: определение причин заболевания по симптомам, наблюдаемым у пациентов; локализация неисправностей в электронных схемах и определение неисправных компонент в системе охлаждения ядерных реакторов. Диагностические системы часто являются консультантами, которые не только ставят диагноз, но также помогают в *отладке*. Они могут взаимодействовать с пользователем, чтобы оказать помощь при поиске неисправностей, а затем предложить порядок действий по их устранению. Медицина представляется вполне естественной областью для диагностирования, и действительно, в медицинской области было разработано больше диагностических систем, чем в любой другой отдельно взятой предметной области. Однако в настоящее время многие диагностические системы разрабатываются для приложений к инженерному делу и компьютерным системам.

ЭС, выполняющие *проектирование*, разрабатывают конфигурации объектов с учетом набора ограничений, присущих проблеме. Учитывая то, что проектирование столь тесно связано с *планированием*, многие проектирующие системы содержат механизмы разработки и уточнения планов для достижения желаемого проекта. Наиболее часто встречающиеся области применения планирующих ЭС — химия, электроника и военное дело.

ЭС, которые осуществляют *наблюдение*, сравнивают действительное поведение с ожидаемым поведением системы. Примерами могут служить слежение за показаниями измерительных приборов в ядерных реакторах с целью обнаружения аварийных ситуаций или оценку данных мониторинга больных, помещенных в блоки интенсивной терапии. Наблюдающие ЭС подыскивают наблюдаемое поведение, которое подтверждает их ожидания относительно нормального поведения или их предположения о возможных отклонениях. Наблюдающие ЭС по самой своей природе должны работать в режиме *реального времени* и осуществлять зависящую как от времени, так и от контекста интерпретацию поведения наблюдае-

мого объекта. Это может приводить к необходимости запоминать все значения некоторых параметров системы (например, пульса), полученные в различные моменты времени, поскольку скорость и направление изменения могут быть столь же важны, как и действительные его значения в любой момент времени.

ЭС, выполняющие *отладку*, находят рецепты для исправления неправильного поведения устройств. Примерами могут служить настройка компьютерной системы с целью преодолеть некоторый вид затруднений в ее работе; выбор типа обслуживания, необходимого для устранения неисправностей в телефонном кабеле; выбор ремонтной операции для исправления известной неисправности в насосе.

ЭС, реализующие *ремонт*, следуют плану, который предписывает некоторые рецепты восстановления. Примером является настройка масс-спектрометра, т.е. установка ручек регулировки прибора в положение, обеспечивающее достижение оптимальной чувствительности, совместимой с правильным отношением величин пиков и их формы. Пока что было разработано очень мало ремонтных ЭС отчасти потому, что необходимость фактического выполнения ремонтных процедур на объектах реального мира дополнительно усложняет задачу. Ремонтным системам также необходимы диагностирующие, отлаживающие и планирующие процедуры для производства ремонта.

ЭС, выполняющие *обучение*, подвергают диагностике, «отладке» и исправлению («ремонту») поведение обучаемого. В качестве примеров приведем обучение студентов отысканию неисправностей в электрических цепях, обучение военных моряков обращению с двигателем на корабле и обучение студентов-медиков выбору антимикробной терапии. Обучающие системы создают *модель* того, что обучающийся знает и как он эти знания применяет к решению проблемы. Системы диагностируют и указывают обучающемуся его ошибки, анализируя модель и строя планы исправлений указанных ошибок. Они исправляют поведение обучающихся, выполняя эти планы с помощью непосредственных указаний обучающимся.

ЭС, осуществляющие *управление*, адаптивно руководят поведением системы в целом. Примерами служат управление производством и распределением компьютерных систем или контроль за состоянием больных при интенсивной терапии. Управляющие ЭС должны включать наблюдающие компоненты, чтобы отслеживать поведение объекта на протяжении времени, но они могут нуждаться также и в других компонентах для выполнения любых или всех из уже рассмотренных типов задач: интерпретации, прогнозирования, диагностики, проектирования, планирования, отладки, ремонта и обучения. Типичная комбинация задач состоит из наблюдения, диагностики, отладки, планирования и прогноза.

Хотя основные виды деятельности ЭС, перечисленные в табл. 7.2, легко описать, классификация существующих ЭС на основании этих ви-

дов деятельности может привести к неясностям, поскольку многие из этих систем выполняют сразу несколько видов работ. Например, диагностика часто совмещается с отладкой, наблюдение с управлением, а планирование с проектированием. Поэтому специалисты по ИИ находят полезным классифицировать ЭС по типам задач, которые такие системы решают. Перечислим некоторые из предметных областей, в которых применяются ЭС в настоящее время: военное дело; геология; инженерное дело; информатика; компьютерные системы; космическая техника; математика; медицина; метеорология; промышленность; сельское хозяйство; управление процессами; физика; химия; электроника; юриспруденция. Из них *медицина* представляется наиболее популярной; именно в этой области было разработано больше ЭС, чем во всякой другой, хотя химия ненамного отстает от нее, и разрыв быстро сокращается.

В *химии* работы по ЭС начались с новаторского проекта DENDRAL, начатого в Стэнфордском университете в середине 60-х годов и посвященного разработке методов ИИ для определения топологических структур органических молекул. Современные работы по ЭС в области химии включают вывод структуры молекул, синтез органических молекул и планирование экспериментов в молекулярной биологии.

В области *компьютерных систем* типичным образцом может служить ЭС XCON, одна из первых и наиболее успешно применяемых разработок этого рода. Она была начата корпорацией DEC и Университетом Карнеги-Меллон в конце 70-х годов как исследовательский проект, а сейчас XCON достигла уровня коммерческой системы и используется для проектирования конфигураций компьютеров. Современные разработки ЭС в этой области связаны с диагностикой неисправностей, проектированием компьютерных конфигураций и управлением процессом производства компьютеров.

В *электронике* преобладают исследования и разработки, связанные с диагностикой неисправностей и проектированием СБИС. Система AGE, разработанная в начале 80-х годов фирмой Белл, — это типичная диагностическая система в данной области. Она используется для обнаружения неисправностей в телефонной сети и определения их характера. В настоящее время работы по ЭС в области электроники связаны с обучающими системами, помогающими находить отказы в электрических цепях и проектировать электронно-цифровые схемы.

В *инженерном деле* типичный образец ЭС — это DELTA, система диагностики неисправностей, разработанная компанией «Дженерал Электрик» в середине 80-х годов. «Дженерал Электрик» планирует использовать DELTA на коммерческой основе для помощи обслуживающему персоналу при поиске неисправностей в дизель-электрических локомотивах. В настоящее время разрабатываются другие системы диагностирования неисправностей и обучения операторов сложных систем *управления* (рис. 7.12).

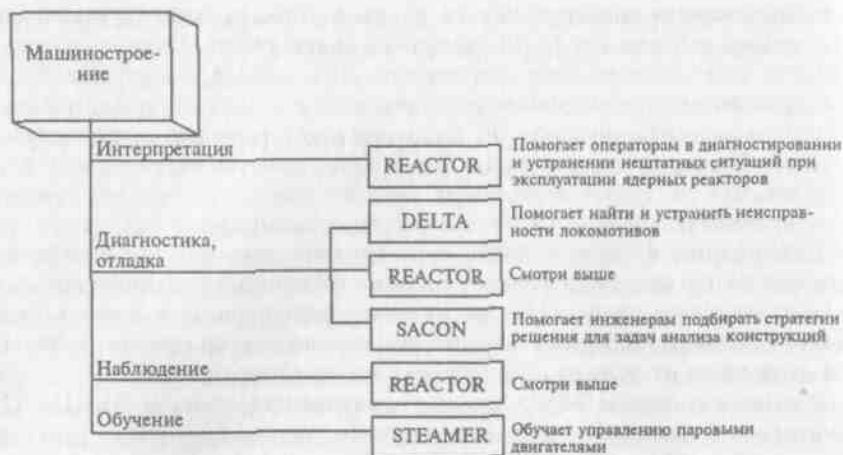


Рис. 7.12. ЭС в инженерном деле

В геологии первой была ЭС PROSPECTOR, разработанная в Стэнфордском исследовательском институте в середине 70-х годов. Система PROSPECTOR предназначена для помощи геологам в разведке рудных месторождений, и в 1980 г. она точно предсказала существование молибденового месторождения, оцененного в многомиллионную сумму. В настоящее время разрабатываются ЭС, связанные с каротажем скважин и диагностикой неполадок при их бурении.

В медицине соответствующие исследования начались с системы MYCIN, одной из первых и наиболее известных ЭС. Разработанная в Стэнфордском университете в середине 70-х годов, MYCIN предназначена для помощи в диагностике и лечении инфекционных заболеваний крови и ныне используется для обучения врачей и исследователей. В настоящее время ведутся разработки медицинских ЭС, осуществляющих интерпретацию лабораторных анализов, диагностику заболеваний, рекомендуя способы лечения и обучающих диагностике и лечению заболеваний, а также методам анестезии.

В военном деле усилия были сконцентрированы на интерпретации, прогнозировании и планировании. Одна из первых военных ЭС, названная HASP/SIAP, разрабатывалась совместно со Стэнфордским университетом и System Control Technology в начале 70-х годов. Эта система определяет типы кораблей, интерпретируя данные от сети гидрофонов, прослушивающих некоторые акватории океана. В настоящее время потенциальные военные приложения ЭС включают интерпретацию информации от датчиков, прогноз боевых действий и тактическое планирование.

Этот обзор применений ЭС дает очень широкую перспективу того, что ЭС делают и какого рода проблемы решают.

7.7. ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Геоинформационные системы (ГИС) и ГИС-технологии объединяют компьютерную картографию и системы управления базами данных. Концепция технологии ГИС состоит в создании многослойной электронной карты, опорный слой которой описывает географию территории, а каждый из остальных слоев — один из аспектов состояния территории. Тем самым ГИС-технологии определяют специфическую область работы с информацией.

Особый интерес к этой области работы с информацией объясняется тем, что сейчас наиболее остро стоят проблемы оперативного анализа пространственно распределенных данных, а в России это более актуально, чем где-либо в мире и объясняется перестройкой экономических отношений и приватизацией недвижимости и земельных участков.

Технология ГИС применима везде, где необходимо учитывать, обрабатывать и демонстрировать территориально распределенную информацию. Пользователями ГИС-технологии могут быть как организации, чья деятельность целиком базируется на земле — владельцы нефтегазовых предприятий, кадастровые и экологические службы, жилищно-коммунальное хозяйство, так и многочисленные коммерческие предприятия — банки, страховые, торговые и строительные фирмы, чья успешная работа во многом зависит от правильного и своевременного учета территориального фактора.

Особенно эффективно применение ГИС-технологии в торговых и обменных операциях с недвижимостью. По прогнозам Европейских служб, занимающихся оценкой различных секторов рынка, в конце XX и начале XXI века пакет ГИС будет таким же неотъемлемым атрибутом рабочего места бизнесмена или управленца, какими сегодня являются текстовые и графические редакторы, статистические пакеты и БД.

В основе любой ГИС лежит информация о каком-либо участке земной поверхности: континенте, стране, городе, улице. БД организуется в виде набора слоев информации. Основной шрифт содержит географически привязанную карту местности (топооснова). На него накладываются другие слои, несущие информацию об объектах, находящихся на данной территории: коммуникации (в том числе линии электропередач, нефте- и газопроводы, водопроводы), промышленные объекты, земельные участки, почвы, коммунальное хозяйство, землепользование и др. В процессе создания и наложения слоев друг на друга между ними устанавливаются необходимые связи, что позволяет выполнять пространст-

венные операции с объектами посредством моделирования и интеллектуальной обработки данных. Как правило, информация представляется графически в векторном виде, что позволяет уменьшить объем хранимой информации и упростить операции по визуализации. С графической информацией связана текстовая, табличная, расчетная информация, координатная привязка к карте местности, видеоизображения, аудиокomentarии, БД с описанием объектов и их характеристик. Многие ГИС включают аналитические функции, которые позволяют моделировать процессы, основываясь на картографической информации.

Программное ядро ГИС можно условно разделить на две подсистемы: СУБД и управление графическим выводом изображения. В качестве СУБД используют SQL-серверы: Sybase SQL Server, MS SQL Server, Watson SQL Server, Borland Interbase.

Уже существуют и используются ГИС для различных служб муниципального хозяйства, для контроля экологической обстановки, активно развиваются системы, ориентированные на ликвидацию последствий стихийно-разрушительных явлений, наконец существуют ГИС справочного назначения для крупнейших городов России и мира.

Рассмотрим типовую схему организации ГИС-технологии. В настоящее время сложился основной набор компонент, составляющих ГИС. К ним относятся:

1. Приобретение и предварительная подготовка данных.
2. Ввод и размещение данных.
3. Управление данными.
4. Манипуляция и данными и их анализ.
5. Производство конечного продукта.

Функциональным назначением данных компонент является:

1. Приобретение и подготовка исходных данных включает манипуляции с исходными карт — материалами на твердой или бумажной основе, данными дистанционного зондирования, результатами полевых испытаний, текстовыми (табличными) материалами, с архивными данными. Эксперты предупреждают от опасности финансовой и организационной (временной) недооценки данной компоненты или фазы функционирования ГИС, потому что точность и корректность функционирования ГИС ограничивается точностью и корректностью исходных данных.

2. Ввод и размещение пространственной и непространственной составляющих данных включает конвертирование информации во внутренние форматы системы и обеспечение структурной и логической совместимости всего множества порождаемых данных. Имеющийся опыт говорит о большой стоимости и временной протяженности мероприятий данной фазы.

3. Управление данными предполагает наличие средств оптимальной внутренней организации данных, обеспечивающий эффективный доступ

к ним. Для большинства пользователей эта компонента явно не представлена при удачной реализации системы ими (пользователями) не замечается.

4. Функции манипуляции и анализа представлены средствами, предназначенными для содержательной обработки данных в целях обработки и реорганизации данных. С точки зрения пользователя эти функции являются главными в ГИС-технологиях, потому что позволяют получать новую информацию, необходимую для управления, исследовательских целей, прогнозирования.

5. Производство конечного продукта включает вывод полученных результатов для конечных потребителей ГИС. Эти продукты могут представлять карты, статистические отчеты, различные графики, стандартные формы определенных документов (например, уведомлений для владельцев участков) и многое другое. При работе с электронными картами на практике различают абстрактную геометрическую форму картографических объектов (например, шоссе дорога представляется кривой линией не имеющей толщины), и ее графическое представление (шоссе дорога может обозначаться двойной линией коричневого цвета).

Кроме этого, каждый картографический объект может иметь атрибутивную информацию, в которой содержится информация, которая не обязательно должна отображаться на карте (например, число жильцов какого-либо дома и их социальный статус).

Подавляющее большинство ГИС-систем различают геометрическую и атрибутивную компоненту баз данных ГИС. Их часто называют также пространственными (картографическими, геометрическими) и непространственными (табличными, реляционными) данными. Геометрическая информация представляется точками, кривыми и площадными объектами.

Атрибутивная информация содержит текстовые, числовые и логические данные о картографических объектах. Большинство современных ГИС-инструментариев позволяют хранить атрибутивную информацию в составе БД, как правило, реляционных. ГИС обычно включают в состав атрибутивной информации также способы картографического (графического) отображения объектов, тем самым достигается большая гибкость при формировании выходных карт.

Классические программные средства ГИС обычно используют понятие слоя (покрытия), определяемого как множество пространственных объектов одного типа (дорог, водных потоков и т.д.) — но, по сути, однотипность объектов определяется возможностью задания одинакового набора атрибутов и построения той или иной топологии. Например, в городских информационных системах типичными являются следующие слои данных: административное деление, земля (лесные угодья, пастбища, сенокосы, пашня, залежи, земли населенных пунктов, промышлен-

ные зоны), водные пространства, животный мир, месторождения полезных ископаемых, недвижимость, население, промышленность, строительство, транспорт, связь, энергетика, инженерные сети и сооружения, экология.

Атрибутивная информация хранится в виде отдельных табличных файлов, как правило в форматах реляционных баз данных систем — DBF, PARADOX, ORACLE, INGRESS. Такой способ характерен для западных коммерческих продуктов и современных отечественных разработок и является более продвинутым, предоставляя пользователю большую гибкость при формировании, конвертировании и реорганизации.

Рассмотрим теперь вопросы, связанные с наиболее трудоемким этапом подготовки, ввода и размещения геоданных. Подготовка, ввод и размещение данных предполагают несколько технологических фаз:

- 1) выбор критериев и средств формализации данных;
- 2) ввод геометрической информации;
- 3) ввод атрибутивной информации;
- 4) построение необходимых связей типа «геометрические данные — атрибутивные данные» и «геометрические данные — геометрические данные»;
- 5) верификация данных.

Остановимся на некоторых вопросах данных фаз. На начальном периоде функционирования ГИС наибольшее значение имеет ввод информации с картографических материалов. Фактически как для подготовки растровых, так и для векторных моделей данных, необходимо провести векторизацию исходных данных. К настоящему времени существуют две самые распространенные технологии ввода векторной картографической информации: оцифровка (использование диджитайзеров) и векторизация отсканированных изображений карт, планов, аэроснимков и космических снимков. В любом случае ввод геоданных потребует тщательно продуманной программно-технической технологической линии и соответствующего персонала.

Построение связей типа «геометрические данные» для традиционных векторных данных означает построение топологических отношений. В ряде случаев построение топологии позволяет создать новые, интегрированные объекты, в других — позволяет моделировать протяженные конструкции, например, сети.

Построение связей типа «геометрические данные — атрибутивные данные» означает, как правило, создание геореляционной модели данных, подключая к геометрическим данным всю мощь операций с реляционными БД.

Ввод атрибутивных данных может быть существенно оптимизирован для классифицированных данных. Ряд векторизаторов (в том числе использующих дигитайзер) предполагают ввод кодов встроенного классификатора. Наследование этого кода от предыдущего элемента позволяет

повысить производительность и сократить число возможных ошибок. Он хорошо согласуется с объектно-ориентированным подходом и естественен для оператора. Классификатор является носителем проведенной для исходных данных разработки принципов формализации. Однако классификатор не освобождает от ввода идентифицирующей информации (например, названия города).

Далее следует построение топологии, что является ответственным и достаточно трудоемким процессом. Вместе с тем это полностью автономный процесс, имеющий сложную специфику.

Рассмотрим вопрос пространственной привязки внешних табличных данных. Приведенные выше схемы ввода предполагают первоначально создание геометрических данных, затем создание атрибутивных данных или связывание с уже существующими табличными данными. Однако огромное количество табличных данных, создаваемых вне ГИС-технологий, содержат данные о пространственном расположении: географические координаты, полный или усеченный почтовый адрес (например, улица или почтовый индекс). В рамках зарубежных ГИС создан механизм, называемый геокодированием (для случаев почтовых адресов — адресным геокодированием), который позволяет для таких табличных данных породить слои, содержащие геометрические (точечные) объекты, соответствующие записям в этих таблицах.

Для ГИС территории представляется более важным адресное геокодирование, поскольку большая часть пространственно-привязанной табличной информации в области управления приводится с указанием административной привязки, в которых присутствуют полные или частичные адреса, например, множество списков лиц указанной категории, предприятия и организации, данные переписи населения или домашнего скота и многое другое. Интеграция таких данных в геоинформационные базы позволяет включить их в разнообразный пространственный анализ, проводя его совместно с использованием других, по существу пространственных объектов — месторождениями полезных ископаемых, трубопроводами, зонами распространения эпидемий, затоплений, районами, благоприятными или неблагоприятными для той или иной деятельности.

Например, в ГИС города в качестве необходимой информационной основы для адресного геокодирования должны на базе геоданных присутствовать адресные файлы, на которых изображены регионы почтовых отделений. Они содержат осевые линии улиц населенных пунктов по сегментам, от угла до угла квартала. В атрибутивной информации для каждого сегмента представлены названия улиц, начальные и конечные номера домов по четной и нечетной стороне. При адресном геокодировании для каждой записи с указанным адресом порождается точка в пределах сегмента согласно пропорциям указанного в записи номера дома, номеров домов на концах сегмента и длины сегмента.

Очевидно, реальное расположение дома может существенно отличаться от сгенерированного расположения точек. Однако точное расположение точек для статистического анализа по большим территориям не требуется. Для задачи, например, относительной обеспеченности предприятиями соцкультбыта микрорайонов города достаточно точности размещения магазинов или парикмахерских с точностью 200 — 300 м, которое вполне обеспечивается функциями геокодирования.

И наконец, последнее замечание. Поступающая с разных рабочих мест по вводу данных (в том числе и через единую компоненту ввода) первичная информация обычно подлежит последующей обработке как перед сохранением в БД, так и после извлечения информации для представления. Первичная обработка включает:

1. Полуавтоматический контроль ошибок вводимой информации.
2. Полуавтоматическая классификация данных.
3. Полуавтоматическое установление связей типа «геометрические данные — атрибутивные данные» (например, привязка данных к карте местности).

4. Автоматическое кодирование и сжатие информации.

Криптование информации, если это предусмотрено регламентом.

Обработка информации, извлекаемой из БД предполагает выполнение следующих операций:

1. Поиск и выборка информации по различным атрибутам.
2. Статистическая обработка данных.
3. Решение задач маршрутизации и оптимизации по геоинформационным данным.
4. Решение задач моделирования процессов и генерации прогнозов по реальным данным.

Это определяет необходимость доступа к единой географической основе со всех рабочих мест, в которых привязываются данные остальных слоев при вводе и выводе информации.

Контрольные вопросы

1. По каким признакам классифицируют ИТ?
2. В чем отличия новых (современных) ИТ от традиционных?
3. Каково назначение ИТ поддержки принятия решений?
4. Что такое электронный офис?
5. Расскажите о назначении компьютерной графики.
6. Охарактеризуйте содержание интерактивной графики.
7. Как Вы понимаете термин «гипертекст»?
8. Какие составляющие входят в мультимедиа - технологию?
9. Расскажите о пакетной, диалоговой и сетевой ИТ.
10. Что такое интегрированная ИТ?
11. Из каких основных частей состоит банк данных?

12. Что такое база данных?
13. Как Вы понимаете разницу между понятиями «информация» и «данные»?
14. Какие Вы знаете модели данных?
15. Каково назначение СУБД?
16. Что такое ключ записи?
17. Каким образом может быть организован поиск записи в БД?
18. Перечислите обязанности администратора баз данных.
19. Расскажите классификацию ППП.
20. Какие пакеты входят в проблемно-ориентированные ППП?
21. Назовите основные функции текстовых и табличных процессоров и графических редакторов?
22. Как Вы понимаете назначение программ и мультимедиа?
23. В чем состоят преимущества и недостатки интегрированных ППП?
24. Дайте определение автоматизированного рабочего места специалиста.
25. Каковы роль и место АРМ в автоматизированной информационной технологии?
26. Какие Вы знаете виды технологического обеспечения АРМ?
27. Что такое ИИ?
28. Какие существуют направления исследований в области ИИ?
29. Охарактеризуйте основные блоки робота.
30. Каковы функции планировщика и решателя в системе управления робота?
31. Объясните понятие «машинное зрение» робота?
32. Что такое нейронные сети?
33. Какие принципы заключены в эвристическое программирование?
34. Дайте краткую характеристику моделям представления знаний.
35. Что такое экспертная система?
36. Каково назначение ЭС?
37. Из каких основных элементов состоит ЭС?
38. Дайте краткую характеристику инструментальным средствам ЭС.
39. Как организованы знания в ЭС?
40. Назовите основные виды деятельности ЭС.
41. Охарактеризуйте типы задач, решаемые ЭС.
42. Объясните термины «геоинформационная система» и «ГИС-технология».
43. Перечислите основной набор компонент ГИС-технологии.
44. Что входит в городскую информационную систему по концепции ГИС-технологии?
45. В чем состоит обработка данных, поступающая в БД и извлекаемая из нее?

ГЛАВА 8

ВЫЧИСЛИТЕЛЬНЫЕ И ИНФОРМАЦИОННЫЕ СЕТИ

Появление сетей ЭВМ по своей сути произвело своего рода техническую революцию, так как было произведено объединение технологии сбора, хранения, обработки и передачи информации на ЭВМ с техникой связи. Первые вычислительные сети ЭВМ (ВС) объединяли большие ЭВМ и ВЦ и были разработаны в 60-х годах в США и СССР. В США это была сеть ARPA, объединявшая пятьдесят университетов и фирм, а в СССР — сеть Академии наук в Ленинграде.

Методические указания

Важным моментом уяснения студентом содержания восьмой главы является то, что любая вычислительная и информационная сеть — это совокупность технических, коммуникационных и программных средств, обеспечивающих эффективное распределение вычислительных ресурсов.

Студенты должны понимать, что вычислительные и информационные сети, являясь одновременно и продуктом, и мощным стимулом развития интеллекта человека, позволяют:

- построить мощные хранилища информации (БД);
 - расширить перечень решаемых задач по обработке информации;
 - повысить надежность информационной системы за счет дублирования работы ПК;
 - создать новые виды сервисного обслуживания, например, электронную почту;
 - снизить стоимость обработки информации.
- Изучив материал восьмой главы, студенты должны знать:
- определение и назначение сетей;
 - структуру сетей;
 - способы коммутации и передачи данных в сетях;
 - виды локальных сетей;

- способы обеспечения защиты информации в сетях.
- Студенты должны иметь представление о:
- признаках классификации сетей;
 - программном обеспечении сетей;
 - методах доступа в сети;
 - моделях взаимодействия пользователей с сетевыми элементами;
 - компьютерном вирусе и способах борьбы с ним;
 - службе безопасности в сетях.

