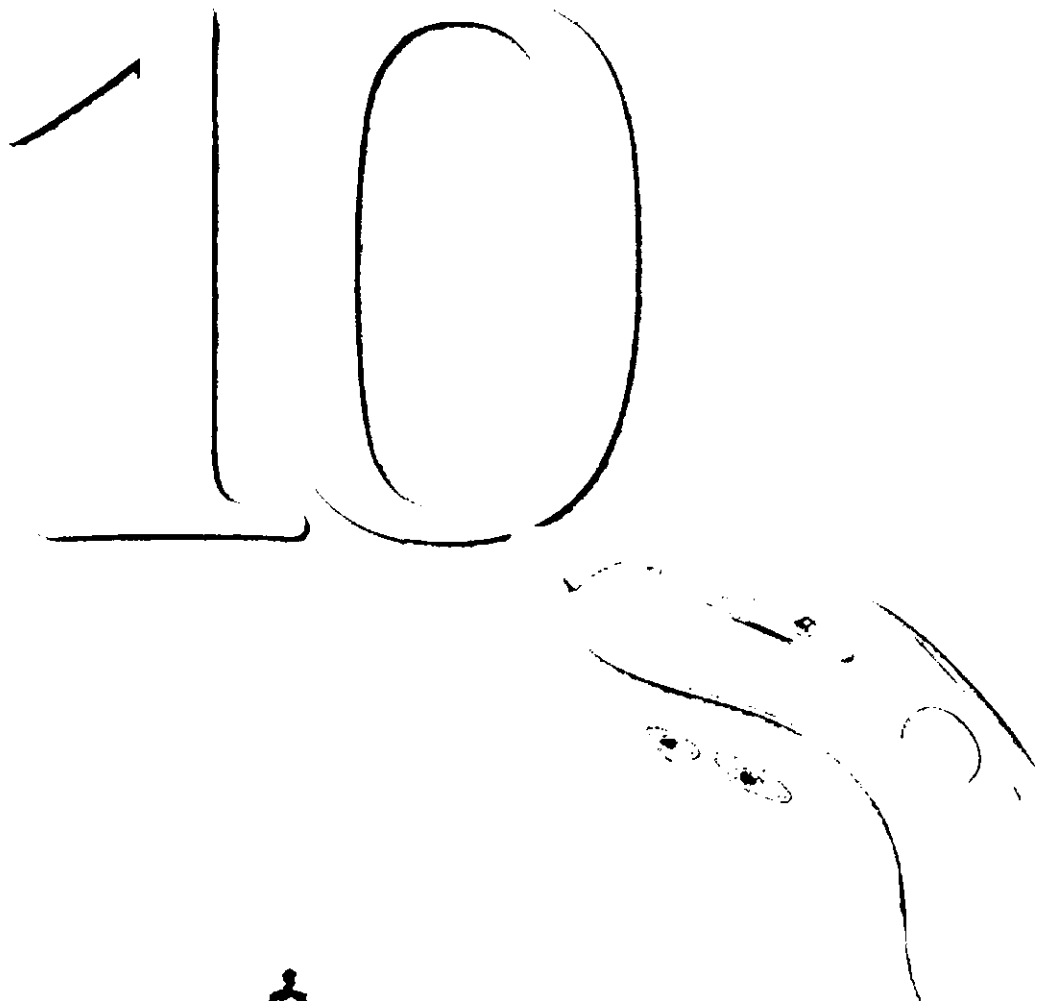


Н. Д. Угринович



ИНФОРМАТИКА

и ИКТ



БИНОМ



Н. Д. Угринович

ИНФОРМАТИКА и ИКТ

Учебник для 10 класса

3-е издание, исправленное

Допущено
Министерством образования и науки Российской Федерации
к использованию в образовательном процессе
в образовательных учреждениях,
реализующих образовательные программы
общего образования



Москва
БИНОМ. Лаборатория знаний
2008

УДК 004.9
ББК 32.97
У27

Угринович Н. Д.
У27 Информатика и ИКТ. Профильный уровень : учебник для 10 класса / Н. Д. Угринович. — 3-е изд., испр. — М. : БИНОМ. Лаборатория знаний, 2008. — 387 с. : ил.

ISBN 978-5-94774-828-4

Учебник по курсу «Информатика и ИКТ» ориентирован на информационно-технологический и физико-математический профили обучения в общеобразовательных учреждениях. Учебник полностью соответствует новому образовательному стандарту и примерной программе профильного обучения, утвержденным Министерством образования и науки РФ. Содержание учебника соответствует программе вступительных экзаменов по информатике в вузы, и он может быть использован для подготовки к экзаменам.