8.1. ПРИНЦИПЫ ПОСТРОЕНИЯ И КЛАССИФИКАЦИЯ СЕТЕЙ

Создание высокоэффективных крупных систем обработки данных связано с объединением средств вычислительной техники, обслуживающей отдельные предприятия, организации и их подразделения, с помощью средств связи в единую распределенную вычислительную систему.

Такое комплексирование средств вычислительной техники позволяет повысить эффективность систем обработки информации за счет снижения затрат, повышения надежности и производительности эксплуатируемых ЭВМ, рационального сочетания преимуществ централизованной и децентрализованной обработки информации благодаря приближению средств сбора исходной и выдачи результатной информации непосредственно к местам ее возникновения и потребления, а также комплексного использования единых мощных вычислительных и информационных ресурсов.

Децентрализация процесса обработки данных реализовывалась по двум направлениям:

путем подключения к отдельным ЭВМ (или комплексу ЭВМ, объединенных в рамках ВЦ) множества абонентских пунктов пользователей, т.е. создания систем телеобработки данных;

путем создания вычислительных и информационных сетей, в которых осуществлялось объединение между собой множества территориально удаленных друг от друга ЭВМ.

Передача информации между территориально удаленными компонентами подобных распределенных систем осуществляется в основном с помощью стандартных телефонных и телеграфных каналов, а также витых пар проводов и коаксиальных кабелей связи. Современный прогресс в области оптоволоконной техники (использование световодов) позволяет резко повысить пропускную способность линий связи. Так, система F6M обеспечивает передачу информации до 6,3 Мбит/с, заменяя до 96 телефонных каналов, а система F400M — передачу до 400 Мбит/с информации, заменяя 5760 телефонных каналов.

Расширение состава и совершенствование аппаратуры приема-передачи, а также резкое снижение стоимости ВТ привели к использова-

нцию в качестве абонентских пунктов систем телеобработки данных интеллектуальных терминалов, создаваемых на базе микропроцессоров и микроЭВМ и обеспечивающих частичную обработку информации (главным образом предварительную обработку исходной информации в виде ее логического контроля, агрегирования и т.д.) непосредственно до ее передачи по каналам связи. Использование интеллектуальных терминалов сближает функциональные возможности систем телеобработки данных и ВС. В настоящее время ВС представляют собой высшую организационную форму применения ЭВМ.

Для современных вычислительных информационных сетей характерно:

объединение многих достаточно удаленных друг от друга ЭВМ и (или) отдельных ВС в единую распределенную систему обработки данных;

применение средств приема-передачи данных и каналов связи для организации обмена информацией в процессе взаимодействия средств ВТ;

наличие широкого спектра периферийного оборудования, используемого в виде абонентских пунктов и терминалов пользователей, подключаемых к узлам сети передачи данных;

использование унифицированных способов сопряжения технических средств и каналов связи, облегчающих процедуру наращивания и замену оборудования;

наличие операционной системы, обеспечивающей надежное и эффективное применение технических и программных средств в процессе решения задач пользователей вычислительной сети.

Особенностью эксплуатации ВС является не только приближение аппаратных средств непосредственно к местам возникновения и использования данных, но и разделение функций обработки и управления на отдельные составляющие с целью их эффективного распределения между несколькими ЭВМ, а также обеспечение надежного и быстрого доступа пользователей к вычислительным и информационным ресурсам и организация коллективного использования этих ресурсов.

Вычислительные сети позволяют автоматизировать управление производством, транспортом, материально-техническим снабжением в масштабе отдельных регионов и страны в целом.

Возможность концентрации в вычислительных сетях больших объемов данных, общедоступность этих данных, а также программных и аппаратных средств обработки и высокая надежность их функционирования — все это позволяет улучшить информационное обслуживание пользователей и резко повысить эффективность применения ВТ.

В условиях ВС предусмотрена возможность:

организовать параллельную обработку данных многими ЭВМ;

создавать распределенные базы данных, размещаемые в памяти различных ЭВМ;

специализировать отдельные ЭВМ (группы ЭВМ) для эффективного решения определенных классов задач;

автоматизировать обмен информацией и программами между отдельными ЭВМ и пользователями сети;

резервировать вычислительные мощности и средства передачи данных на случай выхода из строя отдельных из них с целью быстрого восстановления нормальной работы сети;

перераспределять вычислительные мощности между пользователями сети в зависимости от изменения их потребностей и сложности решаемых задач;

стабилизировать и повышать уровень загрузки ЭВМ и дорогостоящего периферийного оборудования;

сочетать работу в широком диапазоне режимов: диалоговом, пакетном, режимах «запрос-ответ», а также сбора, передачи и обмена информацией.

Как показывает практика, за счет расширения возможностей обработки данных, лучшей загрузки ресурсов и повышения надежности функционирования системы в целом стоимость обработки данных в вычислительных сетях не менее чем в полтора раза ниже по сравнению с обработкой аналогичных данных на автономных ЭВМ.

Вычислительные и информационные сети классифицируются по различным признакам. Сети, состоящие из программно-совместимых ЭВМ, являются *однородными* или *гомогенными*. Если ЭВМ, входящие в сеть, программно несовместимы, то такая сеть называется *неоднородной* или *гетерогенной*.

По типу организации передачи данных различают сети: *с коммутацией каналов*, *с коммутацией сообщений*, *с коммутацией пакетов*. Имеются сети, использующие смешанные системы передачи данных.

По характеру реализуемых функций сети подразделяются на:

вычислительные, предназначенные для решения задач управления на основе вычислительной обработки исходной информации;

информационные, предназначенные для получения справочных данных по запросу пользователей;

смешанные, в которых реализуются вычислительные и информационные функции.

По способу управления вычислительные и информационные сети делятся на сети с *децентрализованным*, *централизованным* и *смешанным управлением*. В первом случае каждая ЭВМ, входящая в состав сети, включает полный набор программных средств для координации выполняемых сетевых операций. Сети такого типа сложны и достаточно дороги, так как операционные системы отдельных ЭВМ разрабатываются с ориентацией на коллективный доступ к общему полю памяти сети. При этом в каждый конкретный момент времени доступ к общему полю памяти пре-

доставляется только для одной ЭВМ. А координация работы ЭВМ осуществляется под управлением единой операционной системы сети.

В условиях смешанных сетей под централизованным управлением ведется решение задач, обладающих высшим приоритетом и, как правило, связанных с обработкой больших объемов информации.

По структуре построения (топологии) сети подразделяются на *одноузловые* и *многоузловые*, *одноканальные* и *многоканальные*. Топология вычислительной сети во многом определяется структурой сети связи, т.е. способом соединения абонентов друг с другом и ЭВМ. Известны такие структуры сетей: радиальная (звездообразная), кольцевая, многосвязная («каждый с каждым»), иерархическая, «общая шина» и др. (рис. 8.1). Подробная характеристика этих сетей будет дана в § 9.3.

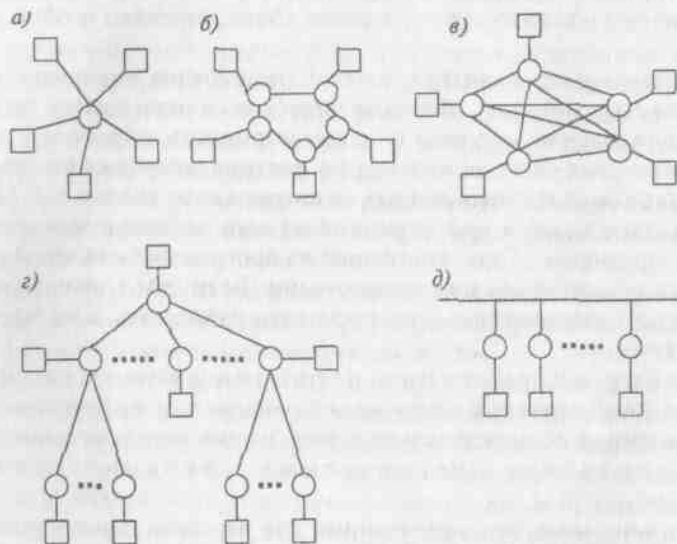


Рис. 8.1. Основные типы структур сетей ЭВМ: радиальная (звездообразная) (а); кольцевая (б); многосвязная (в); иерархическая (г); «общая шина» (д); □ — ЭВМ; О — узел коммутации

8.2. СПОСОБЫ КОММУТАЦИИ И ПЕРЕДАЧИ ДАННЫХ В ВС

Основная функция систем передачи данных в условиях функционирования ВС заключается в организации быстрой и надежной передачи информации произвольным абонентам сети, а также в сокращении затрат на передачу данных. Последнее особенно важно, так как за прошедшее

десятилетие произошло увеличение доли затрат на передачу данных в общей структуре затрат на организацию сетевой обработки информации. Это объясняется главным образом тем, что затраты на техническое обеспечение ВС сократились за этот период примерно в десять раз, тогда как затраты на организацию и эксплуатацию каналов связи сократились только в два раза.

Важнейшая характеристика сетей передачи данных — время доставки информации — зависит от структуры сети передачи данных, пропускной способности линий связи, а также от способа соединения каналов связи между взаимодействующими абонентами сети и способа передачи данных по этим каналам. В настоящее время различают системы передачи данных с постоянным включением каналов связи (*некоммутируемые каналы связи*) и коммутацией на время передачи информации по этим каналам.

При использовании *некоммутируемых каналов связи* средства приема-передачи абонентских пунктов и ЭВМ постоянно соединены между собой, т.е. находятся в режиме «on-line». В этом случае отсутствуют потери времени на коммутацию, обеспечиваются высокая степень готовности системы к передаче информации, более высокая надежность каналов связи и, как следствие, достоверность передачи информации. Недостатками такого способа организации связи являются низкий коэффициент использования аппаратуры передачи данных и линий связи, высокие расходы на эксплуатацию сети. Рентабельность подобных сетей достигается только при условии достаточно полной загрузки этих каналов.

При коммутации абонентских пунктов и ЭВМ только на время передачи информации (т.е. нормальным режимом для которых является режим «off-line») принцип построения узла коммутации определяется способами организации прохождения информации в сетях передачи данных. Существуют три основных способа подготовки и передачи информации в сетях, основанных на коммутации: каналов, сообщений и пакетов.

Коммутация каналов. Способ коммутации каналов заключается в установлении физического канала связи для передачи данных непосредственно между абонентами сети. При использовании коммутируемых каналов тракт (путь) передачи данных образуется из самих каналов связи и устройств коммутации, расположенных в узлах связи.

Установление соединения заключается в том, что абонент посылает в канал связи заданный набор символов, прохождение которых по сети через соответствующие узлы коммутации вызывает установку нужного соединения с вызываемым абонентом. Этот транзитный канал образуется в начале сеанса связи, остается фиксированным на период передачи всей информации и разрывается только после завершения передачи информации.

Такой способ соединения используется в основном в сетях, где требуется обеспечить непрерывность передачи сообщений (например, при использовании телефонных каналов связи и абонентского телеграфа). В

этом случае связь абонентов возможна только при условии использования ими однотипной аппаратуры, одинаковых каналов связи, а также единых кодов.

Коммутация сообщений. При коммутации сообщений поступающая на узел связи информация передается в память узла связи, после чего анализируется адрес получателя. В зависимости от занятости требуемого канала сообщение либо передается в память соседнего узла, либо становится в очередь для последующей передачи. Таким образом, способ коммутации сообщений обеспечивает поэтапный характер передачи информации. В этом случае сообщения содержат адресный признак (заголовок), в соответствии с которым осуществляется автоматическая передача информации в сети от абонента-передатчика к абоненту-приемнику. Все функции согласования работы отдельных участков сети связи, а также управление передачей сообщений и их соответствующую обработку выполняют центры (узлы) коммутации сообщений. Основное функциональное назначение центра коммутации сообщений — обеспечить автоматическую передачу информации от абонента к абоненту в соответствии с адресным признаком сообщения и требованиями к качеству и надежности связи.

Метод коммутации сообщений обеспечивает независимость работы отдельных участков сети, что значительно повышает эффективность использования каналов связи при передаче одного и того же объема информации (которая в этом случае может достигать 80 — 90% от максимального значения). В системе с коммутацией сообщений происходит сглаживание несогласованности в пропускной способности каналов и более эффективно реализуется передача многоадресных сообщений (так как не требуется одновременного освобождения всех каналов между узлом-передатчиком и узлом-приемником). Передача информации может производиться в любое время, так как прямая связь абонентов друг с другом необязательна.

Коммутация пакетов. В последние годы появился еще один способ коммутации абонентов сети — так называемая коммутация пакетов. Этот способ сочетает в себе ряд преимуществ методов коммутации каналов и коммутации сообщений. При коммутации пакетов перед началом передачи сообщение разбивается на короткие пакеты фиксированной длины, которые затем передаются по сети. В пункте назначения эти пакеты вновь объединяются в первоначальное сообщение, а так как их длительное хранение в запоминающем устройстве узла связи не предполагается, пакеты передаются от узла к узлу с минимальной задержкой во времени. В этом отношении указанный метод близок методу коммутации каналов.

При коммутации пакетов их фиксированная длина обеспечивает эффективность обработки пакетов, предотвращает блокировку линий связи и значительно уменьшает емкость требуемой промежуточной памяти

узлов связи. Кроме того, сокращается время задержки при передаче информации, т.е. скорость передачи информации превышает аналогичную скорость при методе коммутации сообщений.

К недостаткам метода следует отнести односторонний характер связи между абонентами сети.

Различают два основных типа систем связи с коммутацией пакетов:

в системах *первого типа* устройство коммутации анализирует адрес места назначения каждого принятого пакета и определяет канал, необходимый для передачи информации;

в системах *второго типа* пакеты рассылаются по всем каналам и терминалам, каждый канал (терминал), в свою очередь, проанализировав адрес места назначения пакета и сравнив его с собственным, осуществляет прием и дальнейшую передачу (обработку) пакета либо игнорирует его.

Первый тип систем коммутации пакетов характерен для глобальных сетей с огромным числом каналов связи и терминалов, второй тип применим для сравнительно замкнутых сетей с небольшим числом абонентов.

Сопряжение ЭВМ и устройств в сетях. Существенное влияние на организацию систем обработки данных оказывают технические возможности средств, используемых для сопряжения (комплексирования) ЭВМ и других устройств. Основным элементом сопряжения является *интерфейс*, определяющий число линий, используемых для передачи сигналов и данных, а также способ (алгоритм) передачи информации по линиям связи.

Все интерфейсы, используемые в ВТ и сетях, разделяются на три вида: параллельные, последовательные, связные.

Параллельный интерфейс состоит из большого числа линий, по которым передача данных осуществляется в параллельном коде (обычно в виде 8 — 128 разрядных слов). Параллельный интерфейс обладает большой пропускной способностью: порядка 10^4 — 10^5 бод (бит/с). Столь большие скорости передачи данных обеспечиваются за счет ограниченной длины интерфейса (обычно от нескольких метров до десятков (очень редко до сотни) метров).

Последовательный интерфейс состоит, как правило, из одной линии, данные по которой передаются в последовательном коде. Пропускная способность последовательного интерфейса составляет 10^3 — 10^4 бит/с при длине линии интерфейса от десятков метров до километра.

Связные интерфейсы содержат каналы связи, работа которых обеспечивается аппаратурой передачи данных, повышающей (в основном с помощью специальных физических методов) достоверность передачи данных. Связные интерфейсы обеспечивают передачу данных на любые расстояния, однако с небольшой скоростью (в пределах 10^2 — 10^3 бит/с). Применение связных интерфейсов экономически целесообразно на расстоянии не менее километра.

В многопроцессорных и многомашинных вычислительных системах используются в основном параллельные интерфейсы для сопряжения отдельных устройств в ЭВМ, и только в отдельных случаях применяются последовательные интерфейсы для подключения периферийных устройств. Параллельные интерфейсы обеспечивают в первую очередь передачу сигналов прерывания, а также отдельных слов (команд) и блоков данных между сопрягаемыми ЭВМ и устройствами.

В распределенных системах из-за значительных расстояний между устройствами применяются последовательные и связные интерфейсы, которые исключают возможность передачи отдельных сигналов прерывания между сопрягаемыми устройствами и требуют представления информации в виде операций ввода-вывода.

От организации интерфейсов между устройствами во многом зависит организация программного обеспечения.

Программное обеспечение ВС обеспечивает организацию коллективного доступа к вычислительным и информационным ресурсам сети, динамическое распределение и перераспределение ресурсов сети с целью повышения оперативности обработки информации и максимальной загрузки аппаратных средств, а также в случае отказа и выхода из строя отдельных технических средств и т.д.

Программное обеспечение ВС включает три компонента:

общее программное обеспечение, образуемое базовым ПО отдельных ЭВМ, входящих в состав сети;

специальное программное обеспечение, образованное прикладными программными средствами, отражающими специфику предметной области пользователей при реализации задач управления;

системное сетевое программное обеспечение, представляющее комплекс программных средств, поддерживающих и координирующих взаимодействие всех ресурсов вычислительной сети как единой системы.

Особая роль в ПО ВС отводится системному сетевому программному обеспечению, функции которого реализуются в виде распределенной операционной системы сети.

Операционная система сети включает в себя набор управляющих и обслуживающих программ, обеспечивающих:

межпрограммный метод доступа (возможность организации связи между отдельными прикладными программами комплекса, реализуемыми в различных узлах сети);

доступ отдельных прикладных программ к ресурсам сети (и в первую очередь к устройствам ввода-вывода);

синхронизацию работы прикладных программных средств в условиях их обращения к одному и тому же вычислительному ресурсу;

обмен информацией между программами с использованием сетевых «почтовых ящиков»;

выполнение команд оператора с терминала, подключенного к одному из узлов сети, на каком-либо устройстве, подключенном к другому удаленному узлу вычислительной сети;

удаленный ввод заданий, вводимых с любого терминала, и их выполнение на любой ЭВМ в пакетном или оперативном режиме;

обмен наборами данных (файлами) между ЭВМ сети;

доступ к файлам, хранимым в удаленных ЭВМ, и обработку этих файлов;

защиту данных и вычислительных ресурсов сети от несанкционированного доступа;

выдачу различного рода справок об использовании информационных, программных и технических ресурсов сети;

передачу текстовых сообщений с одного терминала пользователя на другие (электронная почта).

С помощью операционной системы сети:

устанавливается последовательность решения задач пользователя;

задачи пользователя обеспечиваются необходимыми данными, хранящимися в различных узлах сети;

контролируется работоспособность аппаратных и программных средств сети;

обеспечивается плановое и оперативное распределение ресурсов в зависимости от возникающих потребностей различных пользователей вычислительной сети.

Выполняемое с помощью операционной системы сети управление включает: планирование сроков и очередности получения и выдачи информации абонентам; распределение решаемых задач по ЭВМ сети; присвоение приоритетов задачам и выходным сообщениям; изменение конфигурации сети ЭВМ; распределение информационных вычислительных ресурсов сети для решения задач пользователя.

Оперативное управление процессом обработки информации с помощью операционной системы сети помогает организовать: учет выполнения заданий (либо определить причины их невыполнения); выдачу справок о прохождении задач в сети; сбор данных о работах, выполняемых в сети.

ОС отдельных ЭВМ, входящих в состав вычислительной сети, поддерживают потребности пользователей во всех традиционных видах обслуживания: средствах автоматизации программирования и отладки, доступа к пакетам прикладных программ и информации локальных баз данных и т.д.

Электронная почта обеспечивает передачу документов, успешно используется при автоматизации конторских работ. Передача между терминалами сообщений, например фототелеграмм, может также рассматриваться как разновидность электронной почты. Однако для большинства конкретных случаев использование электронной почты предполагает

ет передачу сообщений через специальные «почтовые ящики», между которыми размещаются устройства обработки данных. («Почтовый ящик» — общая область памяти вычислительной сети, предназначенная для записи информации с помощью одной прикладной программы с целью ее дальнейшего использования другими прикладными программами, функционирующими в других узлах сети.) Накопление документов в таких «почтовых ящиках» и возможности их последующей дополнительной обработки имеют следующие преимущества:

отпадает необходимость в пересылке предварительных результатов и промежуточных рабочих материалов;
достаточно просто реализуется конфиденциальная связь, обеспечиваются приоритетность передачи данных, циркуляция документов в сети и другие виды информационной связи.

8.3. ЛОКАЛЬНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ

Прогресс в развитии микропроцессорной техники сделал ее доступной массовому потребителю, а высокая надежность, относительно низкая стоимость, простота общения с пользователем-непрофессионалом в области вычислительной техники послужили основой для организации систем распределенной обработки данных, включающих от десятка до сотен ПЭВМ, объединенных в вычислительные сети. В отличие от вычислительных сетей, создаваемых на базе больших ЭВМ и охватывающих значительную территорию, сети на базе ПЭВМ получили название *локальных*, так как они ориентированы в первую очередь на объединение вычислительных машин и периферийных устройств, сосредоточенных на небольшом пространстве (например, в пределах одного помещения, здания, группы зданий в пределах нескольких километров). Появление локальных вычислительных сетей (ЛВС) позволило значительно повысить эффективность применения ВТ за счет более рационального использования аппаратных, программных и информационных ресурсов вычислительной системы, значительного улучшения эксплуатационных характеристик (в первую очередь повышения надежности) и создания максимальных удобств для работы конечных пользователей.

Сравнительно низкая стоимость, высокая живучесть и простота комплексования эксплуатации ЛВС, оснащенность современными операционными системами различного назначения, высокоскоростными средствами передачи данных, оперативной и внешней памятью большой емкости способствовали их быстрому распространению для автоматизации управленческой деятельности в учреждениях, на предприятиях, а также для создания на их основе информационных, измерительных и управляющих систем автоматизации технологических и производственных процессов. Одной из главных проблем создания локальных вычис-

лительных сетей является проблема аппаратной совместимости ВТ. В настоящее время вычислительные средства ЛВС в основном объединяются с помощью высокоскоростных либо низкоскоростных каналов передачи данных. Такие вычислительные сети получили название *свободно связанных*, так как протекание вычислительных процессов в них может осуществляться асинхронно.

При незначительной удаленности вычислительного оборудования наиболее эффективным средством связи между отдельными аппаратными компонентами ЛВС является последовательный интерфейс. Его достаточно высокая пропускная способность позволяет иметь единственный канал передачи данных — моноканал; при этом работа всей системы осуществляется в режиме мультиплексирования.

Классификация ЛВС. Все множество видов ЛВС можно разделить на четыре группы.

К *первой группе* относятся ЛВС, ориентированные на массового пользователя. Такие ЛВС объединяют в основном персональные ЭВМ с помощью систем передачи данных, имеющих низкую стоимость и обеспечивающих передачу информации на расстояние 100 — 500 м со скоростью 2400 — 19200 бод.

Ко *второй группе* относятся ЛВС, объединяющие, кроме ПЭВМ, микропроцессорную технику, встроенную в технологическое оборудование (средства автоматизации проектирования, обработки документальной информации, кассовые аппараты и т.д.), а также средства электронной почты. Система передачи данных таких ЛВС обеспечивает передачу информации на расстояние до 1 км со скоростью от 19200 бод до 1 Мбод. Стоимость передачи данных в таких сетях примерно на 30% превышает стоимость этих работ в сетях первой группы.

К *третьей группе* относятся ЛВС, объединяющие ПЭВМ, мини-ЭВМ и ЭВМ среднего класса. Эти ЛВС используются для организации управления сложными производственными процессами с применением робототехнических комплексов и гибких автоматизированных модулей, а также для создания крупных систем автоматизации проектирования, систем управления научными исследованиями и т.п. Системы передачи данных в таких ЛВС имеют среднюю стоимость и обеспечивают передачу информации на расстояние до нескольких километров со скоростью 120 Мбод.

Для ЛВС *четвертой группы* характерно объединение в своем составе всех классов ЭВМ. Такие ЛВС применяются в сложных системах управления крупным производством и даже отдельной отраслью: они включают в себя основные элементы всех предыдущих групп ЛВС. В рамках данной группы ЛВС могут применяться различные системы передачи данных, в том числе обеспечивающие передачу информации со скоростью от 10 до 50 Мбод на расстояние до 10 км. По своим функциональным возможностям ЛВС этой группы мало чем отличаются от ре-

гиональных вычислительных сетей, обслуживающих крупные города, районы, области. В своем составе они могут содержать разветвленную сеть соединений между различными абонентами — отправителями и получателями информации.

По топологическим признакам ЛВС делятся на сети следующих типов: с общей шиной, кольцевые, иерархические, радиальные и многосвязные.

В ЛВС с общей шиной (см. рис. 8.1, д) одна из машин служит в качестве системного обслуживающего устройства, обеспечивающего централизованный доступ к общим файлам и базам данных, печатающим устройствам и другим вычислительным ресурсам. ЛВС данного типа приобрели большую популярность благодаря низкой стоимости, высокой гибкости и скорости передачи данных, легкости расширения сети (подключение новых абонентов к сети не сказывается на ее основных характеристиках). К недостаткам шинной топологии следует отнести необходимость использования довольно сложных протоколов и уязвимость в отношении физических повреждений кабеля.

Кольцевая топология (см. рис. 8.1, б) характеризуется тем, что информация по кольцу может передаваться только в одном направлении и все подключенные ПЭВМ могут участвовать в ее приеме и передаче. При этом абонент-получатель должен пометить полученную информацию специальным маркером, иначе могут появиться «заблудившиеся» данные, мешающие нормальной работе сети.

Как последовательная конфигурация кольцо особенно уязвимо в отношении отказов: выход из строя какого-либо сегмента кабеля приводит к прекращению обслуживания всех пользователей. Разработчики ЛВС приложили немало усилий, чтобы справиться с этой проблемой. Защита от повреждений или отказов обеспечивается либо замыканием кольца на обратный (дублирующий) путь, либо переключением на запасное кольцо. И в том, и в другом случае сохраняется общая кольцевая топология.

Иерархическая ЛВС (конфигурация типа «дерево») представляет собой более развитый вариант структуры ЛВС, построенной на основе общей шины (см. рис. 8.1, г). Дерево образуется путем соединения нескольких шин с корневой системой, где размещаются самые важные компоненты ЛВС. Оно обладает необходимой гибкостью для того, чтобы охватить средствами ЛВС несколько этажей в здании или несколько зданий на одной территории, и реализуется, как правило, в сложных системах, насчитывающих десятки и даже сотни абонентов.

Радиальную (звездообразную) конфигурацию (см. рис. 8.1, а) можно рассматривать как дальнейшее развитие структуры «дерево с корнем» с ответвлением к каждому подключенному устройству. В центре сети обычно размещается коммутирующее устройство, обеспечивающее жизнеспособность системы. ЛВС подобной конфигурации находят наиболее частое применение в автоматизированных учрежденческих системах

управления, использующих центральную базу данных. Звездообразные ЛВС, как правило, менее надежны, чем сети с общей шиной или иерархические, но эта проблема решается дублированием аппаратуры центрального узла. К недостаткам можно также отнести значительное потребление кабеля (иногда в несколько раз превышающее расход в аналогичных по возможностям ЛВС с общей шиной или иерархических).

Наиболее сложной и дорогой является *многосвязная топология*, в которой каждый узел связан со всеми другими узлами сети (см. рис. 8.1, в). Эта топология в ЛВС применяется очень редко, в основном там, где требуются исключительно высокие надежность сети и скорость передачи данных.

На практике чаще встречаются *гибридные ЛВС*, приспособленные к требованиям конкретного заказчика и сочетающие фрагменты шинной, звездообразной и других топологий.

Методы доступа в ЛВС. По методам доступа в сети выделяются такие наиболее распространенные сети, как Ethernet, ArcNet, Token Ring.

Метод доступа Ethernet, пользующийся наибольшей популярностью, обеспечивает высокую скорость передачи данных и надежность. Для него используется топология «общая шина», поэтому сообщение, отправляемое одной рабочей станцией, принимается одновременно всеми остальными станциями, подключенными к общей шине. Но поскольку сообщение включает адреса станций отправителя и адресата, то другие станции это сообщение игнорируют. Это метод множественного доступа. При нем перед началом передачи рабочая станция определяет, свободен канал или занят. Если свободен, то станция начинает передачу.

Метод доступа ArcNet получил распространение в силу дешевизны оборудования. Он используется в ЛВС со звездообразной топологией. Одна из ПЭВМ создает специальный маркер (сообщение специального вида), который последовательно передается от одной ПЭВМ к другой. Если станция передает сообщение другой станции, она должна дождаться маркера и добавить к нему сообщение, дополненное адресами отправителя и назначения. Когда пакет дойдет до станции назначения, сообщение будет отделено от маркера и передано станции.

Метод доступа Token Ring рассчитан на кольцевую топологию и также использует маркер, передаваемый от одной станции к другой. Но при нем имеется возможность назначать разные приоритеты разным рабочим станциям. При этом методе маркер перемещается по кольцу, давая последовательно расположенным на нем компьютерам право на передачу. Если компьютер получает пустой маркер, он может заполнить его сообщением кадром любой длины, однако лишь в течение того промежутка времени, который отводит специальный таймер для нахождения маркера в одной точке сети. Кадр перемещается по сети и каждая ПЭВМ регенерирует его, но только принимающая ПЭВМ копирует тот кадр в свою память и отмечает его как принятый, однако не

выводит сам кадр из кольца. Эту функцию выполняет передающий компьютер, когда его сообщение возвращается к нему обратно. Тем самым обеспечивается подтверждение факта передачи сообщения.

Вернемся к вопросу о способах соединения персональных компьютеров в единый вычислительный комплекс. Самый простой из них — соединить компьютеры через последовательные порты. В этом случае имеется возможность копировать файлы с жесткого диска одного компьютера на другой, если воспользоваться программой из операционной оболочки Norton Commander. Для получения прямого доступа к жесткому диску другого компьютера разработаны специальные сетевые платы (адаптеры) и программное обеспечение. В простых локальных сетях функции выполняются не на серверной основе, а по принципу соединения рабочих станций друг с другом, поэтому пользователю можно не приобретать специальные файловые серверы и дорогостоящее сетевое ПО. Каждая ПЭВМ такой сети может выполнять функции как рабочей станции, так и сервера.

В ЛВС с развитой архитектурой функции управления выполняет сетевая операционная система, устанавливаемая на более мощном, чем рабочие станции, компьютере (файловом сервере). Серверные сети делятся на сети среднего класса (до 100 рабочих станций) и мощные (корпоративные), объединяющие до 250 рабочих станций и более. Основным разработчиком сетевых программных продуктов для сервера ЛВС является фирма Novell.

Здесь следует отметить, что наблюдается тенденция ускорения передачи данных до гигабитовых скоростей. К тому же требуется передавать данные типа высококачественного звука, речи и изображения. Все это приводит к постепенному вытеснению таких «старых» ЛВС как Token Ring, ArcNet, но позволяющих использовать новые ИТ. Операционная система Windows NT фирмы Microsoft вытесняет с рынка ОС Unix.

В последние годы большой популярностью стали пользоваться «виртуальные» ЛВС VLAN. Их отличие от обычных ЛВС заключается в том, что они не имеют физических ограничений. Виртуальные ЛВС определяют, какие рабочие станции включаются в физические группы на основе протокольной адресации, что позволяет располагать их в любом месте сети.

Модели взаимодействия в ЛВС. В серверных ЛВС реализованы две модели взаимодействия пользователей с рабочими станциями: модель *файл-сервер* и модель *клиент-сервер*.

В первой модели сервер обеспечивает доступ к файлам базы данных для каждой рабочей станции, и на этом его работа заканчивается. Например, если используется база данных типа файл-сервер, для получения сведений о налогоплательщиках, проживающих на какой-либо конкретной улице Москвы, по сети будет передана вся таблица по территориальному округу, и решать, какие записи в ней удовлетворяют запросу, а какие нет,

приходится самой рабочей станции. Таким образом, работа модели «файл-сервер» приводит к перегрузке сети.

Устранение этих недостатков достигается в модели «клиент-сервер». В этом случае прикладная система делится на две части: внешнюю, обращенную к пользователю и называемую клиентом, и внутреннюю, обслуживающую и называемую сервером. Сервером является машина, обладающая ресурсами и предоставляющая их, а клиентом — потенциальный потребитель этих ресурсов. Роль ресурсов может играть файловая система (файловый сервер), процессор (вычислительный сервер), база данных (сервер базы данных), принтер (принтер-сервер) и др. Так как сервер (или серверы) обслуживает одновременно многих клиентов, то на серверном компьютере должна функционировать многозадачная операционная система.

В модели «клиент-сервер» сервер играет активную роль, ибо его программное обеспечение заставляет сервер «сначала подумать, а потом сделать». Потоки информации, текущие по сети, становятся меньшими, поскольку сервер сначала обрабатывает запросы, а затем посылает клиенту то, в чем он нуждается. Сервер так же контролирует допустимость обращения к записям на индивидуальной основе, что обеспечивает большую безопасность данных.

Если вспомнить сети, созданные на основе больших универсальных ЭВМ, то модель «большая ЭВМ-терминал» и есть модель «клиент-сервер».

В модели «клиент-сервер», созданной на основе ПЭВМ, предлагается следующее:

сеть содержит значительное количество серверов и клиентов;

основу вычислительной системы составляют рабочие станции, каждая из которых функционирует как клиент и запрашивает информацию, которая находится на сервере;

пользователь системы освобожден от необходимости знать, где находится требуемая ему информация, он просто запрашивает то, что ему нужно;

система реализуется в виде открытой архитектуры, объединяющей ЭВМ различных классов и типов с различными системами.

Из действующих ЛВС наибольшее распространение получили сети типа Ethernet.

Сеть Ethernet фирмы Xerox появилась в 1972 г.; топология сети — общая шина; передающая среда — коаксиальный кабель; скорость передачи — 10 Мбит/с; метод доступа — недетерминированный; максимальная теоретически возможная длина сетей — до 6,5 км, а на практике — 11,2 км. Сеть состоит из сегментов, соединенных специальными повторителями для усиления сигналов.

Особенности сети Ethernet фирмы 3COM:

работает на кабелях разного типа (витых проводах, коаксиальных, оптоволоконных);

адаптеры сети сверхнадежные (вероятность ошибки составляет 1 на 55, 70 и 120 лет для разных моделей);

адаптеры сети поддерживают различные сетевые ОС (3+, Novell NetWare, IBM Lan server и др.);

максимальное количество компьютеров сети достигает 1024 единицы; звездообразная топология;

максимальное количество ПЭВМ на сегменте — 100;

максимальная длина сегмента сети — 304 м.

Вариант сети Ethernet фирмы Novell реализован на шинной топологии; максимальная длина сегмента сети составляет — 185 м; максимальное количество компьютеров на сегмент — 30; максимальное количество ПЭВМ сети — 87.

Помимо этих следует упомянуть и другие, имеющие значительную практическую реализацию зарубежные ЛВС: ARCnet фирмы DatapointCorp; PC Network фирмы IBM; Cluster/One и Plan 4000 фирмы Nestar Systems др.

Из отечественных ЛВС назовем такие, как «Руслан», «Эстафета», «КвантС», «Сибирь», «Курьер», «Орбита», «Диалог», «Невод», «Лаура», «Ива», «Хронос» и др.

8.4. ОБЕСПЕЧЕНИЕ ЗАЩИТЫ ИНФОРМАЦИИ В СЕТЯХ

В ВС сосредоточивается информация, исключительное право на пользование которой принадлежит определенным лицам или группам лиц, действующим в порядке личной инициативы или в соответствии с должностными обязанностями. Такая информация должна быть защищена от всех видов постороннего вмешательства: чтения лицами, не имеющими права доступа к информации, и преднамеренного изменения информации. К тому же в ВС должны приниматься меры по защите вычислительных ресурсов сети от их несанкционированного использования, т.е. должен быть исключен доступ к сети лиц, не имеющих на это права. Физическая защита системы и данных может осуществляться только в отношении рабочих ЭВМ и узлов связи и оказывается невозможной для средств передачи, имеющих большую протяженность. По этой причине в ВС должны использоваться средства, исключающие несанкционированный доступ к данным и обеспечивающие их секретность.

Неизбежным следствием борьбы с этой опасностью стали постоянно увеличивающиеся расходы на защиту информации. Например, по оценке немецких экспертов, лишь в 1987 г. в промышленности и учебных заведениях Западной Европы потрачено 1,7 млрд. марок на обеспечение безопасности компьютеров.

Исследования практики функционирования систем обработки данных и вычислительных систем показали, что существует достаточно много возможных направлений утечки информации и путей несанкционированного доступа в системах и сетях. В их числе:

чтение остаточной информации в памяти системы после выполнения санкционированных запросов;

копирование носителей информации и файлов информации с преодолением мер защиты;

маскировка под зарегистрированного пользователя;

маскировка под запрос системы;

использование программных ловушек;

использование недостатков операционной системы;

незаконное подключение к аппаратуре и линиям связи;

злоумышленный вывод из строя механизмов защиты;

внедрение и использование компьютерных вирусов.

Обеспечение безопасности информации в ВС и в автономно работающих ПЭВМ достигается комплексом организационных, организационно-технических, технических и программных мер.

К организационным мерам защиты информации относятся:

ограничение доступа в помещения, в которых происходит подготовка и обработка информации;

допуск к обработке и передаче конфиденциальной информации только проверенных должностных лиц;

хранение магнитных носителей и регистрационных журналов в закрытых для доступа посторонних лиц сейфах;

исключение просмотра посторонними лицами содержания обрабатываемых материалов через дисплей, принтер и т.д.;

использование криптографических кодов при передаче по каналам связи ценной информации;

уничтожение красящих лент, бумаги и иных материалов, содержащих фрагменты ценной информации.

Организационно-технические меры защиты информации включают:

осуществление питания оборудования, обрабатывающего ценную информацию от независимого источника питания или через специальные сетевые фильтры;

установку на дверях помещений кодовых замков;

использования для отображения информации при вводе-выводе жидкокристаллических или плазменных дисплеев, а для получения твердых копий — струйных принтеров и термопринтеров, поскольку дисплей дает такое высокочастотное электромагнитное излучение, что изображение с его экрана можно принимать на расстоянии нескольких сотен километров;

уничтожение информации, хранящейся в ПЗУ и на НЖМД, при списании или отправке ПЭВМ в ремонт;

установка клавиатуры и принтеров на мягкие прокладки с целью снижения возможности снятия информации акустическим способом;

ограничение электромагнитного излучения путем экранирования помещений, где проходит обработка информации, листами из металла или из специальной пластмассы.

Технические средства защиты информации — это системы охраны территорий и помещений с помощью экранирования машинных залов и организации контрольно-пропускных систем. Защита информации в сетях и вычислительных средствах с помощью технических средств реализуется на основе организации доступа к памяти с помощью:

контроля доступа к различным уровням памяти компьютеров;

блокировки данных и ввода ключей;

выделения контрольных битов для записей с целью идентификации и др.

Архитектура программных средств защиты информации и включает:

контроль безопасности, в том числе контроль регистрации вхождения в систему, фиксацию в системном журнале, контроль действий пользователя;

реакцию (в том числе звуковую) на нарушение системы защиты контроля доступа к ресурсам сети;

контроль мандатов доступа;

формальный контроль защищенности операционных систем (базовой общесистемной и сетевой);

контроль алгоритмов защиты;

проверку и подтверждение правильности функционирования технического и программного обеспечения.

Для надежной защиты информации и выявления случаев противоправных действий проводится регистрация работы системы: создаются специальные дневники и протоколы, в которых фиксируются все действия, имеющие отношение к защите информации в системе. Фиксируются время поступления заявки, ее тип, имя пользователя и терминала, с которого инициализируется заявка. При отборе событий, подлежащих регистрации, необходимо иметь в виду, что с ростом количества регистрируемых событий затрудняется просмотр дневника и обнаружение попыток преодоления защиты. В этом случае можно применять программный анализ и фиксировать сомнительные события. Используются также специальные программы для тестирования системы защиты. Периодически или в случайно выбранные моменты времени они проверяют работоспособность аппаратных и программных средств защиты.

К отдельной группе мер по обеспечению сохранности информации и выявлению несанкционированных запросов относятся программы обнаружения нарушений в режиме реального времени. Программы данной группы формируют специальный сигнал при регистрации действий, которые могут привести к противоправным действиям по отношению к за-

щищаемой информации. Сигнал может содержать информацию о характере нарушения, месте его возникновения и другие характеристики. Кроме того, программы могут запретить доступ к защищаемой информации или симулировать такой режим работы (например, моментальная загрузка устройств ввода-вывода), который позволит выявить нарушителя и задержать его соответствующей службой.

Один из распространенных способов защиты — явное указание секретности выводимой информации. В системах, поддерживающих несколько уровней секретности, вывод на экран терминала или печатающего устройства любой единицы информации (например, файла, записи или таблицы) сопровождается специальным грифом с указанием уровня секретности. Это требование реализуется с помощью соответствующих программных средств.

В отдельную группу выделены средства защиты от несанкционированного использования программного обеспечения. Они приобретают особое значение вследствие широкого распространения ПК. Исследования, проведенные зарубежными экспертами, свидетельствуют, что на одну проданную копию оригинальной программы приходится минимум одна нелегальная копия. А для особо популярных программ это соотношение достигает 1:7.

Особое внимание уделяется законодательным средствам, регулирующим использование программных продуктов. В соответствии с Законом Российской Федерации об информации, информатизации и защите информации от 25 января 1995 г. предусматриваются санкции к физическим и юридическим лицам за нелегальное приобретение и использование программных средств.

Большую опасность представляют компьютерные вирусы. *Компьютерный вирус* — это специально написанная небольшая по размерам программа, которая может «приписывать» себя к другим программам (т.е. заражать их), а также выполнять различные нежелательные действия. Программа, внутри которой находится компьютерный вирус, называется зараженной. Когда такая программа начинает работу, то сначала управление получает вирус, который находит и заражает другие программы, а также выполняет ряд вредных действий, в частности «засоряет» активную память, портит файлы и т.д.

Для маскировки вируса его действия по заражению других программ и нанесению вреда могут выполняться не всегда, а при выполнении каких-либо условий. После того как вирус выполнит нужные ему действия, он передает управление той программе, в которой он находится, и она работает как обычно, т.е. внешне работа зараженной программы какое-то время не отличается от работы незараженной программы.

Действия вируса могут выполняться достаточно быстро и без выдачи сообщений, поэтому пользователь часто и не замечает, что компьютер

работает несколько странно. Однако по прошествии некоторого времени на компьютере может происходить следующее:

некоторые программы перестают работать или работают неправильно; на экран выводятся посторонние сообщения, символы, рисунки и т.д.; работа на компьютере существенно замедляется; некоторые файлы оказываются испорченными и т.д.

Многие вирусы устроены так, что при запуске зараженной программы они остаются постоянно (точнее, до перезагрузки DOS) в памяти компьютера и время от времени заражают программы. Кроме того, зараженные программы с данного компьютера могут быть перенесены с помощью дискет или по локальной сети на другие компьютеры.

Если не принимать мер по защите от вируса, то последствия заражения вирусом компьютеров могут быть серьезными. В число средств и методов защиты от компьютерных вирусов входят:

общие средства защиты информации, которые полезны так же, как и страховка от физической порчи машинных дисков, неправильно работающих программ или ошибочных действий пользователя;

профилактические меры, позволяющие уменьшить вероятность заражения вирусом;

специализированные программы для защиты от вирусов.

Комплексное решение вопросов безопасности ВС принято именовать *архитектурой безопасности*, где выделяются угрозы безопасности, службы безопасности и механизмы обеспечения безопасности.

Под *угрозой безопасности* понимается действие или событие, которое может привести к разрушению, искажению или несанкционированному использованию ресурсов сети, включая хранимую и обрабатываемую информацию, а также программные и аппаратные средства.

Угрозы подразделяются на *случайные* (непреднамеренные) и *умышленные*. Источником первых могут быть ошибки в ПО, неправильные (ошибочные) действия пользователей, выход из строя аппаратных средств и др. Умышленные угрозы преследуют цель нанесения ущерба пользователям (абонентам) ВС и подразделяются на активные и пассивные. *Пассивные* угрозы не разрушают информационные ресурсы и не оказывают влияния на функционирование ВС. Их задача — несанкционированно получать информацию. *Активные* угрозы преследуют цель нарушать нормальный процесс функционирования ВС путем разрушения или радиоэлектронного подавления линий связи ВС, вывод из строя ЭВМ или ее ОС, искажение баз данных и т.д. Источниками активных угроз могут быть непосредственные действия физических лиц, злоумышленников, компьютерные вирусы и т.д.

К основным угрозам безопасности относятся: раскрытие конфиденциальной информации, несанкционированное использование ресурсов ВС, отказ от информации.

Создаваемая служба безопасности вычислительной сети призвана обеспечивать:

подтверждение подлинности того, что объект, который предлагает себя в качестве отправителя информации в сети, действительно им является;

целостность информации, выявляя искажения, вставки, повторы и уничтожение данных, передаваемых в сетях, а также последующее восстановление данных;

секретность всех данных, передаваемых по каналам ВС;

нейтрализацию попыток несанкционированного использования ресурсов ЭВМ. При этом контроль доступа может быть либо избирательным, т.е. распространяться только на некоторые виды доступа к ресурсам, например, на обновление информации в базе данных, либо полным;

нейтрализацию угрозы отказа от информации со стороны ее отправителя и/или получателя;

получателя информации доказательствами, которые исключают попытки отправителя отрицать факты передачи указанной информации или ее содержания.

К механизмам обеспечения безопасности относятся: идентификация пользователей, шифрование данных, электронная подпись, управление маршрутизацией и др.

Рассмотрим некоторые из них.

Идентификация пользователей. Для защиты ВС от несанкционированного доступа применяется идентификация пользователей (сообщений), позволяющая устанавливать конкретного пользователя, работающего за терминалом и принимающего либо отправляющего сообщения. Право доступа к определенным вычислительным и информационным ресурсам, программам и наборам данных, а также ВС в целом предоставляется ограниченному контингенту лиц, и система должна идентифицировать пользователей, работающих за терминалами. Идентификация пользователей чаще всего производится с помощью паролей. *Пароль* — совокупность символов, известных подключенному к сети абоненту, — вводится им в начале сеанса взаимодействия с сетью, а иногда и в конце сеанса (в особо ответственных случаях пароль нормального выхода из сети может отличаться от входного). Наконец, система может предусматривать ввод пароля для подтверждения правомочности пользователя через определенные кванты времени. Вычислительная система определяет подлинность пароля и тем самым — пользователя.

Для защиты средств идентификации пользователей от неправомерного использования пароли передаются и сравниваются в зашифрованном виде, а таблицы паролей также хранятся в зашифрованном виде, что исключает возможность прочтения паролей без знания ключей.

Для идентификации пользователей могут использоваться и физические методы, например карточка с магнитным покрытием, на которой записывается персональный идентификатор пользователя, карточки

с встроенным чипом. Для уменьшения риска злоупотреблений карточки, как правило, используются с каким-либо другим способом идентификации пользователя, например с коротким паролем.

Наиболее надежным (хотя и наиболее сложным) является способ идентификации пользователя на основе анализа его индивидуальных параметров: отпечатков пальцев, рисунка линий руки, радужной оболочки глаз и др.

Шифрование данных. Секретность данных обеспечивается методами криптографии, т.е. методами преобразования данных из общепринятой формы в кодированную (шифрование) и обратного преобразования (дешифрование) на основе правил, известных только взаимодействующим абонентам сети. Криптография применяется для защиты передаваемых данных, а также информации, хранимой в базах данных: в пакетах дисков, на гибких дисках и лентах.

Шифрование данных производится по алгоритму, определяющему порядок преобразования исходного текста в зашифрованный текст, а дешифрование — по алгоритму, реализующему обратное преобразование. К криптографическим средствам предъявляются требования сохранения секретности, даже когда известна сущность алгоритмов шифрования-дешифрования. Секретность обеспечивается введением в алгоритмы специальных ключей (кодов). Зашифрованный текст превращается в исходный только в том случае, когда и в процессе шифрования и дешифрования использовался один и тот же ключ. Область значений ключа выбирается столь большой, что практически исключается возможность его определения путем простого перебора возможных значений.

Контрольные вопросы

1. Назовите назначение вычислительных сетей.
2. Дайте классификацию ВС.
3. Изобразите схемы построения ВС.
4. Как осуществляется передача данных в сетях ЭВМ?
5. Что такое коммутация каналов?
6. Перечислите преимущества и недостатки коммутации сообщений и пакетов.
7. Какие Вы знаете виды интерфейсов?
8. В чем назначение операционной системы сети?
9. Расскажите об электронной почте.
10. Какие признаки положены в основу классификации ЛВС?
11. Каким образом организуется обмен информацией в ЛВС?
12. Какие существуют сходства и различия между локальными и глобальными вычислительными сетями?
13. Дайте характеристику методам доступа в ЛВС.
14. В чем состоит отличие моделей «файл-сервер» и «клиент-сервер»?
15. Назовите отечественные и зарубежные ЛВС.

16. Охарактеризуйте основные методы защиты информационно-вычислительных ресурсов сети от несанкционированного доступа.

17. Какие Вы знаете программные средства, применяемые для обеспечения защиты в вычислительных сетях?

18. Что такое компьютерный вирус?

19. Каковы задачи службы безопасности вычислительных сетей?

20. Раскройте содержание механизма обеспечения защиты информации в вычислительных сетях.

ГЛАВА 9

ГЛОБАЛЬНЫЕ ИНФОРМАЦИОННЫЕ СЕТИ. ИНТЕРНЕТ

Компьютер стал недорогим и высокопроизводительным рабочим инструментом. Чем дальше, тем быстрее наш мир приходит к повсеместному использованию ПК и информационных сетей. Среди более чем двухсот глобальных информационных сетей Интернет, как ее часто называют, — это просто Сеть с большой буквы. В настоящее время ПК и Интернет составляют не менее важную инфраструктуру, чем пути сообщения или электрическая сеть. Что дает людям эта новая инфраструктура?

Доступ к информации. Сегодня коммуникации и Интернет нужны для успешной работы в любой отрасли индустрии, торговле, транспорте, образовании, науке. Поэтому страны, использующие высокотехнологичные производства и информационные технологии, закладывают фундамент развития всей своей экономики.

Образование. Любая нация, которая заботится о своем будущем, создает информационную среду непрерывного образования. Оно направлено на формирование новой информационной культуры населения, вхождение российского образования в мировую образовательную систему.

Мощь информационных технологий. Информационные технологии развиваются в несколько раз быстрее любых других технологий. В странах Западной Европы, США и Японии информационные технологии — главная инфраструктура, способствующая росту экономики, создающая новые рабочие места. ПК и Интернет — технологии — ключ к успеху в любом деле. В США и других развитых странах отрасли, взявшие на вооружение информационные технологии, значительно увеличили производительность и стали более конкурентоспособными. Информационные технологии способствуют уменьшению накладных расходов и снижают порог вхождения в рынок. Снижение расходов очень важно, но еще важнее то, что информаци-

онные технологии дают людям возможность, избавившись от рутинной работы, генерировать новые идеи и претворять их в жизнь.

«Электронная нервная система». Хорошо налаженная система электронной почты, совершенная система групповой работы — это не просто удобный способ общения, это электронная нервная система современного предприятия, которая (как и нервная система живого организма) обладает способностью мгновенно реагировать на любые изменения в окружающем мире, анализировать ситуации и помогать принимать быстрые и правильные решения.

Интеллектуальное богатство. Страна, для которой важны интеллектуальные возможности ее людей, должна ценить и защищать интеллектуальную собственность. Имея строгие законы о патентном и авторских правах, государство должно обеспечить нормальную работу производителя программного обеспечения, средств и систем информатики. Пиратство в этой области снижает национальный доход страны в целом.

Методические указания

Изучив материал девятой главы, студенты должны понять, что появление новейших ИТ в среде глобальных сетей открыло дорогу к централизации обработки и использования информации, т.е. осуществляется возврат к централизации ИТ для коллективного использования общих информационных ресурсов.

Студенты должны знать:

- определение понятия «глобальная сеть»;
- основные информационные ресурсы Интернет;
- что такое «протокол»;
- назначение и возможности электронной почты;
- понятия «гипертекст» и «мультимедиа»;
- элементы технологии WWW.

Студенты должны иметь представление о:

- классификации глобальных сетей;
- архитектуре сетевых протоколов;
- функциях и режимах работы электронной почты;
- гипертексте как новой информационной технологии;
- возможностях мультимедиа-технологии;
- Web-технологий.

9.1. ХАРАКТЕРИСТИКА ГЛОБАЛЬНЫХ СЕТЕЙ

Сетевые технологии в настоящее время чрезвычайно разнообразны. К классификации вычислительных и информационных сетей, рассмотренных в гл. 8, добавим классификацию по такому ключевому признаку,

как «охват территории». По охвату территории информационно-вычислительные сети подразделяются на: ЛВС, региональные (территориальные), федеральные и глобальные.

Использование персональных компьютеров (ПК) в составе локальных вычислительных сетей обеспечивает постоянное и оперативное взаимодействие между отдельными пользователями в пределах коммерческой либо научно-производственной структуры. Свое название ЛВС получила за то, что все ее компоненты (ПК, каналы коммуникаций, средства связи) физически размещаются на небольшой территории одной организации или ее отдельных подразделений.

Территориальной (региональной) называют технологию (сеть), компьютеры которой находятся на большом удалении друг от друга, как правило, от десятков до сотен километров. Иногда территориальную сеть называют корпоративной или ведомственной. Такая сеть обеспечивает обмен данными между имеющими доступ к ресурсам сети абонентами по телефонным каналам сети общего назначения, каналам сети «Телекс», а также спутниковым каналам связи. Количество абонентов сети не ограничено. Им гарантируется надежный обмен данными в режиме реального времени, передача факсов и телефонная связь по спутниковым каналам. Территориальные сети строятся по идеологии открытых систем. Их абонентами являются отдельные ПК, ЛВС, телексы, установки, факсимильные и телефонные установки, сетевые элементы (узлы сети связи).

Основная задача федеральной сети — создание магистральной сети передачи данных с коммутацией пакетов и предоставление услуг по передаче данных в реальном масштабе времени широкому кругу пользователей, к числу которых относятся и территориальные сети.

Наконец, глобальные сети обеспечивают возможность общения по переписке и телеконференции. Основная задача глобальной сети — обеспечение абонентам не только доступа к компьютерным ресурсам, но и возможности взаимодействия между собой различных профессиональных групп, рассредоточенных на большой территории.

В 1997 г. в мире зарегистрировано более 200 глобальных сетей, 56 из которых созданы в США, 16 — в Японии. Одна из первых глобальных сетей ARPANet охватывала всю территорию США, часть Европы и Азии. Сеть ARPANet доказала техническую возможность и экономическую целесообразность разработки больших сетей для более эффективного использования ЭВМ и программного обеспечения. В Европе сначала были разработаны и внедрены международные сети, затем национальные сети. В 1972 г. в Вене была создана сеть МИПСА, а в 1979 г. к ней присоединились 17 стран Европы, СССР, США, Канада и Япония. Она была создана для проведения фундаментальных исследовательских работ по проблемам энергетики, продовольствия, сельского хозяйства, здравоохранения и др. Кроме того, она создала технологию, позволяющую всем национальным институтам развивать связь друг с другом.

К созданной в 60-х годах в Ленинграде в Академии наук сети в 1985 г. подключилась региональная сеть «Северо-запад» с академическими центрами Москвы и Риги. В 1980 г. была сдана в эксплуатацию система телеобработки статистической информации (СТОСИ), обслуживающая Главный ВЦ ЦСУ СССР в Москве и республиканские ВЦ в союзных республиках.

В течение последнего десятилетия получают все более широкое развитие *глобальные вычислительные и информационные сети* — уникальный симбиоз компьютеров и коммуникаций. Идет активное включение всех стран во всемирные сетевые структуры. Мировой системой компьютерных коммуникаций ежедневно пользуются более 30 млн. чел. Возрастает потребность в средствах структурирования, накопления, хранения, поиска и передачи информации. Удовлетворению этих потребностей служат информационные сети и их ресурсы. Совместное использование ресурсов сетей (библиотек программ, баз данных, вычислительных мощностей) обеспечивается технологическим комплексом и средствами доступа.

Глобальные сети (Wide Area Network, WAN) — это телекоммуникационные структуры, объединяющие локальные информационные сети, имеющие общий протокол связи, методы подключения и протоколы обмена данными. Каждая из глобальных сетей (Internet, Bitnet, DECnet и др.) организовывалась для определенных целей, а в дальнейшем расширялась за счет подключения локальных сетей, использующих ее услуги и ресурсы.

Крупнейшей глобальной информационной сетью является Internet. Передача данных в этой сети организована на основе протокола Internet — IP (Internet Protocol), представляющего собой описание работы сети, которое включает правила налаживания и поддержания связи в сети, обращения с IP-пакетами и их обработки, описания сетевых пакетов семейства IP. Сеть спроектирована таким образом, что пользователь не имеет никакой информации о конкретной структуре сети. Чтобы послать сообщение по сети, компьютер размещает данные в некий «конверт», называемый, например IP, с указанием конкретного адреса.

Процесс совершенствования сети идет непрерывно, большинство новаций происходит незаметно для пользователей. Любой желающий может получить доступ к сети.

В России подключение к Internet началось в начале 90-х годов. ИАЭ им. Курчатова, МГУ, Госкомвуз, МГУ им. Баумана, НГУ и многие другие научные организации и вузы имеют выход в глобальную сеть.

Архитектура сетевых протоколов TCP/IP, на основе которых построена Internet, предназначена специально для объединенной сети. Сеть может состоять из совершенно разнородных подсетей, соединенных друг с другом шлюзами. В качестве подсетей могут выступать локальные сети (Token Ring, Ethernet, пакетные радиосети и т.п.), национальные, регио-

нальные и специализированные сети, а также другие глобальные сети, например, Bitnet или Sprint. К этим сетям могут подключаться машины разных типов. Каждая из подсетей работает в соответствии со своими специфическими требованиями и имеет свою природу связи, сама разрешает свои внутренние проблемы. Однако предполагается, что подсеть может принять пакет информации и доставить его по указанному в этой подсети адресу. Таким образом, две машины, подключенные к одной подсети, могут напрямую обмениваться пакетами, а если возникает необходимость передать сообщение машине другой подсети, то вступают в силу межсетевые соглашения, для чего подсети используют межсетевой язык — протокол IP. Сообщение передается по цепочке шлюзов и подсетей, пока оно не достигнет нужной подсети, где доставляется непосредственно получателю. Аналогом Internet в России является сеть EUnet/Relcom.

Основная задача Relcom — обеспечить не только доступ к компьютерным ресурсам, но и взаимодействие различных профессиональных групп, рассредоточенных на большой территории.

В настоящее время сеть акционерного общества Relcom является скорее средством общения разработчиков новых решений, чем частью устойчивых общественных структур. Предполагается, что дальнейшее развитие глобальной сети приведет к появлению специализированных сетей, отражающих потребности конкретных групп общения (например, муниципальных, банковских, биржевых сетей для обмена информацией в области науки и образования. Relcom объединяет пользователей почти 2500 организаций, расположенных в более чем 200 городах России и государствах СНГ.

Узловые машины сети осуществляют передачу почтовых сообщений и новостей между регионами и распространение сообщений на своей территории. Пользовательские персональные машины под управлением операционных систем UNDC или MS DOS используют для общения с региональными узлами протокол UUCP. Скорость обмена от 1200 до 9600 бит/с. Крупные региональные центры обмениваются сообщениями со скоростью 19,2 Кбит/с. Используются коммутируемые телефонные линии, специализированная телефонная сеть и выделенные линии, протоколы UUCP и TCP/IP (в зависимости от возможностей физических каналов).

Каждый узел сети является самостоятельным юридическим лицом. Координацию работ по развитию и эксплуатации сети осуществляет акционерное общество Relcom (зарегистрированная торговая марка фирмы — Демос+). Профессиональное взаимодействие сотрудников узлов сети проводится в рамках специальной группы по интересам Ассоциации групп пользователей системы UNDC.

Relcom обеспечивает передачу электронной почты внутри страны и за рубеж абонентам сетей Internet, напрямую в сети EUnet, BITNET,

MCI-mail, CompuServe и др. По соглашению с информационными агентствами пользователь сети Relcom может получать аналитические материалы по коммерческой деятельности, политические и экономические новости, обзоры материалов популярных изданий. Пользователь также может знакомиться с состоянием рынка ценных бумаг, получая предложения и результаты биржевых торгов из разных регионов страны.

В сети Relcom распространяются материалы системы ClariNet, включающей в себя электронные версии различных газет и журналов со всего мира. Это позволяет получить доступ к коммерческим источникам информации, таким, как ClariNews (новости агентства UPI), TechWire (обзоры наиболее значимых событий в области науки, техники и технологии), ClariNews-Biz (анализ экономических показателей, биржевые отчеты, курсы валют и ценных бумаг и др.), NewsBytes (ежедневный электронный журнал, посвященный проблемам компьютерной индустрии).

Осуществляется переход на протоколы более высокого уровня, предоставляется такой вид услуг, как выделенный доступ по соответствующей линии. При таком доступе имеется возможность работы с протоколами ftp, telnet и рядом других протоколов и соответствующим прикладным и системным программным обеспечением.

Для дальнейшего развития услуг сети планируется расширить число информационных источников, организовать специализированные экспертные услуги, обеспечить возможность доставки «электронных» писем с использованием факсимильной связи. Техническое развитие сети прежде всего связывается с повышением пропускной способности каналов связи, широким переходом на протоколы более высокого уровня и расширением сервиса, предоставляемого пользователю.

Другим примером российской глобальной телекоммуникационной системы является сеть «Спринт».

«Спринт» — это система, созданная с целью обмена финансовой и деловой информацией между абонентами сети. Сеть обеспечивает интегрированные решения в области телекоммуникаций, высокую надежность, скорость и мировое качество услуг связи. Официальным поставщиком услуг сети является АО «Спринт-сеть». С 1995 г. «Спринт-сеть» предлагает широкий спектр услуг, таких, как электронная почта, факсимильная, телексная, телетайпная связь, предоставляет доступ к информационным ресурсам и финансовым базам данных абонентам сети, доступ в глобальную сеть Internet, возможность банковских платежей, услуги финансово-информационной системы Reuters, телекоммуникационные услуги в системах платежей на основе пластиковых карточек, осуществляет создание глобальных и частных клиентских сетей. Спринт-сеть имеет свои центры доступа примерно в 150 городах. АО «Спринт-сеть» предоставляет услуги в области передачи данных в России и ежедневно передает несколько десятков гигабайт информации между своими клиентами.

Для обеспечения доступа к глобальным сетям пользователю необходимо осуществить подключение к подсети, используя определенные методы доступа, основанные на взаимосвязи протокола обмена и типа линии связи.

В Интернет приходят за информацией. Ее источником являются ресурсы, расположенные на компьютерах Сети, которые, так же как и на любом не связанном с Сетью персональном компьютере, представляют собой информационные объекты, существующие в виде логически завершенных записей, или файлов. Существуют две важные категории файлов: первая из них — это исполняемые программы, вторая — файлы, содержащие данные всевозможных типов (текст, графику, аудио и видео). Работа с программами, безусловно, требует наиболее серьезных навыков со стороны пользователя, в то время как обращение с текстовыми документами в принципе допускает знание всего одной-единственной программы их просмотра. Естественно, что именно текстовые документы востребованы сегодня в Интернет в наибольшей степени.

Сеть тем не менее открывает пользователю доступ к обоим видам этих ресурсов, если он в состоянии ответить на следующие вопросы:

Как найти нужный информационный объект?

Как его использовать — на удаленной машине или перенести его на свой (локальный) компьютер?

Какими программными средствами сделать его воспринимаемым, т.е. прочитанным, озвученным и т.д.

В силу колоссального объема и разнородности организации информационных ресурсов в Сети возникает ряд естественных проблем. Каждый ресурс имеет структуру определенного типа, базируется на машине со своей операционной системой (платформой) и специальной программой обслуживания доступа к ней — *программой-сервером*. Машину, непрерывно функционирующую в Сети, где исполняется такая программа, также часто называют *сервером*. Само соединение пользователя с сервером происходит с помощью соответствующей программы, запускаемой на его компьютере (*программы-клиента*), и выполняется такое соединение на основе заранее определенного свода правил, или *протокола* взаимодействия между *клиентом* и сервером. Таким образом, для начала работы в Сети необходимо:

1. Иметь какую-либо программу-клиент на своем компьютере.

2. Располагать адресом хотя бы одного сервера (например, из книжного справочника, такого, как знаменитые Желтые страницы Интернет), к которому можно обратиться по протоколу, поддерживаемому собственной программой-клиент.

3. Владеть набором команд, используемых в рамках данного протокола.

Отметим отдельно, что пользователь Интернет может получить доступ к ресурсам других сетей благодаря существованию межсетевых шлю-

зов. Под *шлюзом* (*gateway*) принято понимать специализированный узел (рабочую станцию, компьютер) локальной сети, обеспечивающий доступ других узлов данной локальной сети к внешней сети передачи данных и другим вычислительным сетям. Говоря о *межсетевом шлюзе*, часто подразумевают и аппаратные, и программные средства, обеспечивающие межсетевую связь.

Передача информации в Интернет происходит небольшими порциями данных, имеющими строго определенную структуру и называемыми *пакетами*. Сообщение может быть разбито на несколько пакетов, размер которых может варьироваться, но, как правило, не превышает 1500 байт.

Важнейшим моментом при функционировании Интернет является стандартизированный свод правил передачи пакетов данных в Сети и за ее пределы в рамках меж сетевого обмена, закрепленный базовым транспортным протоколом TCP (Transmission Control Protocol) и межсетевым протоколом IP (Internet Protocol). Протокол TCP дает название всему семейству протоколов TCP/IP, главной задачей которых является объединение в сети пакетных подсетей через шлюзы. Каждая сеть работает по своим собственным законам, однако предполагается, что шлюз может принять пакет из другой сети и доставить его по указанному адресу. Реально, пакет из одной сети передается в другую подсеть через последовательность шлюзов, что становится возможно благодаря реализации во всех узлах сети протокола меж сетевого обмена IP.

Величину потока информации (объем последней измеряется в битах или байтах и единицах, им кратных), прошедшего за определенный промежуток времени через выделенный канал связи, шлюз или другую систему, принято называть *трафиком*.

В Интернет каждой машине (*host-y*) приписан определенный адрес, по которому к ней и осуществляется доступ в рамках одного из стандартных протоколов, причем существует одновременно как числовая адресация (так называемый IP-адрес, состоящий из набора четырех чисел, разделенных точками, например 144. 206.160.32), так и более удобная для восприятия человеком система осмысленных *доменных имен* (например, *apollo.polyn.kiae.su*). Пользователь для обращения к машине может использовать как и ее IP-адрес, так и ее имя. Для упрощения работы в сети используется специальная система DNS (Domain Name System), представляющая собой базу данных, которая обеспечивает преобразования доменных имен компьютеров в числовые IP-адреса, поскольку базовым элементом адресации для семейства протокола TCP/IP являются IP-адреса, а доменная адресация выполняет роль сервиса.

Информационные ресурсы Интернет — это вся совокупность информационных технологий и баз данных, доступных при помощи этих технологий и существующих в режиме постоянного обновления. К их числу относятся, например:

электронная почта;
 система телеконференций Usenet;
 система файловых архивов FTR;
 базы данных WWW;
 базы данных Gopher;
 базы данных WAIS;
 информационные ресурсы LISTSERV;
 справочная служба WHOIS;
 информационные ресурсы TRICKLE;
 поисковые машины Open Text Index, Alta Vista, Yahoo, Lycos и др.

Интернет — это главным образом возможность получить информацию в тот же момент, когда она нужна, т.е. в режиме on-line. Но если нет возможности работать в on-line, то для доступа к услугам большинства информационных серверов Интернет можно воспользоваться электронной почтой, хотя в этом случае все будет происходить не так быстро, как в стандартном режиме telnet, ftp или WWW, о которых будет сказано ниже.

Общий принцип доступа к любому информационному ресурсу через электронную почту заключается в том, что пользователь посылает сообщение почтовому роботу (специальному почтовому серверу), который реализует стандартный доступ к ресурсу и отправляет ответ по почте пользователю (рис. 9.1).

При такой схеме доступа общение между пользователем и почтовым роботом происходит в режиме работы электронной почты, а между почтовым роботом и сервером (ftp, wais или WWW) по протоколу робота этого сервера.

Отметим, что многие информационные ресурсы Сети имеют программы-роботы, которые способны общаться с почтовыми клиентами по схеме, представленной на рис. 9.1. Дадим краткую характеристику этих ресурсов.

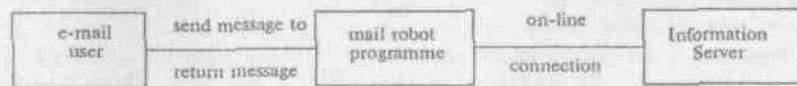


Рис. 9.1. Взаимодействие с информационным ресурсом через электронную почту

Usenet — это система телеконференций Интернет. Система построена по принципу электронных досок объявлений, когда любой пользователь может поместить свою информацию в одну из групп новостей Usenet и эта информация станет доступной другим пользователям, которые на данную группу новостей подписаны. Именно этим способом распространяется большинство сообщений Интернет, например списки наиболее часто задаваемых вопросов (FAQ) или реклама программных про-

дуктов. По Usenet можно получить и вирус, если заказывать и распаковывать все подряд, что приходит на ваш почтовый адрес. Usenet — хорошее место для объявления международных конференций и семинаров.

FTP — система файловых архивов (File Transfer Protocol) — это огромное распределенное (т.е. расположенное на машинах сети, в том числе и функционирующих на разных платформах) хранилище всевозможной информации, накопленной за последние 10—15 лет в Сети. Любой пользователь может воспользоваться услугами анонимного доступа к этому хранилищу и скопировать интересующие его материалы. Объем программного обеспечения в архивах FTP составляет терабайты информации, и ни один пользователь или администратор сети не может просто физически обозреть эту информацию. Кроме программ в FTP-архивах можно найти стандарты Интернет-RFC (Request for Comments), пресс-релизы, книги по различным отраслям знаний, главным образом по компьютерной проблематике, и многое другое. Практически любой архив строится как иерархия директорий. Многие архивы дублируют информацию из других архивов (так называемые «зеркала» — *mirrors*). Для того чтобы получить нужную информацию, вовсе не обязательно ждать, когда информация будет передана из Австралии или Южной Африки, можно поискать «зеркало» где-нибудь ближе, например в Финляндии или Швеции. Для этой цели существует специальная программа *Archie*, которая позволяет просканировать FTP-архивы и найти тот, который устраивает пользователя по составу программного обеспечения и коммуникационным условиям.

World Wide Web — распределенная гипертекстовая информационная система — это последний хит Интернет, темпы развития которого стремительно нарастают. World Wide Web (WWW) представляет удобный доступ к большинству информационных архивов Сети. Особенностью системы является механизм гипертекстовых ссылок, который позволяет просматривать материалы в порядке выбора этих ссылок пользователем. Многие интерфейсы данной технологии позволяют выбирать интересные материалы простым нажатием кнопки манипулятора «мышь» на нужном слове или поле графической картинке. Система универсальных адресов позволяет проадресовать практически все информационные ресурсы Интернет. Многие издательства взяли WWW на вооружение для электронных версий своих журналов. В WWW существует большое количество различного рода каталогов, которые позволяют ориентироваться в сети, кроме этого пользователи могут выполнять даже удаленные программы или смотреть фильмы по сети. Такой сервис не обеспечивается другими информационными системами Интернет.

Gopher — это еще одна распределенная информационная система Интернет. В основу ее интерфейсов положена идея иерархических каталогов. Внешне Gopher выглядит как огромная файловая система, которая расположена на машинах сети. Первоначально Gopher задумывался как информационная система университета с информационными ресурсами

факультетов, кафедр, общежитий и т.п. До сих пор основные информационные ресурсы Gopher сосредоточены в университетах. Gopher считается простой системой, легкой в установке, администрировании, достаточно надежной и защитной. В России Gopher-серверы не так распространены, как во всем мире: профессионалы предпочитают World Wide Web.

WAIS — это распределенная информационно-поисковая система Интернет. Родилась WAIS как перспективная разработка четырех ведущих американских компаний и первое время была коммерческим продуктом, пока не появилась ее свободно-распространяемая версия free WAIS. В основу системы положен принцип поиска информации с использованием логических запросов, основанных на применении ключевых слов. Клиент «обшаривает» все серверы WAIS на предмет наличия в них документов, удовлетворяющих запросу. WAIS широко применяется как поисковая машина в других информационных серверах Интернет, например WWW и Gopher. Наиболее известным проектом, где была применена WAIS, является электронная версия энциклопедии «Британика».

LISTSERV — это, строго говоря, не сервис Интернет, а система почтовых списков сети BIT-NET (сеть образовательных учреждений). Однако это очень популярный ресурс в глобальных компьютерных сетях, и в Интернет существуют шлюзы для доступа к нему. LISTSERV специально ориентирован на применение в качестве транспорта электронной почты. Доступ к нему в интерактивном режиме затруднен. В мире насчитываются многие сотни списков LISTSERV, которые организованы по группам интересов, например существуют группы разработчиков программ ядерно-физических расчетов EGS-4 или группы любителей научной фантастики.

LISTSERV довольно сильно пересекается с Usenet, однако это не мешает существованию как одной, так и другой системы.

WHOIS — служба содержит информацию о пользователях сети, их электронные и обычные адреса, идентификаторы и реальные имена. В последнем случае дается краткое описание основных направлений их деятельности. WHOIS — распределенная система. Это значит, что запросы отправляются по всему множеству серверов WHOIS в Интернет, если только не указан адрес конкретного сервера.

TRICKLE — это доступ по почте к архивам FTP, который организован через специальный шлюз. Этот шлюз имеет специальные навигационные средства для поиска нужной информации в Сети, пользователь может вести с ним своеобразный диалог по почте, выбирая нужную информацию путем ввода специальных команд TRICKLE.

Поисковые машины Open Text Index, AltaVista, Yahoo, Lycos и другие представляют собой мощные информационно-поисковые системы, размещенные на серверах свободного доступа, специальные программы которых непрерывно в автоматическом режиме сканируют информацию

Сети на основе заданных алгоритмов, проводя индексацию документов. В последующем поисковые машины предоставляют пользователю на основе созданных баз данных доступ к распределенной на узлах Сети информации через выполнение поискового запроса в рамках собственного интерфейса.

9.2. ИЕРАРХИЯ ПРОТОКОЛОВ ИНТЕРНЕТ

Иерархия протоколов Интернет. Слово «протокол» в сетевых технологиях имеет смысл, близкий, но несколько отличный от значения, такого, как «документ с записью всего происходящего», приводимого в словаре Ожегова. За ним стоит многозначное понятие, применяемое в разных контекстах, наиболее важным из которых для конечного пользователя является представление о протоколе как о некотором своде четко определенных правил, которые одинаково реализованы в различных системах (программах, шлюзах, пакетах данных и др.). Благодаря этому в местах взаимодействия этих систем, например, при инициировании соединения программы-клиента с программой-сервером или при попадании передаваемого пакета данных на машину-шлюз, все происходит по заранее определенному сценарию.

Пример. Чтобы пояснить понятие протокола, рассмотрим пример, не имеющий отношения к компьютерным сетям, а именно: обсудим взаимодействие двух предприятий *A* и *B*, связанных между собой деловым сотрудничеством (рис. 9.2.). Между предприятиями существуют многочисленные договоренности и соглашения, такие, например, как регулярные поставки продукции одного предприятия другому. В соответствии с этой договоренностью начальник отдела продаж одного из предприятий каждый месяц должен посылать сообщению начальнику отдела закупок второго предприятия о том, сколько и какого товара они могут поставить в этом месяце. В ответ на это сообщение начальник отдела закупок посылает заявку на требуемое количество продукции. (Условленный порядок взаимодействия начальников в данном случае соответствует понятию «протокол уровня начальников».) Начальники посылают свое сообщение и заявки через своих секретарей.

После того как сообщения переданы секретарям, начальников не волнует, каким образом эти сообщения будут перемещаться дальше — обычной или электронной почтой, факсом или нарочным. Выбор способа передачи — это уровень компетенции секретарей, они могут решать этот вопрос, не уведомляя об этом своих начальников, так как их протокол взаимодействия связан только с передачей сообщений, поступающих сверху, и не касается содержания этих сообщений. Отправив письмо, секретари считают свою функцию выполненной, разве что в правила работы хорошего секретаря входит еще и проверка получения сообщения адресатом.

При решении других вопросов начальники могут взаимодействовать по другим правилам, но это не повлияет на работу секретарей, для которых не важно, какие сообщения отправлять, а важно, чтобы они дошли адресату. Итак, в данном случае имеем дело с двумя уровнями — начальниками и секретарями, и каждый уровень имеет собственный протокол, который может быть изменен независимо от протокола другого уровня. Эта независимость протоколов друг от друга и делает привлекательным многоуровневый подход.

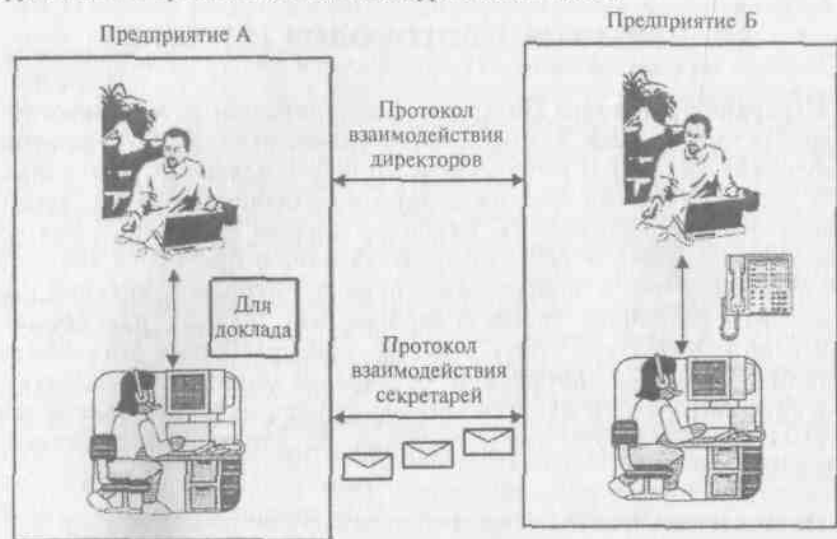


Рис. 9.2. Пример многоуровневого взаимодействия предприятий

Как и в рассмотренном примере, по мере продвижения пакета данных по сети на каждом этапе его взаимодействия с другими сетевыми элементами отрабатывают протоколы разных уровней. Полную совокупность таких протоколов, необходимых для успешного взаимодействия разных элементов в рамках сети данного типа, принято называть *семейством* или *стеком*. Интернет работает под семейством протоколов TCP/IP, которое имеет многоуровневую структуру.

За долгие годы использования в сетях различных стран и организаций стек TCP/IP накопил большое количество протоколов и сервисов прикладного уровня. Проблема соединения разнородных сетей в единую Сеть была решена посредством применения многоуровневой системы протоколов. Так, протокол IP осуществляет минимальные услуги по перемещению данных в Сети (адресация, фрагментация, сборка). TCP, являясь протоколом следующего уровня, обеспечивает надежную доставку данных, а протокол TELNET отвечает за «взаимопонимание» приложений различных компьютеров. Протокол UDP (User Datagram Protocol) также отвечает за «ненадежную» доставку данных, а протокол верхнего

уровня SMTP (Simple Management Transfer Protocol) обеспечивает работу электронной почты, SNMP (Simple Network Management Protocol) — управление сетью, FTP — передачу файлов, NFS — доступ к файловым системам удаленных компьютеров и т.д.

Таким образом, Сеть Интернет можно определить как совокупность ЛВС, удовлетворяющих протоколу TCP/IP, которая имеет общее адресное пространство, где у каждого компьютера есть свой уникальный IP-адрес.

Модель «клиент-сервер» как основа построения информационных сервисов Интернет. В основу взаимодействия компонентов информационных сервисов Сети в большинстве случаев положена модель «клиент-сервер». Как правило, в качестве клиента выступает программа, которая установлена на компьютере пользователя, а в качестве сервера — программа, установленная у провайдера. В данном контексте под *провайдером* понимаем организацию или частное лицо, которые ведут (поддерживают) информационные ресурсы.

При этом возможны два варианта организации самой информационной системы, которая обеспечивает доступ к информационному ресурсу. Большинство систем Интернет построены по принципу взаимодействия «каждый с каждым», например система World Wide Web, т.е. каждый пользователь может напрямую взаимодействовать с каждым сервером без посредников. Такой подход позволяет упростить всю технологическую схему построения системы, однако приводит к порождению большого трафика в Сети. Альтернативный вариант построения системы, например системы Usenet, когда пользователь может взаимодействовать только со «своим» сервером и не может обратиться к произвольному серверу в Сети. Однако доступ он получает ко всей информации, которая присутствует в данной информационной системе, так как серверы обмениваются ею между собой.

Наиболее «древние» услуги Интернет: электронная почта (e-mail), Telnet и FTP. Telnet обеспечивает доступ к БД, каталогам библиотек, другим информационным услугам, создавая ощущение, что Вы сидите за удаленным компьютером и пользуетесь всем, что он имеет, например, программам, или вступаете в беседу. FTP — протокол передачи файлов — позволяет перемещать любые файлы между двумя ПК. Чтобы попасть в открытую для доступа область FTP-сервера, достаточно представиться как анонимный пользователь, посредством FTP можно получить доступ к библиотеке Ватикана, многих университетов и даже послать письмо вице-президенту США. Можно запросить справочные документы, учебники, программы. В Интернет имеются материалы по 4500 темам, доступ к которым обеспечивается посредством системы Usenet.

Система Bitnet создает списки рассылки. Для поиска требуемых материалов можно воспользоваться системой Archie, которая представляет

собой распределенную БД, хранящую каталоги по более чем 1000 хранилищ информации. Система Gopher предоставит меню для поиска.

Для проведения досуга можно воспользоваться услугами системы IRC, а посредством Talk — побеседовать с удаленным пользователем. MUDS (Multi-User Dimension/Dungeons) позволяет войти в мир фантазий.

Так как нет организации, ведущей централизованный учет всех подключенных к Интернет компьютеров, трудно описать ее протяженность и размеры. По оценкам фирмы IBM ее размер достигает 3,2 млн. host-ЭВМ, соединяющих 25 000 сетей и 100 млн. пользователей.

9.3. ЭЛЕКТРОННАЯ ПОЧТА

Технология компьютерного способа пересылки и обработки информации, позволяющая поддерживать оперативную связь между сотрудниками, руководителями, учеными, деловыми людьми и всеми желающими, получила название *электронной почты*. Электронная почта (e-mail) — это специальный пакет программ для хранения и пересылки сообщений между пользователями ЭВМ. Посредством электронной почты реализуется служба безбумажных почтовых отношений. Она является системой сбора, регистрации, обработки и передачи любой информации (текстовых документов, изображений, цифровых данных, звукозаписи и т.д.) по сетям ЭВМ и выполняет такие функции, как редактирование документов перед передачей, их хранение в специальном банке, пересылка корреспонденции, проверка и исправление ошибок, возникающих при передаче, выдача подтверждения о получении корреспонденции адресатом, получение и хранение информации в собственном «почтовом ящике», просмотр полученной корреспонденции.

Электронная почта является услугой вычислительных сетей, и поставщики сетевых операционных систем комплектуют свои продукты средствами поддержки электронной почты.

Электронная почта в локальных сетях обеспечивает передачу документов. Успешно используется при автоматизации конторских работ. При использовании для связи между сотрудниками всего офиса она оказывается удобнее телефона, так как позволяет передавать такую информацию, как отчеты, таблицы, диаграммы и рисунки, которые по телефону передать трудно.

Передача между терминалами сообщений, например фототелеграмм, может также рассматриваться как разновидность электронной почты. Однако для большинства конкретных случаев использование электронной почты предполагает передачу сообщений через специальные «почтовые ящики», между которыми размещаются устройства обработки данных. «Почтовый ящик» — общая область памяти вычислительной сети, предназначенная для записи информации с помощью одной при-

кладной программы с целью ее дальнейшего использования другими прикладными программами, функционирующими в других узлах сети. Почтовый ящик представляет собой специально организованный файл для хранения корреспонденции. Он состоит из двух корзин: отправления и получения. Любой пользователь может обратиться к корзине получения другого пользователя и сбросить туда информацию. Но просмотреть ее он не может. Из корзины отпавлений почтовый сервер забирает информацию для рассылки другим пользователям. Каждый почтовый ящик имеет сетевой адрес. Для пересылки корреспонденции можно установить связь с почтовым ящиком в режиме on-line. Например, в сети Springmail пользователь, зарегистрировавшись и получив определенный статус, по телефонным каналам может входить в ближайший к нему узел сети и общаться с нужными абонентами в режиме on-line. Этот способ неудобен, так как необходимо ждать, пока будет включена ЭВМ получателя.

Более распространенным методом является выделение отдельных компьютеров в качестве почтовых отделений, называемых *почтовыми серверами*. При этом все компьютеры получателей подключены к ближайшему почтовому серверу, получающему, хранящему и пересылающему дальше по сети почтовые отправления, пока они не дойдут до адресата. Отправка адресату осуществляется по мере его выхода на связь с ближайшим почтовым сервером в режиме off-line. Примером может служить сеть Relcom. Пользователь передает сообщение вместе с адресом по телефонному каналу через модем на ближайший почтовый сервер в режиме on-line. Сообщение регистрируется, ставится в очередь и по первому свободному каналу передается на следующий почтовый сервер, пока адресат не заберет его в свой почтовый ящик. Почтовые серверы реализуют следующие функции: обеспечение быстрой и качественной доставки информации, управление сеансом связи, проверку достоверности информации и корректировку ошибок, хранение информации «до востребования» и извещение пользователя о поступившей в его адрес корреспонденции, регистрацию и учет корреспонденции, проверку паролей при запросах корреспонденции, поддержку справочников с адресами пользователей.

Пересылка сообщений пользователю может выполняться в индивидуальном, групповом и общем режимах. В индивидуальном режиме адресатом является отдельный компьютер пользователя, и корреспонденция рассылается одновременно группе адресатов. Почтовые серверы имеют средства распознавания пользователей. В общем режиме корреспонденция отправляется всем пользователям — владельцам почтовых ящиков. Посредством двух последних режимов можно организовать телеконференцию, электронные доски объявлений. Во избежание перегрузки почтовых ящиков в почтовых серверах хранятся справочники адресов, содержащих фильтры для групповых и общих сообщений.

К преимуществам электронной почты относятся скорость и надежность доставки корреспонденции, относительно низкая стоимость услуг, возможность быстро ознакомиться с сообщением широкий круг пользователей.

Любая система электронной почты состоит из двух главных подсистем: клиентского программного обеспечения, с которым непосредственно взаимодействует пользователь;

серверного программного обеспечения, которое управляет приемом сообщения от пользователя-отправителя, передачей сообщения, направлением сообщения в почтовый ящик адресата и его хранением в этом ящике до тех пор, пока пользователь-получатель его оттуда не достанет. Серверное программное обеспечение при совместимости протоколов передачи данных может обрабатывать почту, подготовленную различными клиентскими программами. Это программное обеспечение различается уровнями производительности, надежности, совместимости, устойчивостью к ошибкам, возможностями расширения.

Клиентское программное обеспечение предоставляет пользователям удобные средства для работы с почтой.

Несмотря на их многообразие в различных системах электронной почты, все они имеют общие функции: оповещение о прибытии новой почты; чтение входящей почты; создание исходящей почты; адресация сообщений; использование адресной книги, содержащей список абонентов, которым часто посылают почту; отправка сообщений; обработка сообщений и их сохранение. К обработке сообщений относятся такие функции, как печать, удаление, переадресация письма, сортировка, архивирование сообщений, хранение связанных сообщений. Особо следует выделить программы, позволяющие работать с папками, создавать свои папки для хранения в них сообщений по различным темам. Это очень удобно и помогает быстрее и эффективнее обрабатывать почту.

Различные почтовые программы могут быть классифицированы по разным признакам. Например, в какой операционной системе они могут работать. Сейчас получили наиболее массовое распространение продукты, работающие в ОС Windows 95. Широко используются программы e-mail версии 10.2 компании Демос, МиниХост Интернет Клиент компании Суперфизика, Eudora Light for Windows компании Qualcomm, программы для обработки почты, входящие в состав браузеров Microsoft Internet Explorer 3.0, Netscape Navigator 3.0. Существуют программы для пользователей систем UNIX и OS/2; на старых моделях компьютеров применяются программы, работающие под управлением MS-DOS.

Электронная почта глобальных сетей передачи сообщений, где могут объединяться компьютеры самых различных конфигураций и совместимостей, обеспечивает:

— работу в офлайновом режиме, когда не требуется постоянного присутствия на почтовом узле. Достаточно указать специальной про-

грамме-почтовику (Mailer) время системных события и адреса, где следует забирать почту;

— доступ к телеконференциям (Echo Conference);

— доступ к файловым телеконференциям (File Echo Conference).

Файловые телеконференции отличаются от обычных тем, что в качестве сообщений в них существуют не письма, а файлы. Например, создается файловая телеконференция, посвященная экономике, где каждый участник конференции может поместить свой файл, а другие участники этот файл непременно получают.

Существуют и другие возможности, предоставляемые членам сети. Можно, например, послать заказ на посылку или прием факса. Составляется обычное электронное письмо, оформленное должным образом, и посылается на адрес компьютерного узла, занимающегося факсимильными операциями. Текст этого письма в виде факса будет доставлен на факсимильный аппарат адресата.

Электронная почта в Интернет. Электронная почта является чрезвычайно важным информационным ресурсом Интернет. Помимо того что она представляет собой самое массовое средство электронных коммуникаций, через нее можно принять или послать сообщения еще в два десятка международных компьютерных сетей, часть из которых вообще не имеют on-line сервиса (т.е. прямого подключения к Интернет).

Электронная почта во многом похожа на обычную почтовую службу. Корреспонденция подготавливается пользователем на своем рабочем месте либо программой подготовки почты, либо обычным текстовым редактором. Затем пользователь должен вызвать программу отправки почты (программа подготовки почты вызывает программу отправки автоматически), которая посылает сообщение на почтовый сервер отправителя. Тот в свою очередь посылает его на почтовый сервер адресата, где специальная программа занимается сортировкой почты и рассылкой ее по ящикам конечных пользователей. После запуска программы получения почты адресат устанавливает соединение со своим почтовым сервером и организует пересылку всех полученных на свое имя сообщений. Отметим, что почтовые серверы постоянно подключены к Сети, тогда как компьютеры участников переписки могут устанавливать соединение с ними по мере необходимости. Кроме того, получить и отправить почту можно через разные серверы Интернет. При настройке программы работы с электронной почтой независимо от ее интерфейса необходима следующая информация от провайдера: имя сервера исходящей почты, имя сервера входящей почты, имя пользователя и пароль, а также типы протоколов, используемые при почтовом обмене.

Протокол Simple Mail Transfer Protocol (SMTP). Для работы электронной почты в Интернет специально разработан этот протокол, который является протоколом прикладного уровня и использует транспортный протокол TCP. Однако совместно с этим протоколом используется

и UUCP (Unix — to — Unix-Copy — Program) протокол. UUCP хорошо подходит для использования телефонных линий связи. Разница между SMTP и UUCP заключается в том, что при использовании первого протокола почтового обмена программа, функционирующая на сервере, пытается найти машину получателя почты и установить с ней взаимодействие в режиме on-line для того, чтобы передать почту в ее почтовый ящик. В случае использования SMTP почта достигает почтового ящика получателя за считанные минуты и время получения сообщения зависит только от того, как часто получатель просматривает свой почтовый ящик. При использовании UUCP почта передается по принципу «stop — go», т.е. почтовое сообщение передается по цепочке почтовых серверов от одной машины к другой, пока не достигнет машины-получателя или не будет отвергнута по причине отсутствия абонента-получателя. С одной стороны, UUCP позволяет доставлять почту по плохим телефонным каналам, так как не требуется поддерживать линию все время доставки от отправителя к получателю, а с другой стороны, время доступа к адресату значительно возрастает. В целом же общие рекомендации таковы: если имеется возможность надежно работать в режиме on-line и это является нормой, то следует настраивать почту для работы по протоколу SMTP, если линии связи плохие или on-line используется чрезвычайно редко, то лучше использовать UUCP.

Основой любой почтовой службы является система адресов. Без точного адреса невозможно доставить почту адресату. В Интернет принята система адресов, которая базируется на доменном адресе машины. Например, для пользователя tala машины с адресом citmgu.ru почтовый адрес будет выглядеть так:

tala@citmgu.ru

Таким образом, адрес состоит из двух частей: идентификатора пользователя, который записывается перед знаком «коммерческого эй» — «@», и доменного адреса машины, который записывается после знака «@».

Протокол SMTP был разработан для обмена почтовыми сообщениями в сети Интернет, он не зависит от транспортной среды и может использоваться для доставки почты в сетях с протоколами, отличными от TCP/IP.

Модель протокола. Взаимодействие в рамках SMTP строится по принципу двусторонней связи, которая устанавливается между отправителем и получателем почтового сообщения. При этом отправитель иницирует соединение и посылает запросы на обслуживание, а получатель на эти запросы отвечает. Фактически, отправитель выступает в роли клиента, а получатель — сервера (см. рис. 9.3.).

Канал связи устанавливается непосредственно между отправителем и получателем сообщения. При таком взаимодействии почта достигает абонента в течение нескольких секунд после отправки.

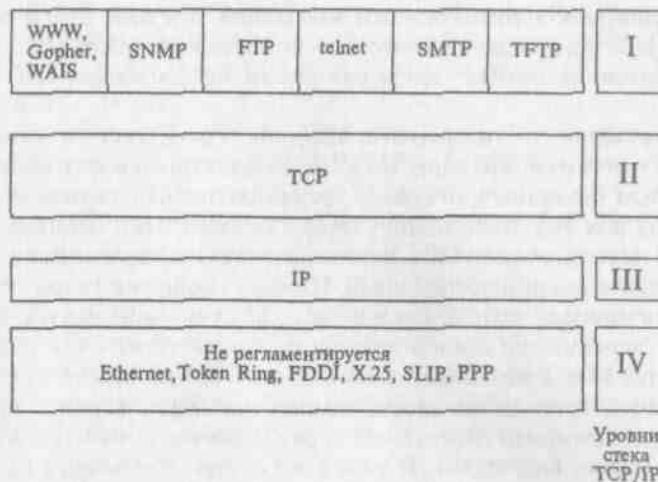


Рис. 9.3. Схема взаимодействия по протоколу SMTP

POP3 (Post Office Protocol, версия 3). Протокол обмена почтовой информацией POP3 предназначен для разбора почты из почтовых ящиков пользователей на их рабочие места при помощи программ-клиентов. Если по протоколу SMTP пользователи отправляют корреспонденцию через Интернет, то по протоколу POP3 они получают корреспонденцию из своих почтовых ящиков на почтовом сервере в локальные файлы (см. § 9.4.).

Такая схема предполагает, что пользователь имеет почтовый ящик на машине-сервере, которая не выключается круглосуточно. Все почтовые сообщения складываются в этот почтовый ящик. По мере необходимости пользователь из своего почтового клиента обращается к почтовому ящику и забирает из него пришедшую на его имя почту. При отправке программа-клиент обращается непосредственно к серверу рассылки почты и передает отправляемые сообщения на этот сервер для дальнейшей рассылки.

9.4. ГИПЕРТЕКСТ

Сетевые технологии существенно повышают интеллектуальные возможности человека, примером этого могут служить две ИТ: гипертекст и мультимедиа. Еще в 1945 г. научный советник президента США Г. Трумена В. Буш, проанализировав способы представления информации в виде отчетов, докладов, проектов, графиков, планов и поняв неэффек-

тивность такого размещения, предложил способ размещения информации по принципу ассоциативного мышления. На базе этого принципа была разработана модель гипотетической машины МЕМЕКС. Через 20 лет Т. Нельсон реализовал этот принцип на ЭВМ и назвал его гипертекстом.

Представление о гипертексте. Внешне гипертекст отличается от обычного текста тем, что часть слов или целые строки в нем, будучи выделены особым шрифтом или цветом, оказываются чувствительными к появлению на них указателя манипулятора «мышь». При попадании на такую область текста указатель (часто стрелочка) изменяет первоначальный вид, становясь, например, ладошкой. Щелчок «мышь» в таком положении приводит к инициированию какого-либо события, чаще всего к загрузке в программу просмотра нового документа, привязанного так называемой гипертекстовой ссылкой к выделенной строке текста. В результате у пользователя появляется возможность самому выбирать порядок просмотра тех или иных страниц, двигаясь по перемежающимся между собой нитям — паутинкам ссылок. Если при этом компьютер подключен к глобальной сети Интернет, то в сценарий просмотра могут входить ресурсы всего мира, доступ к которым происходит по протоколу работы с гипертекстом, или HTTP (Hyper Text Transfer Protocol). После сказанного становится понятным представление об этих ресурсах как о Всемирной паутине (World Wide Web, WWW).

Поскольку нетривиальный характер взаимодействия клиента и сервера по протоколу HTTP с удаленными ресурсами Сети скрыт от конечного пользователя за интерфейсом дружественной программы-просмотра гипертекстовых страниц (броузером, от англ. *browse* — просматривать), начало работы в Web не представляет больших проблем.

Итак, гипертекст не может корректно отображаться обычным текстовым редактором, хотя последний вполне пригоден для его приготовления. Специально разработанный язык гипертекстовой разметки HTML (Hyper Text Markup Language) позволяет превращать нужные элементы документа, включая не только текстовые поля, но и графику, в области «мышечувствительности», или в гипертекстовые ссылки. Существует ряд серьезных причин, по которым необходимо остановиться на этом языке ниже чуть более подробно.

Гипертекст обладает нелинейной сетевой формой организации материала, разделенного на фрагменты — для каждого из которых указан переход к другим фрагментам по определенным типам связей. При установлении связей можно опираться на разные основания (ключи), но в любом случае речь идет о смысловой (семантической) близости связываемых фрагментов. Следуя указанным связям, можно читать или осваивать материал в любом порядке, а не в единственном. Текст теряет свою замкнутость, становится принципиально открытым, в него можно вставлять новые фрагменты, указывая для них связи с имеющимися

фрагментами. Структура текста не разрушается, и вообще у гипертекста нет априорно заданной структуры. Таким образом, *гипертекст — это новая технология представления неструктурированного свободно нарастающего знания.* Этим он отличается от других моделей представления информации.

Таким образом, под *гипертекстом* понимают систему информационных объектов (статей), объединенных между собой направленными связями, образующими сеть. Каждый объект связывается с информационной панелью экрана, на которой пользователь может ассоциативно выбирать одну из связей. Объекты не обязательно должны быть текстовыми, они могут быть графическими, музыкальными, с использованием средств мультимедиа, аудио- и видеотехники. Обработка гипертекста открыла новые возможности освоения информации, качественно отличающиеся от традиционных. Вместо поиска информации по соответствующему *поисковому* ключу гипертекстовая технология предполагает перемещение от одних объектов информации к другим с учетом их смысловой связанности. Обработке информации по правилам формального вывода в гипертекстовой технологии соответствует запоминающие пути перемещения по гипертекстовой сети.

Гипертекстовая технология ориентирована на обработку информации не вместо человека, а вместе с человеком, т.е. *становится авторской.* Удобство ее использования состоит в том, что пользователь сам определяет подход к изучению или созданию материала с учетом своих индивидуальных способностей, знаний, уровня квалификации и подготовки. Гипертекст содержит не только информацию, но и аппарат ее эффективного поиска. По глубине формализации информации гипертекстовая технология занимает промежуточное положение между документальными и фактографическими информационными системами.

Структурно гипертекст состоит из информационного материала, тезауруса гипертекста, списка главных тем и алфавитного словаря.

Информационный материал подразделяется на информационные статьи, состоящие из заголовка статьи и текста. Заголовок содержит тему или наименование описываемого объекта. Информационная статья содержит традиционные определения и понятия, должна занимать одну панель и быть легко обозримой, чтобы пользователь мог понять, стоит ли ее внимательно читать или перейти к другим, близким по смыслу статьям. Текст, включаемый в информационную статью, может сопровождаться пояснениями, примерами, документами, объектами реального мира. Беглый просмотр текста статьи упрощается, если эта вспомогательная информация визуально отличается от основной, например подсвечена или выделена другим шрифтом.

Тезаурус гипертекста — это автоматизированный словарь, отображающий смысловые отношения между лексическими единицами дескрипторного информационно-поискового языка и предназначенный для

поиска слов по их смысловому содержанию. Термин «тезаурус» был введен в XIII в. флорентийцем Брунетто Лотти для названия энциклопедии. С латыни этот термин переводится как сокровище, запас, богатство. Тезаурус гипертекста состоит из тезаурусных статей. Тезаурусная статья имеет заголовок и список заголовков родственных тезаурусных статей, где указаны тип родства и заголовки тезаурусных статей. Заголовок тезаурусной статьи совпадает с наименованием информационной статьи и является наименованием объекта, описание которого содержится в информационной статье. В отличие от традиционных тезаурусов-дескрипторов тезаурус гипертекста содержит не только простые, но и составные наименования объектов. Формирование тезаурусной статьи гипертекста означает индексирование текста.

Полнота связей, отражаемых в тезаурусной статье, и точность установления этих связей, в конечном итоге, определяют полноту и точность поиска при обращении к данной статье гипертекста. Существуют следующие типы родства, или отношений: вид — род, род — вид, предмет — процесс, процесс — предмет, целое — часть, часть — целое, причина — следствие, следствие — причина и т.д. Пользователь получает более общую информацию по родовому типу связи, а по видовому — специфическую информацию без повторения общих сведений из родовых тем. Тем самым глубина индексирования текста зависит от родовидовых отношений. Список заголовков родственных тезаурусных статей представляет собой локальный справочный аппарат, в котором указываются ссылки только на ближайших родственников. Тезаурус гипертекста можно представить в виде сети: в узлах находятся текстовые описания объекта (информационные статьи), ребра сети указывают на существование связи между объектами и на тип родства. В гипертексте поисковый аппарат не делится на тезаурус и массив поисковых образов — документов, как в обычных информационно-поисковых системах. В гипертексте весь поисковый аппарат реализуется как тезаурус гипертекста.

Список главных тем содержит заголовки всех справочных статей, для которых нет ссылок с отношениями ни род — вид, ни часть — целое. Желательно, чтобы список занимал не более одной панели экрана.

Алфавитный словарь содержит перечень наименований всех информационных статей в алфавитном порядке.

Гипертекст используется для предоставления какой-либо информации в виде ссылок на другие темы или документы.

Гипертексты, составленные вручную, используются давно, это справочники, энциклопедии, а также словари, снабженные развитой системой ссылок. Область применения гипертекстовых технологий очень широка: это издательская деятельность, библиотечная работа, обучающие системы, разработка документации, законов, справочных руководств, баз данных, баз знаний и т.д. Наиболее распространенными системами являются HyperCard, HyperStudio, SuperCard, QuickTime фирмы APPLE

для персональных компьютеров Macintosh. Linkway — для IBM: из отечественных — Flexis II, автоматизированная система формирования и обработки гипертекста (АСФОГ) и др. В большинстве современных программных продуктов вся помощь (help) составлена с использованием гипертекстовой технологии на базе меню.

9.5. МУЛЬТИМЕДИА

HyperCard — первый продуманный и удобный авторский инструмент для работы с Multimedia, поскольку имеет аппарат ссылок на видео- и аудиоматериалы, цветную графику, текст с его озвучиванием.

Мультимедиа — это интерактивная технология, обеспечивающая работу с неподвижными изображениями, видеоизображением, анимацией, текстом и звуковым рядом. Одним из первых инструментальных средств создания технологии мультимедиа явилась гипертекстовая технология, которая обеспечивает работу с текстовой информацией, изображением, звуком, речью.

Появлению систем мультимедиа способствовал технический прогресс: возросла оперативная и внешняя память ЭВМ, появились широкие графические возможности ЭВМ, увеличилось качество видеотехники, появились лазерные компакт-диски и др.

Теле-, видео- и большинство аудиоаппаратуры в отличие от компьютеров имеют дело с аналоговым сигналом. Поэтому возникла проблема стыковки разнородной аппаратуры с компьютером и возможности управления ими. Для хранения изображения неподвижной картинке на экране с изображением 512×482 точек (пикселей) требуется 250 Кбайт. При этом качество изображения — низкое. Потребовалась разработка программных и аппаратных методов сжатия и развертки данных. Такие устройства и методы были разработаны с коэффициентом сжатия 100:1 и 160:1. Это позволило на одном компакт-диске разместить около часа полноценного озвученного видео. Наиболее прогрессивными методами сжатия и развертки данных считаются JPEG и MPEG. Были разработаны звуковые платы (Sound Blaster), платы мультимедиа, которые аппаратно реализуют алгоритм перевода аналогового сигнала в дискретный. К компакт-дискам было подсоединено постоянное запоминающее устройство CD-ROM (Compact Disk - Read Only Memory).

В 1988 г. Стив Джобс создал принципиально новый тип персонального компьютера — NeXT, у которого базовые средства систем мультимедиа заложены в архитектуру, аппаратные и программные средства. Были применены новые мощные центральные процессоры 68030 и 68040, процессор обработки сигналов DSP, который обеспечивал обработку звуков, изображений, синтез и распознавание речи, сжатие изображения, работу с цветом. Объем оперативной памяти равнялся 32 Мбайтам, ис-

пользовались стираемые оптические диски, стандартно встроенные сетевые контроллеры, которые позволяют подключаться в сеть, обеспечены методы сжатия, развертки и т.д., объем памяти винчестера — 10,5 Мбайт и 1,4 Гбайт.

Технология работы с NeXT — это новый шаг в общении с машиной. До сих пор работали с интерфейсом SILK (речь, образ, язык, знания). В состав NeXT входит система электронной мультимедиапочты, позволяющая обмениваться сообщениями типа речи, текста, графической информации и т.д.

Многие операционные системы поддерживают технологию мультимедиа: Windows 3.1, DOS 7.0, OS/2. Операционная система Windows-95 включила аппаратные средства поддержки мультимедиа, что позволяет пользователям воспроизводить оцифрованное видео, аудио, анимационную графику, подключать различные музыкальные синтезаторы и инструменты. В Windows-95 разработана специальная версия файловой системы для поддержки высококачественного воспроизведения звука, видео и анимации (файлы произведения звука, видео и анимации). Файлы с мультимедийной информацией хранятся на CD-ROM, жестком диске или на сетевом сервере. Оцифрованное видео обычно хранится в файлах с расширением AVI, аудиоинформация — в файлах с расширением WAV, аудио в форме интерфейса MIDI — в файлах с расширением MID. Для их поддержки разработана файловая подсистема, обеспечивающая передачу информации с CD-ROM с оптимальной скоростью, что существенно при воспроизведении аудио- и видеоинформации.

Даже из такого краткого перечисления возможностей технологии мультимедиа видно, что идет сближение рынка компьютеров, программного обеспечения, потребительских товаров и средств производства того и другого. Наблюдается тенденция развития мультимедиа-акселераторов. *Мультимедиа-акселератор* — программно-аппаратные средства, которые объединяют базовые возможности графических акселераторов с одной или несколькими мультимедийными функциями, требующими обычно установки в компьютер дополнительных устройств. К *мультимедийным функциям* относятся цифровая фильтрация и масштабирование видео, аппаратная цифровая сжатие-развертка видео, ускорение графических операций, связанных с трехмерной графикой (3D), поддержка «живого» видео на мониторе, наличие композитного видеовыхода, вывод TV-сигнала (телевизионного) на монитор. *Графический акселератор* также представляет собой программно-аппаратные средства ускорения графических операций: перенос блока данных, закраска объекта, поддержка аппаратного курсора. Происходит развитие микросхемотехники с целью увеличения производительности электронных устройств и минимизации их геометрических размеров. Микросхемы, выполняющие функции компонентов звуковой платы, объединяются на одной микросхеме — размером со спичечный коробок. И предела этому нет.

К 1990-му году было разработано более 60 пакетов программ с технологией мультимедиа. При этом стандарта не существовало, и в этом же году фирмы Microsoft и IBM одновременно предложили два стандарта. IBM предложила стандарт Ultimedia, а Microsoft — MPC. Остальные фирмы-производители стали разрабатывать пакеты программ на основе этих стандартов. В настоящее время используется стандарт MPC-2, кроме того, разработаны стандарты на приводы CD-ROM, Sound Bluster — звуковые карты, MIDI-интерфейс — стандарт для подключения различных музыкальных синтезаторов, DCI-интерфейс — интерфейс с дисплейными драйверами, позволяющими воспроизводить полноэкранную видеоинформацию, MCI-интерфейс — интерфейс для управления различными мультимедийными устройствами, стандарты на графические адаптеры. Фирма Apple совместно с FujiFilm разработали первый промышленный стандарт IEEE P1394 для разработки набора микросхем Fire Ware, позволяющий оснастить цифровым интерфейсом многие потребительские товары, такие как видеокамера, для использования их в технологии мультимедиа.

Появление систем мультимедиа произвело революцию в таких областях, как образование, компьютерный тренинг, бизнес и в других сферах профессиональной деятельности. Технология мультимедиа создала предпосылки для удовлетворения растущих потребностей общества. Она позволила заменить техноцентрический подход (планирование индустрии зависит от прогноза возможных технологий) на антропоцентрический подход (индустрия управляется рынком). Это дает возможность динамически отслеживать индивидуальные запросы мирового рынка, что отражается в тенденции перехода к мелкосерийному производству. Феномен мультимедиа демократизирует научное, художественное и производственное творчество. Именно авторские технологии обеспечили процесс информатизации общества.

Самое широкое применение технология мультимедиа получила в сфере образования. Созданы видеоэнциклопедии по многим школьным предметам, музеям, городам, маршрутам путешествий. Их число продолжает расти. Созданы игровые ситуационные тренажеры, что сокращает время обучения. Тем самым игровой процесс сливается с обучением, в результате мы имеем Театр обучения, а обучаемый реализует творческое самовыражение. Идет создание базы знаний, в которой сконструированы «живые» миры. Посредством ЭВМ эти базы доступны любому члену человеческого общества. Создается также диалоговое кино, где потребитель может управлять ходом зрелища с клавиатуры дисплея посредством реплик, если к компьютеру подключена плата распознавания речи. Видеоигры дают инструмент манипулирования общественным сознанием: негативом здесь является культ насилия. Технология мультимедиа создает предпосылки для развития «домашней индустрии», что приводит к сокращению производственных площадей, увеличивает

производительность труда. Особенные перспективы открывает Multimedia для дистанционного обучения. Многие вузы в настоящее время занимаются разработками мультимедийных технологий (МГТУ, МГУ, МЭСИ, МЭИ, Ярославский ГУ и др.). Представляет интерес опыт Московского государственного университета экономики, статистики и информатики, где в начале 1994 г. создан центр дистанционного обучения, деятельность которого основана на опыте ведущих учебных заведений Англии, Германии, Голландии, Швеции. Здесь же разрабатывается ряд мультимедийных продуктов.

Как говорится в программистском фольклоре, «сегодня программируется все кроме вкуса и обоняния».

К авторским инструментам относятся QuickTime для Apple, Authoware Professional фирмы MacroMedia, Delphi фирмы Borland и др.

9.6. ТЕХНОЛОГИЯ WORLD WIDE WEB (WWW)

Общая характеристика WWW. В течение последних лет предпринималось немало попыток разработать концепцию универсальной информационной базы данных, в которой можно было бы не только получать информацию из любой точки земного шара, но и иметь удобный способ связи информационных сегментов друг с другом, так чтобы наиболее важные данные быстро могли быть найдены. В 60-е годы исследования в этой области породили понятие «информационной Вселенной» (docuverse = documentation + universe), которая преобразила бы всю информационную деятельность, в частности в области образования. Но только в настоящее время появилась технология, воплотившая эту идею и предоставляющая возможности ее реализации в масштабах планеты.

WWW — это аббревиатура от «World Wide Web» («Всемирная паутина»). Официальное определение World Wide Web звучит как мировая виртуальная файловая система — «широкомасштабная гипермедиа-среда, ориентированная на предоставление универсального доступа к документам».

Проект WWW возник в начале 1989 г. в Европейской Лаборатории физики элементарных частиц (European Laboratory for Particle Physics (CERN) in Geneva, Switzerland). Основное назначение проекта — предоставить пользователям не профессионалам «on-line» доступ к информационным ресурсам. Результатом проекта World Wide Web (WWW, W3) является предоставление пользователям сетевых компьютеров достаточно простого доступа к самой разнообразной информации.

Используя популярный программный интерфейс, проект WWW изменил процесс просмотра и создания информации. Идея заключается в том, что по всему миру хаотично разбросаны тысячи информационных серверов и любую машину, подключенную к Internet в режиме on-line,

можно преобразовать в сервер и начинить его информацией. С любого компьютера, подключенного к Internet, можно свободно установить сетевое соединение с таким сервером и получать от него информацию.

Первый такой сервер был организован в CERN'e, там же с целью развития и поддержки стандартов WWW-технологий создан The World Wide Web Consortium (или W3C). WWW-сервер The W3C's Web site является интегрирующим сервером по поддержке WEB-технологий Internet.

Позднее к проекту подключились и многие другие организации. Большой вклад в развитие WWW-технологий внес Национальный центр суперкомпьютерных приложений (National Centre for Supercomputing Applications — NCSA).

Информационный WWW-сервер использует гипертекстовую технологию. Для записи документов в гипертексте используется специальный, но очень простой язык HTML (Hypertext Markup Language), который позволяет управлять шрифтами, отступами, вставлять цветные иллюстрации, поддерживает вывод звука и анимации. В стандарт языка также входит поддержка математических формул.

Вся польза WWW состоит в создании гипертекстовых документов, и если вас заинтересовал какой-либо пункт в таком документе, то достаточно «ткнуть» в него курсором для получения нужной информации. Также в одном документе возможно делать ссылки на другие, написанные другими авторами или даже расположенные на другом сервере. Одно из главных преимуществ WWW над другими средствами поиска и передачи информации — «многосредность». В WWW можно увидеть на одной странице одновременно текст и изображение, звук и анимацию.

Для удобства ввода информации предусмотрены специальные формы, меню. Программы просмотра позволяют получать доступ не только к WWW серверам, но и к другим службам Internet. С их помощью можно путешествовать по Gopher серверам, искать информацию в WAIS базах, получать файлы с файловых серверов по протоколу FTP. Поддерживается протокол обмена сетевыми новостями Usenet NNTP.

Таким образом, WWW — это новая информационная технология для работы в сети, которая совмещает сетевую технологию с гипертекстом и мультимедиа. WWW реализована на базе сети Интернет. Серверы сети WWW (Web-серверы) могут быть построены на разных платформах.

Web-сервер разбит на Web-страницы посредством языка гипертекстовой разметки HTML. Для перемещения по Web-страницам и передачи гипертекстовых документов (ГТ-документов) по сети разработан протокол HTTP. Для поиска Web-страницы с нужным ГТ-документом разработаны программы поиска и просмотра, называемые навигаторами, или браузеры (Browser). Они обеспечивают интерфейс пользователя со «Всемирной паутиной». При этом стиль оформления экрана и форма представления документа задаются пользователем.

Web-сервер содержит Web-страницы с информацией любого типа (тексты, электронные документы, мультимедийные объекты), редактор разметки HTML, браузеры, программы, обеспечивающие протоколы TCP/IP, HTTP и др., сетевую операционную систему, инструменты для организации дискуссий, гипертекстовые СУБД, системы ГТ-документооборота и многие другие инструменты, обеспечивающие дружественный интерфейс пользователя с Internet.

Web-технология заключается в следующем. Пользователь посредством редактора HTML создает ГТ-документ. Он размещается на Web-сервере. Администратор делает ссылку в каталоге Web-сервера на Web-страницу, чтобы браузер смог ее найти. Любой другой пользователь посредством браузера может получить доступ к данной Web-странице.

Разработано множество браузеров. Примером могут служить Navigator Netscape или Mosaic. Navigator Netscape обеспечивает простоту просмотра WWW-страниц и не требует общей базы данных для разных пользователей. Написанный на языке Java искатель HotJava помимо навигации обеспечивает свободную миграцию программ.

Объектно-ориентированные языки Java и HotJava разработаны фирмой Sun Microsystems. Появление Java произвело переворот в области создания, применения и распространения программ и мультимедийной информации. Программы, написанные на языке Java, можно переносить на разные платформы, т.е. на разные типы компьютеров, выполнять в мультипроцессорных системах, загружать по сети и выполнять программы в машине пользователя, а также загрузить одновременно по сети всем пользователям.

Многие фирмы — производители программных продуктов и технических средств стремятся утвердиться на рынке решений для intranet/internet. Например, фирма Novell делает Web-серверы, Java-приложения. Компания SUN совместно с Insignia Solutions создает Java-совместимый сервер Ntrigne, позволяющий работать с Windows-приложениями. Компания Hewlett-Packard выпустила на рынок радиомодемы, позволяющие реализовать комплекс услуг беспроводной internet-почты.

WWW — это в настоящее время самый популярный и самый интересный сервис Интернет, самое популярное и удобное средство работы с информацией. Самое распространенное имя для компьютера в Интернет сегодня — WWW, больше половины потока данных Интернет приходится на долю WWW. Количество серверов WWW сегодня нельзя оценить сколько-либо точно, но по некоторым оценкам их более 300 тысяч. Скорость роста WWW даже выше, чем у самой сети Интернет.

WWW работает по принципу клиент-сервер, точнее, клиент-серверы: существует множество серверов, которые по запросу клиента возвращают ему гипермедийный документ — документ, состоящий из частей с

разнообразным представлением информации, в котором каждый элемент может являться ссылкой на другой документ или его часть. Ссылки эти в документах WWW организованы таким образом, что каждый информационный ресурс в глобальной сети Интернет однозначно адресуется, и документ, который вы читаете в данный момент, способен ссылаться как на другие документы на этом же сервере, так и на документы (и вообще на ресурсы Интернет) на других компьютерах Интернет. Причем пользователь не замечает этого и работает со всем информационным пространством Интернет как с единым целым. Ссылки WWW указывают не только на документы, специфичные для самой WWW, но и на прочие сервисы и информационные ресурсы Интернет. Более того, большинство программ-клиентов WWW (*browsers*, навигаторы) не просто понимают такие ссылки, но и являются программами-клиентами соответствующих сервисов: ftp, gopher, сетевых новостей Usenet, электронной почты и т.д. Таким образом, программные средства WWW являются универсальными для различных сервисов Интернет, а сама информационная система WWW играет интегрирующую роль.

Контрольные вопросы

1. Что нового дают пользователям глобальные информационные сети?
2. Какие информационные ресурсы содержит Интернет?
3. Дайте характеристику основным ресурсам Интернет.
4. Раскройте понятия «протокол».
5. Объясните иерархию протоколов Интернет.
6. Расскажите о схеме «клиент-сервер» для информационных серверов Интернет.
7. Расшифруйте аббревиатуру WWW.
8. Что такое гипертекст?
9. Как организована информация в WWW?
10. Что такое Web-сервер и Web-страница? Какова структура языка гипертекстовой развертки Web-документов?
11. Как организована электронная почта?
12. Дайте характеристику почтовому серверу.

ЗАКЛЮЧЕНИЕ

Современное общество живет в период, характеризующийся небывалым ростом объема информационных потоков. Это относится как к экономике, так и к социальной сфере. Наибольший объем информации наблюдается в промышленности, торговле, финансово-банковской деятельности. В промышленности рост объема информации обусловлен увеличением объема производства, усложнением выпускаемой продукции, используемых материалов, технологического оборудования, расширением, в результате концентрации и специализации производства, внешних и внутренних связей экономических объектов. Рыночные отношения предъявляют повышенные требования к своевременности, достоверности, полноте информации, без которых немислима эффективная маркетинговая, финансово-кредитная, инвестиционная деятельность.

К известным видам ресурсов — материальным, трудовым, энергетическим, финансовым — прибавился новый, ранее не учитываемый, — информационный. Только на основе своевременного пополнения, накопления, переработки информационного ресурса, т.е. владения достоверной информацией, возможно рациональное управление любой сферой человеческой деятельности, правильное принятие решений. Особенно актуально это для сферы экономики. Применение современных ЭВМ дает возможность переложить трудоемкие операции на автоматические или автоматизированные устройства, которые могут работать со скоростью, превышающей скорость обработки информации человеком в миллионы раз.

Использование ЭВМ приводит к коренной перестройке технологии производства практически во всех отраслях промышленности, коммерческой и финансово-кредитной деятельности и, как следствие, к повышению производительности и улучшению условий труда людей. Именно поэтому современный специалист должен владеть теоретическими знаниями в области информатики и практическими навыками использования вычислительной техники, техники связи и других средств управления.

В условиях перехода от системы жесткого командного распределения ресурсов к рыночным отношениям интенсивно развиваются новые формы организации труда, производственных и межличностных отношений, растет потребность в разнообразной информации и, в частности, в оперативных сведениях коммерческого и правового характера. Все эти изменения требуют, чтобы будущие специалисты-профессионалы, являясь пользователями компьютерных информационных систем, были готовы к работе в новых условиях, владели основами информационной технологии, умели оценивать действия информационных систем, качество обработки, точность, полноту информации, закладываемой в основу принимаемых управленческих решений.

Одно из важных направлений переработки информации — ее вычислительная обработка — реализуется современными ЭВМ. Настоящему состоянию вычислительной техники предшествовал полувековой период, который принято делить на этапы — поколения ЭВМ. Основным показателем, определяющим отнесение конкретной ЭВМ к тому или другому поколению, считается тип ее элементной базы. Смене поколений сопутствовало изменение основных технико-эксплуатационных и экономических характеристик ЭВМ, и в первую очередь таких, как производительность, емкость памяти, надежность, габаритные размеры и стоимость. Важным фактором тенденции развития ЭВМ было и остается стремление разработчиков уменьшить трудоемкость подготовки задач для решения их на машинах, облегчить связь человека с ЭВМ и повысить эффективность использования ЭВМ в целом.

В середине 80-х годов началась разработка ЭВМ пятого поколения на базе сверхбольших интегральных схем (СБИС). Модели машин пятого поколения ориентированы на потоковую архитектуру, на реализацию интеллектуального человеко-машинного интерфейса, обеспечивающего не только системное решение задач, но и способность машины к логическому мышлению, к самообучению, ассоциативной обработке информации и получению логических выводов. Предполагается, что общение человека с ЭВМ будет осуществляться на естественном языке, в том числе и в речевой форме.

Развитие современной микропроцессорной вычислительной техники, интегральных сетей связи, новых информационных технологий привело к бурному подъему индустрии переработки информации, появлению новой науки — информатики. *Информатика* — наука о совокупности процессов получения, передачи, обработки, хранения, представления и распространения информации во всех сферах человеческой деятельности.

Наука информатика охватывает как теоретический аспект — методологию информационной деятельности в условиях массовой компьютеризации, так и практический — информационную технологию эффективного применения комплекса технических средств для конкретных приложений.

Развитие прикладной информатики — технологии использования компьютерной техники для реализации конкретных приложений — прошло три этапа: решение задач прямого счета (1-й этап), создание информационной поддержки принятия решений (2-й этап, предусматривающий использование традиционных экономико-математических методов и моделей для решения экономических и других видов задач) и наконец, современный 3-й этап — поиск методов решения интеллектуальных задач с применением новых информационных и интеллектуальных технологий, созданием экспертных систем, использованием эвристических методов моделирования исследуемых ситуаций, баз данных и знаний, машинного вывода результатов исследования конкретных ситуаций.

Современная материально-техническая база информатики позволяет широко использовать автоматизированные рабочие места (АРМ) специалистов, работающих во всех сферах и на различных уровнях управленческой деятельности, создавать вычислительные системы, которые в пределах специализированной предметной области способны принимать решения на уровне эксперта-профессионала (экспертные системы), и информационно-коммуникационные сети, формируемые на базе ЭВМ и систем передачи данных.

Экспертные информационные системы, банки данных, базы знаний являются мощным средством накопления интеллекта в конкретных сферах человеческой деятельности, способствующим принятию профессионалом управленческих решений.

С появлением микроЭВМ и персональных компьютеров возникли локальные вычислительные сети. Они позволили поднять на качественно новую ступень управление объектами, повысить эффективность использования ЭВМ, поднять качество обрабатываемой информации, реализовать безбумажную технологию, создать новые технологии. Объединение ЛВС и глобальных сетей позволило получить доступ пользователя к мировым информационным ресурсам. Новейшие современные ИТ такие, как WWW-технология, Интернет, интрасети, технология информационных хранилищ, технология распределенных объектов и др., ведут к широчайшему повышению интеллектуальных способностей человека, быстрому росту его творческих возможностей.

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1

КРАТКИЙ АНГЛО-РУССКИЙ СЛОВАРЬ

А

abonent - абонент
 abonent call - вызов абонента
 abstract information - реферативная информация
 abstraction - абстракция
 access - доступ к порции данных
 access permission - разрешение на доступ
 access right - право доступа
 access time - время выборки
 accounting file - учетный файл
 active file - открытый файл
 Ada - Ада
 adapter - адаптер
 adaptive dialog - адаптивный диалог
 addition load - нагрузка
 addition record - добавляемая запись
 address - адрес
 address field - поле адреса
 address mark - метка адреса
 aggregation - агрегирование
 alert information - сигнальная информация
 Algol - Алгол
 algorithm - алгоритм
 algorithmic - алгоритмический
 algorithmic access - алгоритмический доступ
 algorithmic language - алгоритмический язык
 algorithmization - алгоритмизация
 alignment - юстировка
 Almo - Алмо
 alphabet - алфавит
 alphabetic index - алфавитный указатель
 alphabetic word - буквенное слово
 alphanumeric display - алфавитно-цифровое дисплейное устройство
 alphanumeric keyboard - алфавитно-цифровая клавиатура ввода данных

alphanumeric printer - алфавитно-цифровое печатающее устройство
 вычислительной машины
 amount of information - количество информации
 analog computer - аналоговая вычислительная машина
 analog computer speed - быстродействие аналоговой вычислительной машины
 analog computing system - аналоговая вычислительная система
 analog computing system speed - быстродействие аналоговой вычислительной системы
 analog computing technique - аналоговая вычислительная техника
 analog facsimile - аналоговая факсимильная связь
 analog hardware and software means - средства аналоговой вычислительной техники
 analog variable - аналоговая величина
 analytical-synthetic processing of documents - аналитико-синтетическая переработка научно-технической информации
 analyzer - анализатор
 annex memory - буферная память
 answering - ответ
 antivirus - антивирус
 APL - АПЛ
 application program - прикладная программа
 application program package - пакет прикладных программ
 archive database - архивная база данных
 archive file - архив файлов
 archives - архив
 archiving - архивное хранение
 arrangement - систематизирование

array - массив данных
array identifier - идентификатор массива
artificial intelligence - искусственный интеллект
artificial language - искусственный язык
assembler - ассемблер
assembly language - язык ассемблера
assertion - утверждение
associative access - ассоциативный доступ к данным
asynchronous transmission - асинхронная передача
attribute - реквизит
audit-review file - контрольный файл
audit trail - контрольный журнал
authorized access - санкционированный доступ
author right - авторское право
autocode - автокод
autodump - авторазгрузка
automated system of information retrieval - автоматизированная информационно-поисковая система
automatic answering - автоматический ответ
automatic calling - автоматический вызов
automatic classification - автоматическая классификация
automatic data processing - автоматическая обработка данных
automatic graphic input device - автоматическое графическое устройство ввода данных
automatic print reader - читающий автомат
automatic storage - автоматическая память
automation - автоматизация
automatized data processing - автоматизированная обработка данных
automaton - автомат
autoreader - читающий автомат

В

backup file - резервный файл

bank - банк
bank switching - коммутация банков
base - база
Basic - Бейсик
batch-header document - заголовок пакета
batch processing - режим пакетной обработки; пакетная обработка
bibliographic information - библиографическая информация
binary character - двоичный знак
binary code - двоичный код
biological cybernetics - биологическая кибернетика
bionics - бионика
bit - бит
block - блок
block buffer - буфер блоков
block diagram - блок-схема
bootstrap - загрузка
broadband lan - локальная сеть с модулированной передачей
buffer - буфер
buffer storage - буферное запоминающее устройство
bundled software - стандартное программное обеспечение
burst mode - пакетный режим
byte - байт

С

cache memory - *сверхоперативная память*
calling - вызов
calling sequence - соглашения о связях
cancellation - стирание информации
catalog - каталог; каталог наборов данных
catalogue - каталог
certificate of Conformity - сертификат соответствия
certification of conformity - сертификация соответствия
chained list - цепной список данных
chained list search - цепной поиск
chain printer - цепочное печатающее устройство
channel - канал передачи

character - символ; знак
character generator - генератор знаков
character identification - идентификация знака
character printer - буквенно-цифровое печатающее устройство
character recognition - распознавание знаков
character set - набор знаков
checking program - программа контроля
check system - система контроля технических средств аналоговой вычислительной техники
class - класс
classification scheme - классификация
classification system - классификация
closed user group - замкнутая группа пользователей
cluster analysis - кластерный анализ
Cobol - Кобол
code - код
code extension character - знак расширения кода
code translation - декодирование
codification - кодирование
coding - кодирование
coding (programming) automation - автоматизация программирования
coding scheme - код
commercial information - коммерческая информация
communication - взаимодействие; общение; коммуникация
communication channel - канал связи
communication channel capacity - емкость канала связи
communication channel carrying capacity - пропускная способность канала связи
compatible information systems - совместимые информационные системы
compilation - компиляция
complex automation - комплексная автоматизация
computational linguistics - вычислительная лингвистика
computational process - процесс обработки данных

computation centre - вычислительный центр
computer - вычислительная машина
computer-aided instruction - автоматизированное обучение
computer-assisted design - автоматизированное проектирование
computer-assisted learning - автоматизированное обучение
computer channel - машинный канал
computer generation - поколение ЭВМ
computer graphics - машинная графика
computer information - машинная информация
computer-Integrated manufacturing - компьютеризированное интегрированное производство
computerization - компьютеризация
computer literacy - компьютерная грамотность
computer micrographics - машинная микрографика
computer network - вычислительная сеть
computer-oriented language - машиноориентированный язык
computer science - вычислительная техника; информатика
computer system - система обработки данных
computer-system audit - ревизия вычислительной системы
computer system security - защита системы обработки данных
computer time - машинное время
computer virus - компьютерный вирус
computer word - машинное слово
computing centre - вычислительный центр
computing system - система обработки данных
concept - концепция
conception - концепция
conceptual scheme - концептуальная схема базы данных
configuration - конфигурация системы обработки информации
connection - связь
connect (logged-on) time - время сеанса

console - консоль
constant - константа
contact protection - защита контакта
control character - управляющий знак
control flow - управляющая логика
control program - программа управления; управляющая программа
control station - управляющая станция
conversational entry - диалоговый ввод данных
conversational language - диалоговый язык
conversational mode - диалоговый режим
conversational remote Job entry - диалоговый удаленный ввод заданий
conversion - конверсия; перекодирование (в языках программирования)
converter - конвертор языка
convertor - конвертор
coordinate graphics - координатная графика
coordinate indexing - координатное индексирование
coordination - координация
copyright - авторское право
coroutine - подпрограмма
current awareness information - сигнальная информация
current information - текущая информация
cursor - курсор
cybernetics - кибернетика

D

data - данные
data abstraction - абстракция данных
data accessibility - доступность данных
data actuality - актуальность данных
data adequacy - адекватность информации
data aggregate - порция данных
data bank - банк данных
databank administration - администрация банка данных
data base - база данных

database administrator - администратор базы данных
database declaration - описание базы данных
database elaboration - конструирование баз данных
database generation - генерация баз данных
database integrity - целостность базы данных
database load - загрузка базы данных
database maintenance - ведение базы данных
database management system - система управления базами данных
database organization - организация базы данных
database processor - процессор базы данных
database recovery - восстановление баз данных
database scheme - схема базы данных
database state - состояние базы данных
data block - блок данных
data character - символ данных
data check - контроль данных
data circuit - цепь данных
data code - код данных
data coding - кодирование данных
data collection - сбор данных
data collection station - устройство ввода данных
data communication - обмен данными
data concentrator - концентратор данных
data converter - преобразователь данных
data definition language - язык описания данных
data description language - язык описания данных
data design - конструкция данных
data dictionary - словарь (базы) данных
data duplication - дублирование данных
data element - элемент данных
data entity - объект данных

data entry screen - трафарет ввода данных
data field - поле данных
data flow - поток данных
dataflow machine - вычислительная машина, управляемая потоком данных
data format - формат данных
datagram - дейтаграмма
datagram service - функция дейтаграммы
data input - ввод данных
data input station - устройство ввода данных
data integrity - целостность данных
data list - список данных
data management - управление данными
data manipulation language - язык манипулирования данными
data medium - носитель данных
data medium protection device - устройство защиты носителя данных
data message - сообщение данных
data model - модель данных
data network - сеть передачи данных
data organization - организация данных
data origination device - устройство подготовки данных
data origination system - система подготовки данных
data output - вывод данных
data pack - пакет данных
data packing - упаковка данных
data processing - обработка данных
data processing system - система обработки данных
data processing system security - защита системы обработки данных
data protection - защита данных
data quality - качество данных
data reading - считывание данных
data redundancy - избыточность данных
data representation - представление данных
data security - безопасность данных
data selection - выбор данных
data set - набор данных

data set catalog - каталог наборов данных
data set generation - поколение набора данных
data set label - метка набора данных
data set name - имя набора данных
dais sink - приемник данных
data source - источник данных
data station - станция данных
data structure - структура данных
data translation - конвертирование данных
data transmission - передача данных
data transmission channel - канал передачи
data unit - единица данных
data updating - актуализация данных
data validation - проверка данных
data validity - достоверность данных
dead file - потерянный файл
decision support system - информационная модель
declaration - описание
declarative language - декларативный язык
declarative representation - декларативное представление
decoding - декодирование
decompiler - детранслятор
derived data item - производный элемент данных
descriptor - дескриптор
descriptor field - дескрипторное поле
descriptor group - дескрипторная группа
descriptor heading - дескрипторное заглавие
descriptor language - дескрипторный язык
descriptor vocabulary - дескрипторный словарь
design and development records - конструкторская документация
device control character - знак управления устройством
device driver - драйвер устройства
device space - пространство устройства
diagnostic program - диагностическая программа

dictionary - словарь
differential file - файл различий
digital optical disk - оптический диск
digroup - дигруппа
direct access - прямой доступ
direct-access method - прямой метод доступа
direct address - прямой адрес
direct data set - прямой набор данных
direct file - файл прямого доступа
direct memory access - прямой доступ к памяти
disk - диск
disk cartridge - кассета диска
diskette - дискета
disk pack - пакет дисков
dispatching - диспетчеризация
display - видеотерминал; воспроизведение; отображение
display complex - дисплейный комплекс
display console - дисплейный пульт
display device - дисплейное устройство
display facility system - система средств отображения информации
display file - дисплейный файл
display image - визуальное отображение
display means - средство отображения информации
display surface - носитель изображения
display terminal - дисплейный терминал
distributed databank - распределенный банк данных
distributed data base - распределенная база данных
distributed data processing - распределенная обработка данных
distributed file system - распределенная файловая система
distributed system - распределенная (вычислительная) система
document - документ
documentary information - документальная информация
documentation - информационная деятельность

document reader - устройство считывания с документов
document retrieval system - документальная информационно-поисковая система
downward communication - передача информации сверху вниз
driver - драйвер
dump - дамп
duplicate - дубликат
dynamic memory - динамическое запоминающее устройство
dynamic storage - динамическое запоминающее устройство

Е

echoprinting - эхопечать
economic information - экономическая информация
editing - редактирование; редактирование языках программирования
editing terminal - редакторский терминал
editor - редактор
edit session - сеанс редактирования
electronic document - электронный документ
electronic mail - электронная почта
electronics - радиоэлектроника
electronization - электронизация
emulation - эмуляция
encapsulated type - скрытый тип данных
encoding - кодирование
end user - конечный пользователь
enquiry - запрос
error control joint - защитный стык устройств передачи сигнала данных
error detection - обнаружение ошибок; поиск ошибок (в программе)
error log - журнал ошибок
exclusive access - монопольный доступ
execution sequence - последовательность действий
expression - выражение
express information service - экспресс информация

external file - внешний файл
external label - внешняя метка
external noise - внешняя помеха вычислительной машине
external scheme - внешняя схема базы данных
external storage - внешнее запоминающее устройство; внешняя память

F

facet - фасет
facet classification - многоаспектная классификация
faciographic information - фактографическая информация
faciographic retrieval system - фактографическая информационно-поисковая система
false drops - поисковый шум
FAX - телефакс
feedback - обратная связь
fetching - выборка
fetch protection - защита от несанкционированного доступа
file - файл
file copying - копирование файлов
file maintenance - сопровождение файла
file updating - корректировка файла
flatbed plotter - планшетный графопроектор
flat file - плоский файл
flexible disk - гибкий диск
floppy (disk) - гибкий диск
flowchart - блок-схема
flow control - управление потоком данных
flow diagram - блок-схема
flowline - линия связи
formalization - формализация
formalized language - формальный язык
formal language - формальный язык
format - формат
format effector - знак спецификации формата
formatting - форматирование
Fortran - Фортран

forward channel - прямой канал
forward recovery - прямое восстановление
frame - кадр данных
free-running mode - режим свободного доступа
front-end Interface - внешний интерфейс
front end processor - процессор телеобработки данных
functional language - функциональный язык
function (procedure) - функция (в языках программирования)

G

general purpose computer - ЭВМ общего назначения
graphical information - графическая информация
graphic character - графический символ; графический знак
graphic display - графическое дисплейное устройство
graphic display device - графический дисплей
graphic input - графический ввод
graphic input device - графическое устройство ввода данных
graphics editing - редактирование отображаемой информации
group adapter - групповой адаптер

H

hard copy - документальная копия
hard disk - жесткий диск
hard error - постоянная ошибка
hardware - аппаратные средства; технические средства системы обработки информации
hardware for mechanization and automation of information processes - информационная техника
header - заголовок сообщения (блока) данных
heading - рубрика
help - подсказка

heuristic knowledge - эвристические знания
hierarchical access method - иерархический метод доступа
hierarchical addressing - иерархическая адресация
hierarchical classification - иерархическая классификация
hierarchical data base - иерархическая база данных
hierarchical direct access method - иерархический прямой метод доступа
hierarchical network - иерархическая сеть
hierarchical pointer - иерархический указатель
hierarchical sequential access method - иерархический последовательный метод доступа
hierarchical storage - иерархическая память
hierarchical structure - иерархическая структура
hierarchy - иерархия
high-level language - язык высокого уровня
holding - хранение данных
holographies - голография
horizontal communication - передача информации по горизонтали
host computer - главная электронная вычислительная машина

I

identification - идентификация
identifier - идентификатор
identity - идентичность
ideogram - идеограмма
ideographic character - идеограмма
impulse - импульс
incompatible information systems - несовместимые информационные системы
index - индекс; индекс доступа; указатель
indexed file - индексированный файл
indexing - индексирование
indexing depth - глубина индексирования

indexing language - информационно-поисковый язык
industrial information - промышленная информация
informatics - информатика
information - информация
information activities - информационная деятельность
information analysis - информационный анализ
information base - информационная база
information category - информационная категория
information collection - массив информации
information communication - передача информации
information content - количество информации
information content unit - единица количества информации
information dissemination - информирование; распространение информации
information distribution - информирование; распространение информации
information file - массив информации
information infrastructure - информационная инфраструктура
information network - информационная сеть
information output - выдача информации
information process - информационный процесс
information processing system - система обработки информации
information quantity - объем информации
information receiver - приемник информации
information reception - прием информации
information request - информационный запрос
information request formulation - формулирование информационного запроса

information retrieval - поиск информации
information retrieval and search - информационный поиск
information retrieval file - информационно-поисковый массив
information retrieval language - информационно-поисковый язык
information retrieval system - информационно-поисковая система
information sensing - восприятие информации
information service - справочно-информационное обслуживание; информационная служба; информационный сервис
information storage - хранение информации
information system - информационная система
information systems compatibility - совместимость информационных систем
information task - информационная задача
information technology - информационная технология
information theory - теория информации
information transfer - передача информации
information transmission rate - скорость передачи информации
information unit - единица информации
information user - потребитель научно-технической информации
information volume - объем информации
information work - информационная деятельность
informatization - информатизация
initial data - исходные данные
initial information - исходная информация
inked ribbon - красящая лента
inking - рисование
input data - входные данные
input document - входной документ
input information - входящая информация

input-output - ввод-вывод
input-output channel - канал ввода/вывода
input-output device - устройство ввода/вывода данных
input-output processor - процессор ввода/вывода
input-output unit - устройство ввода/вывода данных
input (process) - ввод
input protection - защита ввода
inquiries composition - композиция запросов
insert mode - режим вставки
installation - инсталляция
instruction - команда
instruction register - регистр команд
integrated adapter - интегрированный адаптер
integrated data base - интегрированная база данных
integrated data processing - интегрированная обработка данных
integrated system - интегрированная система
integration - интеграция
intelligence - интеллект
intelligent data base - интеллектуальная база данных
intelligent information - интеллектуальная информация
intelligent system - система искусственного интеллекта
intelligent terminal - интеллектуальный терминал
interaction - взаимодействие
interactive language - диалоговый язык
interest profile - постоянный запрос
interface - взаимодействие: интерфейс
interface data unit - интерфейсный блок данных
interface integrated circuit - интерфейсная интегральная схема
intermediate language - промежуточный язык
internal file - внутренний файл
internal label - внутренняя метка
internal memory - внутреннее запоминающее устройство

internal scheme - внутренняя схема
базы данных
internal storage - внутреннее запоминающее устройство
international information system - международная информационная система
international standard - международный стандарт
International Organization for Standardization (ISO) - Международная организация по стандартизации (ИСО)
international standard publication number - международный стандартный номер издания
interpretation - декодирование
interpreter - интерпретатор
interprocessor link adapter - адаптер межпроцессорной связи
interruption depth - глубина прерывания
interval updating - периодическая актуализация базы данных
inverted file - инвертированный файл
irrelevant information (documents) - поисковый шум
item of information - единица информации

J

job - работа: задание системе обработки информации
job batch - пакет заданий
job file - файл задания
job-recovery control file - резервный файл
job step - пункт задания
journal - журнал
juridical information - юридическая информация

K

key- ключ
keyword - ключевое слово
knowledge - знания
knowledge base - база знаний
knowledge base demon - демон базы знаний

knowledge representation - представление знаний
knowledge representation system - система представления знаний

L

label - метка /в языках программирования/
language - язык
language formalization - формализация языка
language processor - языковой процессор
laser beam printer - лазерное печатающее устройство
laser printer - лазерное печатающее устройство
layout character - знак спецификации формата
legal information - юридическая информация
letter - буква
letter-quality printer - высококачественное печатающее устройство
lexical analysis (scan) - лексический анализ
lexical token - лексическая единица
lexical unit - лексическая единица
library data set - библиотечный набор данных
linear search - линейный поиск
line graphics - координатная графика
linguistics - лингвистика
linkage editor - редактор связей
linked list search - цепной поиск
linker - редактор связей
link file - файл связей
listing - листинг
list processing - обработка списков
load - загрузка в память
load module - загрузочный модуль
local access - локальный доступ
local databank - локальный банк данных
local database - локальная база данных
local storage - локальная память

lock-out - защита
log - журнал
logical data organization - логическая организация данных
logical file - логический файл
logical information (processing) system - информационно-логическая система
logical programming - логическое программирование
logical record - логическая запись
logic design - логическое проектирование
logic diagram - логическая схема
logic element - вентиль
logoff - выход из системы
logout - выход из системы

M

machine dictionary - машинный словарь
machine instruction - машинная команда
machine language - машинный язык
machine-readable document - машиночитаемый документ
machine translation - машинный перевод
magnetic card storage - запоминающее устройство на магнитных картах
magnetic disk - магнитный диск
magnetic disk storage - запоминающее устройство на магнитных дисках
magnetic ink character recognition - распознавание магнитных знаков
magnetic tape - магнитная лента
magnetic tape storage - запоминающее устройство на магнитной ленте
mainframe computer - ЭВМ общего назначения
maintenance - сопровождение
main updating - оперативная актуализация базы данных
management information system - административная информационная система; информационно-управляющая система
man-machine interface - человеко-машинный интерфейс

manual answering - ручной ответ
mark - маркер
marker - маркер; метка
masking - маскирование
master file - основной файл
mechanical dictionary - машинный словарь
memory - память
memory hierarchy - иерархия памяти
menu - меню
message - сообщение
metadata - метаданные
metaknowledge - метазнания
metamodel - метамодель
method - метод
microcomputer - микроЭВМ
microfiche - микрофиша
microfilming - микрофильмирование
micromainframe - супермикроэвм
minicomputer - мини-ЭВМ
misinformation - дезинформация
mixed notation system - смешанная индексация
mnemonic - мнемоника
mnemoscheme - мнемосхема
model - модель
modem - модем
monitor - монитор
monitoring - мониторинг
multiaccess system - система коллективного пользования
multiaspect classification - многоаспектная классификация
multiaspect indexing - многоаспектное индексирование
multicoloured display facility - многоцветное средство отображения информации
multidimensional classification - многоаспектная классификация
multidimensional indexing - многоаспектное индексирование
multilist - мультисписок
multiple access - мультидоступ
multiplexing - мультиплексирование
multiprocessing - мультипроцессирование
multiprocessor - мультипроцессор
multisystem - мультисистема

N

national standard - национальный стандарт
 natural language - естественный язык
 network data base - сетевая база данных
 network data model - сетевая модель данных
 network security - защита вычислительной сети
 network structure - сетевая структура
 network teleprocessing - сетевая телеобработка данных
 noise - коэффициент поискового шума; поисковый шум
 noise coefficient - коэффициент поискового шума
 noise ratio - коэффициент поискового шума; коэффициент шума
 nondestructive reading - считывание без разрушения данных
 nonverbal communication - обмен невербальной информацией
 noology - ноология
 normative document - нормативный документ
 null string - пустая строка

O

object description language (for simulation) - язык описания объекта моделирования
 object file - объектный файл
 object language - объектный язык
 object module - объектный модуль
 object of information - объект информации
 object-oriented language - объектно-ориентированный язык
 object-oriented programming - объектно-ориентированное программирование
 object program - объектная программа
 octed - байт
 off-line data processing - автономная обработка данных

off-line storage - автономное хранилище данных
 one-way communication - симплексная связь
 on-line data - данные памяти вычислительной системы
 open system - открытая система обработки данных
 operating storage - оперативная память
 operating system - операционная система
 operation abstraction - абстракция операций
 operation on file - операция над файлом
 operator console - пульт оператора
 operator precedence - приоритет операций
 optical disk - оптический диск
 optical entry - оптический вход
 optical scanner - оптическое сканирующее устройство
 optimization - оптимизация
 order - упорядоченность
 ordering - систематизирование
 original copy - оригинал
 output document - выходной документ
 output (process) - вывод
 output queue - очередь вывода
 output stream - выходной поток

P

packet - пакет
 packet-mode terminal - пакетный терминал
 packet switching - коммутация пакетов
 paging - листание
 paperless office - безбумажное дело-производство
 paperless technology - безбумажная технология
 paradigmatic relations - парадигматические отношения
 parallel computer - параллельная ЭВМ
 parallel interface - параллельный интерфейс
 parallel search - параллельный поиск
 parameter passing - передача параметров

part of text following a rubric - рубрика
 Pascal - Паскаль
 password - пароль
 pattern recognition - распознавание образов
 peripheral device - периферийное оборудование
 peripheral devices - внешние устройства ЭВМ; периферийные устройства ЭВМ
 peripheral equipment - внешнее оборудование; периферийное оборудование
 permanent request - постоянный запрос
 personal computer - персональная ЭВМ
 personal data base - личная база данных
 pertinence - pertinентность
 pertinent information - релевантная информация
 physical data organization - физическая организация данных
 pictorial information - графическая информация
 Planner - Пизнер
 planning - оргтехника
 playback - воспроизведение информации
 plotter - графопостроитель
 P1/I - ПЛ/1
 political information - политическая информация
 polled network - сеть с опросом
 potydimensional classification - многоаспектная классификация
 position - позиция
 positioning - юстировка
 postamble - заключительная часть сообщения
 potential information users - потенциальные потребители информации
 preamble - заголовок
 presentation layer - уровень представления данных
 presentation payer protocol - протокол представления данных
 primary document - первичный документ
 primary information - первичная информация

printer - печатающее устройство
 priority - приоритет процесса
 priority system - система приоритетов
 privacy protection - защита собственности
 private data - приватные данные; частные данные
 problem-oriented language - проблемно-ориентированный язык
 procedural language - процедурный язык
 procedure - процедура
 process - процесс; процесс обработки данных
 process control - управление процессом
 processing unit - устройство обработки данных
 processor - процессор
 production information - производственная информация
 program - программа; программа (в языках программирования); программа общего назначения
 program check-out - проверка программы
 program document - программный документ
 program driver - программный драйвер
 program language alphabet - алфавит языка программирования
 program library - библиотека программ
 programmer - программист
 programming - программирование
 programming language - язык программирования
 programming system - система программирования
 program module - программный модуль
 program product - программное изделие
 program product maintenance - сопровождение программного изделия
 program setting - настройка программы
 program (source) listing - распечатка программы

program validation - аттестация программы
program verification - верификация программы
Prolog - Пролог
protection - защита
protocol - протокол
protocol data unit - протокольный блок данных
pseudolanguage - псевдоязык
pulse - импульс
pulse string - серия импульсов
pulse train - серия импульсов

Q

qualifier - квалификатор
quantity of information - количество информации
question-answering system - вопросно-ответная система
queued sequential access method - последовательный метод доступа с очередями
queued telecommunication access method - телекоммуникационный метод доступа с очередями

R

random access - произвольный доступ к данным
random access memory - оперативное запоминающее устройство
rapid information - оперативная информация
raster display device - растровое устройство отображения
raster graphics - растровая графика
raster plotter - растровый графопроектор
readback - эхосчитывание
reading - считывание
read out - считывание
read-write cycle time - время цикла считывания-записи
real time processing - режим реального времени

reconstitution of data - воспроизведение данных
reconstruction of data - воспроизведение данных
record - запись
recording - запись информации; записывание
record length - длина записи
record size - длина записи
recovery time - время восстановления
redundant information - избыточная информация
reel - бобина
reenterable program - реентерабельная программа
reference - ссылка
reference to standards (in regulations) - ссылка на стандарты (в регламентах)
refresh - регенерация данных
refresh buffer - буфер изображения
refreshed display - дисплейное устройство с регенерацией изображения
regional standard - региональный стандарт
register - регистр
registration - записывание
relational data base - реляционная база данных
relational data model - реляционная модель данных
relative indexing - многоаспектное индексирование
relevance - релевантность
relevant document - релевантный документ
relevant information - релевантная информация
remote-access data processing - дистанционная обработка данных
remote batch entry - дистанционный пакетный ввод
remote batch terminal - терминал пакетной обработки
remote file - дистанционный файл
remote job entry - дистанционный ввод заданий
replay - воспроизведение информации
report - сообщение

reproduction - воспроизведение информации
resolution - разрешающая способность средства отображения информации
resource - ресурс системы обработки информации
resource allocation - ресурсное распределение памяти
resources - ресурсы
response time - время доставки сообщения
retention period - период сохранности
retrieval - выборка; поиск
retrieval prescription - поисковое предписание; поисковый образ запроса
retrospective information - ретроспективная информация
retrospective retrieval - ретроспективный поиск
reverse clipping - экранирование
rotational delay - время поиска
routine - программа общего назначения
rubricator - рубрикатор
russification - русификация
R & T Information support - информационное обеспечение научно-технических разработок

S

scaling - масштабирование (в машинной графике)
scanner - сканирующее устройство
scanning - сканирование; просмотр
scientific activity - научная деятельность
scientific and technical information - научно-техническая информация
scientific information - научная информация
scientific information activities - научно-информационная деятельность
scientific research automation - автоматизация научных исследований

scientific studies - научная деятельность
screen - экран средства отображения информации
screen editor - экранный редактор
searching - поиск
search logic - логика поиска
search pattern of document - поисковый образ документа
search request formulation - поисковый образ запроса; поисковое предписание
search specification - поисковое предписание; поисковый образ запроса
search time - время поиска
secondary information - вторичная информация
secondary station - вторичная станция сетевой телеобработки данных
security - защита; защита от несанкционированного доступа
segment file - файл сегментов
selecting - выбор
selective dissemination of information - избирательное распространение информации
selective information - выборочная информация
semantic analysis - семантический анализ
semantic database - семантическая база данных
semiotic system - знаковая система
sensing - считывание
sequential access - последовательная выборка; последовательный доступ к данным
sequential data set - последовательный набор данных
sequential file - последовательный файл
sequential search - линейный поиск
serial access - последовательная выборка
serial printer - буквенно-цифровое печатающее устройство
service - сервис
service data unit - сервисный блок данных

service routine program - обслуживающая программа
session - сеанс работы
set - множество
shareable file - файл совместного доступа
shared access - коллективный доступ
shielding - экранирование
sign - знак
signal regeneration - регенерация сигнала
sign-off - выход из системы
sign system - знаковая система
simplex transmission - симплексная передача
SIMSCRIPT - СИМСКРИПТ
simulation - машинное моделирование; моделирование
simulator - модель
single request - разовый запрос
SNOBOL - СНОБОЛ
software - программное обеспечение
software adaptation - адаптация программных средств
solving node - режим решения
sorting - сортировка
source data - исходные данные
source information - входящая информация; исходная информация
source language - исходный язык
source of information - источник информации
source program - исходная программа
space character - знак пробела
special character - специальный знак
species - вид
specification language - язык спецификаций
specific concept - видовое понятие
spelling corrector - орфографический корректор
split-screen - полнэкран
standard - стандарт
standardization - стандартизация
standard request - стандартный запрос
statement - оператор
storage - память данных; хранение
storage area - среда хранения
storage capacity - емкость запоминающего устройства

storage display - дисплейное устройство с запоминанием изображения
storage hierarchy - иерархия памяти
storage item - единица хранения
storage protection - защита запоминающего устройства
storage size - емкость запоминающего устройства
storage space - пространство памяти
storage unit - запоминающее устройство
storing - запоминание
strategy of retrieval - стратегия поиска
structured programming - структурное программирование
subclass - подкласс
subject information - тематическая информация
subprogram - подпрограмма
subscriber - абонент; пользователь
subscriber system - абонентская система обработки данных
subscripting - индексация
subsystem - подсистема
supercomputer - супер-ЭВМ
superfluous information - избыточная информация
supermini - супермини-ЭВМ
supervisor - супервизор
support - обеспечение
survey information - обзорная информация
symbol - символ
symbol (character alphabetic binary digit) string - строка символов
symbolic address - символический адрес
synchronization - синхронизация
syntagmatic relations - синтагматические отношения
syntax analysis - синтаксический анализ
system - система
systematization - систематизирование
system for selective dissemination of information - система избирательного распространения информации
system load - нагрузка системы

system of subject heading - рубрикация
systemotechnique - системотехника
system process - системный процесс
system program - системная программа
system reader (task) - процесс системного ввода
system saturation - насыщение системы
system task - системный процесс
system teleprocessing - системная телеобработка данных
system writer (task) - процесс системного вывода

T

table - таблица
tape - лента
tape file - ленточный файл
target language - выходной язык
task - задача
technique cybernetics - техническая кибернетика
technology - технология
telecode communication - телекодовая связь
telecommunication - телекоммуникация
telecommunication access - теледоступ
teleprocessing - телеобработка данных
teleprocessing system - система телеобработки данных
teletype - телетайп
temporary file - рабочий файл
terminal - абонентский пункт; терминал
test data - контрольные данные
text editor - текстовый редактор
text file - текстовый файл
text processing - обработка текстов
thesaurus - информационно-поисковый тезаурус; тезаурус
time sharing - мультиплексный режим
time scale - масштаб времени
tolerance - толерантность
topical information - тематическая информация

touch screen - сенсорный экран
touch sensitive screen - сенсорный экран
transfer time - время передачи данных
translation - трансляция программы
translator - транслятор
transmission control character - знак управления передачей
turnaround time - длительность цикла обработки
turnround time - длительность цикла обработки
two-way simultaneous communication - дуплексная связь

U

underlying information - исходная информация
uniform referencing - унифицированная ссылка
universal decimal classification - универсальная десятичная классификация
user - абонент; пользователь; пользователь системы обработки информации
user authorization file - файл информации о пользователях
userid - идентификатор пользователя
user identifier - идентификатор пользователя
user terminal - терминал пользователя
user-written driver - драйвер пользователя
utility program - программа обслуживания
utility routine program - обслуживающая программа

V

vertical communication - обмен информацией по вертикали
videodata - видеоданные
video informatics - видеоинформатика
videoierminal - видеотерминал
virtual database - виртуальная база данных

virtual datagram - виртуальная дейта-
грамма
virtual file - виртуальный файл
virtualization - виртуализация
virtual memory - виртуальная па-
мять
visual information - визуальная ин-
формация
voice input/output device - устройст-
во речевого ввода/ вывода дан-
ных

W

wide area network - глобальная
вычислительная сеть
winchester disk - винчестер
window - окно (в машинной графике)
word - слово
word length - длина слова
word size - длина слова
work(ing) area - рабочая область
work(ing) space - рабочая область
write record - запись данных
writing down - записывание

База данных (database) - автоматизиро-
ванное хранилище оперативно обновляе-
мых данных.

Гипертекст (hypertext) - сетевая форма
организации материала, разделенного на
фрагменты, для каждого из которых ука-
зан переход к другим фрагментам по
определенным типам связей.

**Глобальная сеть (global network, Wide Area
Network)** - международная сеть, обе-
спечивающая не только доступ к компьютер-
ным ресурсам, но и возможность взаимодей-
ствия между собой пользователей.

Драйвер (driver) - программа операцион-
ной системы, обслуживающая отдельные
периферийные устройства; в ЛВС - узел
устройства на передающем конце цепи,
осуществляющий вывод последователь-
ности двоичных цифровых сигналов.

Запись (record) - минимальная единица
обмена между программой и внешней
памятью.

**Звездообразная (радиальная) топология
сетей (star network)** - топология сети, при
которой все ее узлы соединены с одним
центральный узлом.

**Иерархическая топология сети (hierar-
chical network)** - многоуровневая сеть,
построенная на базе различных ЭВМ (на
одном уровне - однородных), объединяе-
мых в единое целое по принципу части и
целого.

**Интегральная сеть (integrated commu-
nication network)** - сеть, обеспечивающая
на одном и том же оборудовании комму-
тацию каналов, сообщений и пакетов.

Интерфейс (interface) - правила взаимо-
действия операционной системы с поль-
зователем, соседних уровней в сети ЭВМ.

ПРИЛОЖЕНИЕ 2

ГЛОССАРИЙ СЕТЕВЫХ ТЕХНОЛОГИЙ

Интрасеть (intranet) - внутренняя корпо-
ративная сеть, объединяющая несколько
ЛВС посредством протоколов TCP/IP и
HTTP.

**Информатизация общества (Society infor-
matization)** - совокупность взаимосвя-
занных политических, социально-эконо-
мических, научных факторов, которые
обеспечивают свободный доступ каждому
члену общества к любым источникам
информации, кроме законодательно сек-
ретных.

**Информационная технология (information
technology)** - совокупность методов, про-
изводственных и программно-техни-
ческих средств, объединенных в техноло-
гическую цепочку, обеспечивающую
сбор, хранение, обработку, вывод и рас-
пространение информации для снижения
трудоемкости процессов использования
информационных ресурсов, повышения
надежности и оперативности.

**Информационное обслуживание (informa-
tion service)** - предоставление информации
для выработки и принятия решений,
удовлетворения культурных, научных,
производственных, бытовых и других
потребностей человека.

**Информационное хранилище (information
store)** - автоматизированная система,
которая собирает данные из существую-
щих баз и внешних источников, формиру-
ет, хранит и эксплуатирует информацию
как единое целое.

Клиент (client) - приложение, посылающее
запрос серверу.

Клиент-сервер (client-server) - модель
взаимодействия пользователя с рабочими
станциями, при которой сервер сначала
обрабатывает запросы, а затем посылает
клиенту то, в чем он нуждается.

Ключ (key) - совокупность знаков, используемая для идентификации записи в файле и быстрого доступа к ней.

Кольцевая топология (token - ring network) - есть «эстафетное кольцо» («кольцевая ЛВС с маркером»), в которой для регулирования сетевого трафика и предотвращения конфликтов используется метод эстафетной передачи данных.

Коммуникационная сеть (communication network) - множество серверов и host-ЭВМ, объединенных физическими каналами связи, которые называются магистральными.

Коммутация каналов (channel switching) - способ передачи данных, осуществляемый прямым объединением каналов связи в соответствии с полученной адресной информацией.

Коммутация пакетов (packet switching) - способ передачи данных, при котором сообщения принимаются, накапливаются и передаются в виде пакетов данных.

Коммутация сообщений (message switching) - способ передачи данных, при котором прямое соединение между абонентами не устанавливается, а передача после накопления сообщений начинается после освобождения первого канала.

Компилятор (compiler) - транслятор, выполняющий преобразование программы, составленной на исходном языке, в объектный модуль.

Компьютерная система (computer system) - совокупность аппаратных и программных средств, разного рода физических носителей информации, а также персонала, обслуживающего перечисленные выше компоненты.

Компьютерный вирус (computer virus) - специальная программа, способная самопроизвольно присоединяться к другим (т.е. «заражать» их) и при запуске последних выполнять различные нежелательные действия: порчу файлов и каталогов, искажение результатов вычислений, зосо-

рение или стирание памяти, создание помех в работе ЭВМ.

Корпоративная сеть - то же, что Интрасеть.

Локальная вычислительная сеть (local area network) - сеть, узлы которой расположены на небольшом расстоянии друг от друга, не использующая средства связи общего назначения.

Маршрутизатор (router) - программа, определяющая путь передачи сообщения между узлами и абонентами сети.

Мост (bridge) - устройство для соединения двух сетей, присутствие которого для абонентов невидимо (в отличие от шлюза).

Мейнфрейм (mainframe) - большая ЭВМ с объемом оперативной памяти от 512 Мбайт до 8 Гбайт.

Мультимедиа (multimedia) - интерактивная система, обеспечивающая работу с неподвижными изображениями, движущимся видео, анимированной компьютерной графикой, текстом, речью и высококачественным звуком.

Мультимедиа-акселератор (multimedia - accelerator) - программно-аппаратные средства, которые объединяют базовые возможности графических акселераторов с одной или несколькими мультимедийными функциями, требующими обычно установки в компьютер дополнительных устройств.

Мультимедийные функции (multimedia functions) - цифровая фильтрация и масштабирование видео, аппаратная цифровая компрессия (сжатие) и декомпрессия (развертка) видео, ускорение графических операций, связанных с трехмерной графикой (3D), поддержка живого видео на мониторе, наличие композитного видеовыхода, вывод TV-сигнала (телевизионного) на монитор.

Общая шина (bus topology, common bus) - топология сети, при которой ее узлы подключены к общему линейному информационному каналу.

Операционная система (operation system) - программа, которая автоматически загружается при включении компьютера и предоставляет пользователю базовый набор команд, с помощью которых можно выполнять общение с компьютером и ряд действий.

Пакет (packet) - часть сообщения, удовлетворяющая некоторому стандарту, последовательность битов, содержащая данные и сигналы управления, передаваемые и коммутируемая как одно целое.

Пакетная технология (packet technology) - обработка данных или выполнение заданий, накопленных заранее, таким образом, что они объединяются в пакет и затем обрабатываются. При этом пользователь не может взять на обработку, пока она продолжается.

Пиктограмма (icon) - условное изображение информационного объекта или операции.

Платформа (platform) - тип процессора и ОС, на которых можно установить новый программный продукт.

Пользовательский интерфейс (user interface) - набор приемов взаимодействия пользователя с приложением.

Порт (port) - многоуровневый вход или выход в устройстве.

Почтовый ящик (mail box) - специально организованный файл для хранения корреспонденций.

Почтовый сервер (mail server) - ЭВМ в роли сетевого почтового отделения, которое получает, хранит и пересылает сообщения дальше по сети.

Приложение (application) - совокупность программ, реализующих обработку данных в определенной области применения.

Приоритет (priority) - число, присвоенное задаче, процессу или операции и определяющее очередность их выполнения или обслуживания.

Прозрачность передачи (transparency transmission) - свойство передачи информации, закодированной любым способом, понятным взаимодействующим уровням.

Протокол (protocol) - правила взаимодействия систем сети одного уровня.

Процесс (process) - функция обработки данных любого вида на компьютере.

Разделение времени (time sharing) - режим работы, при котором процессорное время предоставляется различным приложениям последовательно квантами. По истечении кванта времени приложение возвращается в очередь ожидания обслуживания.

Распределенная обработка данных (distributed data processing) - обработка данных, при которой поддержание базы в актуальном состоянии выполняется на одной ЭВМ, а содержательная обработка данных и обращение к базе - на другой ЭВМ.

Растр (raster) - множество точечных элементов, с помощью которых знак фиксируется на бумажном носителе или отображается на экране дисплея.

Реальное время (real time) - режим обработки данных, при котором обеспечивается взаимодействие вычислительной системы с внешними по отношению к ней процессами в темпе, сопоставимом со способностью протекания этих процессов.

Сервер (server) - любая ЭВМ, выполняющая определенные функции обслуживания пользователей. В сетях сервер управляет использованием распределенных ресурсов: баз данных, внешней памяти, принтеров и др.

Сервер базы данных (database server) - содержит базу данных, сетевую операционную систему, сетевую СУБД для обеспечения многопользовательских запросов.

Сетевая операционная система (network operational system) - система, осуществ-

ляющая реализацию протоколов с реализацией управления серверами.

Сетевой сервер (network server) - поддерживает выполнение функций сетевой операционной системы: управление сетью, планирование задач, распределение ресурсов, доступ к сетевой файловой системе, защита информации.

Сортировка (sorting) - распределение упорядоченных данных по возрастанию или убыванию значений признака сортировки; упорядочение записей файла по одному или нескольким ключам.

Тезаурус гипертекста (hypertext thesaurus) - автоматизированный словарь, отображающий семантические отношения между лексическими единицами дескрипторного информационно-поискового языка и предназначенный для поиска слов по их смысловому содержанию.

Технологический процесс (technological process) - упорядоченная последовательность взаимосвязанных операций по сбору, передаче, накоплению, хранению, обработке, анализу, отображению и размножению информации.

Топология (topology) - общая схема, схема соединений, конфигурация.

Транзакция (transaction) - входное сообщение, переводящее базу данных из одного непротиворечивого состояния в другое; запрос на изменение базы данных.

Трафик (traffic) - поток сообщений в сети передачи данных.

Уровень сети (network level) - совокупность станций одинакового ранга, входящих в одну и ту же иерархическую сеть.

Файл (file) - совокупность записей.

Файл-сервер (file-server) - содержит базу данных и программы управления данными для обеспечения многопользовательских запросов.

Шлюз (gateway) - устройство соединения сети с другими сетями, например, ЛВС с глобальной сетью.

Экспертная система (expert system) - система искусственного интеллекта, включающая базу знаний с набором правил и механизмом вывода, позволяющим на основании правил и предоставляемых пользователем фактов распознать ситуацию, поставить диагноз, сформулировать решение или дать рекомендацию для выбора действия.

Электронная почта (electronic mail) - система хранения и пересылки сообщений между пользователями сети ЭВМ.

Электронный офис (electronic office) - интегрированный ППП, включающий предметные программы и ИТ, обеспечивающие реализацию задач предметной области.

HTML (Hypertext Markup Language) - язык гипертекстовой развертки, используемый для создания документов, содержащих специальные команды форматирования и гипертекстовые ссылки и предназначенных для размещения в системе WWW.

HTTP (Hypertext Transfer Protocol) - протокол передачи гипертекста, служащий для установления связи с документами HTML независимо от его местонахождения с помощью указателей URL.

Host-ЭВМ - ЭВМ, установленная в узлах сети и решающая вопросы коммутации сети, доступа к сетевым ресурсам: модемам, факс-модемам, большим ЭВМ и др.; главная, ведущая, центральная ЭВМ.

Hub - концентратор, хаб. Устройство, обеспечивающее преобразование передаваемых сигналов таким образом, что сеть может быть дополнена новыми рабочими станциями.

OLE-технология (Object Linked and Embedding) - технология, позволяющая вставлять ранее созданные объекты (рисунки, графики, таблицы) во вновь создаваемые.

SILK (Speech Image Language Knowledge) - интерфейс, обеспечивающий перемещение на экране по речевой команде от одних поисковых образов к другим по смысловым (семантическим) связям.

SMTP (Simple Management Transfer Protocol) - упрощенный протокол передачи почтовых сообщений системами электронной почты.

SQL (Structured Query Language) - специализированный язык структурированных запросов, используемый при организации запросов, обновлении и управлении реляционными базами данных.

SQL-сервер - сервер.

TCP/IP (Transmission Control Protocol / Internet Protocol) - протокол управления передачей сообщений /межсетевой протокол, охватывающий средства доступа к среде, средства транспортировки пакетов, сеансовые коммуникационные средства, средства передачи файлов, средства электронной почты и средства терминальной эмуляции.

URL (Uniform Resource Locator) - универсальный локатор ресурсов, характеризующий полный путь к конкретному документу или разделу на компьютере, подключенном к сети Internet, а также метод доступа к нему, т.е. протокол работы с программами-серверами, функционирующими на удаленном компьютере.

UUCP (Unix-to-Unix-Copy-Program) - программное обеспечение, облегчающее передачу информации в сети между Unix-системами, использующими последовательные соединения, прежде всего обычную телефонную сеть.

WIMP (Windows Image Menu Pointer) - интерфейс, обеспечивающий размещение на экране окон, содержащих образы программ и меню действий.

Web page - Web-страница в системе WWW.

Web server - Web-сервер системы WWW.

WWW (World Wide Web) - глобальная сетевая система, объединяющая серверы Internet, «всемирная паутина».

ЛИТЕРАТУРА

1. *Алексеева И.В., Васяшин А.В., Острейковский В.А.* Информатика /Под ред. В.А. Острейковского. — Обнинск: Обнинский институт атомной энергетики, 1996.
2. *Вигдорчик Г.В., Воробьев А.Ю., Праченко В.Д.* Основы программирования на ассемблере для СМ ЭВМ. — М.: Финансы и статистика, 1987.
3. *Вирт Н.* Алгоритмы и структура данных. — М.: Мир, 1989.
4. *Власов В.К., Королев Л.Н., Сотников А.Н.* Элементы информатики /Под ред. Л.Н.Королева. — М.: Наука, Гл. ред. физ. мат. лит., 1988.
5. Вопросы прикладной информатики (сборник научных трудов) /Под ред. Р.М. Юсупова. — С.-Петербург: СПИИРАН, 1993.
6. *Гудман С., Хидетниemi С.* Введение в разработку и анализ алгоритмов. — М.: Мир, 1981.
7. Информатика. Энциклопедический словарь для начинающих. — М.: Педагогика-Пресс, 1994.
8. *Кричевский Р.Е.* Сжатие и поиск информации. — М.: Радио и связь, 1989.
9. *Лихачева Г.Н.* Информационные технологии в экономике. — М.: МЭСИ, 1998.
10. *Мейер Б., Бодуэн К.* Методы программирования. — М.: Мир, 1982.
11. *Нардюжев В.И.* Лабораторный практикум по курсу «Информатика и основы работы на персональном компьютере». — Обнинск: Обнинский институт атомной энергетики, 1996.
12. *Острейковский В.А.* Информатика. — М.: Высшая школа, 2000.
13. *Першков В.И., Савицков В.М.* Толковый словарь по информатике. — М.: Финансы и статистика, 1991.
14. Словарь по кибернетике / Под ред. В.С. Михалевича. — 2-е изд. — Киев.: Гл. ред. УСЭ им. Бажана, 1989.
15. Экономическая информатика и вычислительная техника /Под ред. В.П. Косырева, А.Ю. Королева. — М.: Финансы и статистика, 1996.
16. Электронные вычислительные машины. В 8 кн. Кн.2. Основы информатики / Под ред. А.Я.Савельева. — М.: Высшая школа, 1991.

ОГЛАВЛЕНИЕ

СПИСОК ОСНОВНЫХ СОКРАЩЕНИЙ.....	3
ПРЕДИСЛОВИЕ.....	5
ВВЕДЕНИЕ.....	9
РАЗДЕЛ I. ТЕОРЕТИЧЕСКАЯ ИНФОРМАТИКА.....	12
Глава 1. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ ИНФОРМАТИКИ.....	12
Методические указания	12
1.1. Терминология информатики	13
1.2. Объект и предметная область информатики	16
1.3. Виды и свойства информации.....	22
1.4. Информатизация общества	24
1.5. Краткая история развития информатики	31
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	33
Глава 2. МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ	34
Методические указания	34
2.1. Методы и модели оценки количества информации	35
2.2. Системы счисления	37
2.2.1. Позиционные системы счисления	38
2.2.2. Двоичная система счисления	40
2.2.3. Другие позиционные системы счисления	41
2.2.4. Смешанные системы счисления	42
2.2.5. Перевод чисел из одной системы счисления в другую.....	43
2.3. Формы представления и преобразования информации.....	43
2.3.1. Числовая система ЭВМ. Представление целых чисел без знака и со знаком	45
2.3.2. Индикаторы переноса и переполнения	46
2.3.3. Представление символьной информации в ЭВМ.....	47
2.3.4. Форматы данных.....	49
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	53
Глава 3. АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ.....	54
Методические указания	54
3.1. Определение алгоритма	55
3.2. Методы разработки алгоритма	61
3.2.1. Метод частных целей.....	61
3.2.2. Метод подъема	64
3.3. Программирование с отходом назад	65

3.4. Алгоритмы ветвей и границ	70
3.5. Языки программирования	77
3.6. Жизненный цикл программного обеспечения	85
КОНТРОЛЬНЫЕ ВОПРОСЫ И УПРАЖНЕНИЯ	88
РАЗДЕЛ II. ПРИКЛАДНАЯ ИНФОРМАТИКА	90
Глава 4. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОЦЕССОВ СБОРА, ПЕРЕДАЧИ, ОБРАБОТКИ И ХРАНЕНИЯ ИНФОРМАЦИИ	90
Методические указания	91
4.1. Восприятие информации	91
4.2. Сбор информации	93
4.3. Передача информации	100
4.4. Обработка информации	103
4.5. Хранение информации	111
КОНТРОЛЬНЫЕ ВОПРОСЫ	112
Глава 5. ТЕХНИЧЕСКИЕ СРЕДСТВА РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ	113
Методические указания	113
5.1. Определение и принципы организации информационных процессов в вычислительных устройствах	114
5.2. Функционирование ЭВМ с шинной организацией	116
5.3. Функционирование ЭВМ с канальной организацией	122
5.4. Информационная модель ЭВМ	125
5.5. Основные команды ЭВМ	128
5.6. Поколения вычислительных средств	131
КОНТРОЛЬНЫЕ ВОПРОСЫ	136
Глава 6. ПЕРСОНАЛЬНЫЕ ЭВМ	137
Методические указания	137
6.1. Краткая характеристика и классификация ПК	138
6.2. Характеристика основных блоков ПК	141
6.3. Внешние запоминающие устройства ПК	145
6.4. Внешние устройства ПК	147
6.5. Печатающие устройства ПЭВМ	151
6.6. Операционная система и интерфейс пользователя ПК	153
6.7. Технология информационных процессов в ПК	156
6.8. Работа с текстом, графикой и таблицами	159
6.9. Перспективы развития ПК	162
КОНТРОЛЬНЫЕ ВОПРОСЫ	165
РАЗДЕЛ III. ЭЛЕМЕНТЫ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	166
Глава 7. КЛАССИФИКАЦИЯ И ХАРАКТЕРИСТИКА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	166
Методические указания	167
7.1. Классификация и характеристика современных информационных технологий	168
7.2. Автоматизированные банки данных	171
7.2.1. Модели данных	176
7.2.2. Организация поиска данных	179

7.2.3. Администратор базы данных	181
7.3. Пакеты прикладных программ	182
7.3.1. Классификация ППП	182
7.3.2. Проблемно-ориентированные ППП	184
7.3.3. Интегрированные ППП	189
7.4. Автоматизированные рабочие места	191
7.5. Искусственный интеллект	197
7.5.1. Направление исследований в области искусственного интеллекта	198
7.5.2. Машинный интеллект и робототехника	199
7.5.3. Интеллектуальные роботы	203
7.5.4. Моделирование биологических систем	205
7.5.5. Эвристическое программирование и моделирование	207
7.5.6. Система знаний	210
7.5.7. Модели представления знаний	211
7.6. Экспертные системы	213
7.6.1. Классификация инструментальных средств ЭС	217
7.6.2. Организация знаний в ЭС	219
7.6.3. Виды ЭС и типы решаемых ими задач	223
7.7. Геоинформационные системы и технологии	229
КОНТРОЛЬНЫЕ ВОПРОСЫ	234

Глава 8. ВЫЧИСЛИТЕЛЬНЫЕ И ИНФОРМАЦИОННЫЕ СЕТИ	236
Методические указания	236
8.1. Принципы построения и классификация сетей	237
8.2. Способы коммутации и передачи данных в ВС	240
8.3. Локальные вычислительные сети	246
8.4. Обеспечение защиты информации в сетях	252
КОНТРОЛЬНЫЕ ВОПРОСЫ	258

Глава 9. ГЛОБАЛЬНЫЕ ИНФОРМАЦИОННЫЕ СЕТИ. ИНТЕРНЕТ	260
Методические указания	261
9.1. Характеристика глобальных сетей	261
9.2. Иерархия протоколов Интернет	271
9.3. Электронная почта	274
9.4. Гипертекст	279
9.5. Мультимедиа	283
9.6. Технология WORLD WIDE WEB (WWW)	286
КОНТРОЛЬНЫЕ ВОПРОСЫ	289

ЗАКЛЮЧЕНИЕ	290
ПРИЛОЖЕНИЯ	293
ПРИЛОЖЕНИЕ 1. КРАТКИЙ АНГЛО-РУССКИЙ СЛОВАРЬ	293
ПРИЛОЖЕНИЕ 2. ГЛОССАРИЙ СЕТЕВЫХ ТЕХНОЛОГИЙ	311
ЛИТЕРАТУРА	316