Рассматриваются архитектура компьютера и методы защиты информации, понятие «информация» и системы счисления, основы логики и логические основы компьютера, а также объектно-ориентированное программирование на четырех языках: Visual Basic, Delphi, Visual C# и Visual J#. Все необходимое для преподавания курса программное обеспечение содержится на CD-дисках, которые входят в состав методического пособия для учителя.

УДК 004.9
ББК 32.97

По вопросам приобретения обращаться:

«БИНОМ. Лаборатория знаний»
(499) 157-52-72, e-mail: Lbz@aha.ru
<http://www.Lbz.ru>

ISBN 978-5-94774-828-4 © Угринович Н. Д., 2008
© БИНОМ. Лаборатория знаний, 2008

Оглавление

Рекомендации по использованию учебника	7
Глава 1. Архитектура компьютера и защита информации	9
1.1. Магистрально-модульный принцип построения компьютера	10
1.2. Процессор и оперативная память	16
1.2.1. Процессор	16
1.2.2. Оперативная память	23
1.3. Внешняя (долговременная) память	29
1.3.1. Магнитная память	29
1.3.2. Оптическая память	31
1.3.3. Флэш-память	34
1.4. Файл и файловые системы	37
1.4.1. Логическая структура носителя информации	37
1.4.2. Файл	44
1.4.3. Иерархическая файловая система	49
1.5. Операционная система	54
1.5.1. Назначение и состав операционной системы	54
1.5.2. Загрузка операционной системы	59
1.6. Защита информации от вредоносных программ	61
1.6.1. Вредоносные программы и антивирусные программы	61
1.6.2. Компьютерные вирусы и защита от них	64
1.6.3. Сетевые черви и защита от них	71
1.6.4. Троянские программы и защита от них	75
1.6.5. Рекламные и шпионские программы и защита от них	79
1.6.6. Спам и защита от него	84
1.6.7. Хакерские утилиты и защита от них	88
Глава 2. Информация. Системы счисления	94
2.1. Понятие «информация» в науках о неживой и живой природе, обществе и технике	94
2.1.1. Информация в физике	94
2.1.2. Информация в биологии	99
2.1.3. Информация в общественных науках	101
2.1.4. Информация в кибернетике	103


2.2. Количество информации как мера уменьшения неопределенности знания	105
2.3. Алфавитный подход к определению количества информации	110
2.4. Формула Шеннона	111
2.5. Кодирование текстовой, графической и звуковой информации	115
2.6. Хранение информации	119
2.7. Кодирование числовой информации. Системы счисления	121
2.7.1. Непозиционные системы счисления	121
2.7.2. Позиционные системы счисления	125
2.8. Перевод чисел из одной системы счисления в другую	129
2.8.1. Перевод целых чисел из десятичной системы в двоичную, восьмеричную и шестнадцатеричную	129
2.8.2. Перевод дробей из десятичной системы в двоичную, восьмеричную и шестнадцатеричную	133
2.8.3. Перевод чисел из двоичной системы в восьмеричную и шестнадцатеричную и обратно	136
2.9. Арифметические операции в позиционных системах счисления	140
2.10. Представление чисел в компьютере	143
2.10.1. Представление чисел в формате с фиксированной запятой	143
2.10.2. Представление чисел в формате с плавающей запятой	148
Глава. 3. Основы логики и логические основы компьютера	151
3.1. Формы мышления	151
3.2. Алгебра логики	156
3.2.1. Логическое умножение, сложение и отрицание	156
3.2.2. Логические выражения	164
3.2.3. Логические функции	169
3.2.4. Логические законы и правила преобразования логических выражений	174
3.2.5. Решение логических задач	177
3.3. Логические основы устройства компьютера	180
3.3.1. Базовые логические элементы	180
3.3.2. Сумматор двоичных чисел	183
3.3.3. Триггер	188

Алгоритмизация и основы объектно-ориентированного программирования	190
4.1. Алгоритм и кодирование основных алгоритмических структур	190
4.1.1. Алгоритм и его свойства.	190
4.1.2. Алгоритмические структуры «ветвление» и «выбор»	192
4.1.3. Алгоритмическая структура «цикл»	195
4.2. История развития языков программирования.	197
4.3. Введение в объектно-ориентированное визуальное программирование	201
4.3.1. Объекты: свойства и методы.	201
4.3.2. События	203
4.3.3. Проекты и приложения	205
4.4. Система объектно-ориентированного программирования Microsoft Visual Studio .NET	208
4.4.1. Платформа .NET Framework	208
4.4.2. Интегрированная среда разработки языков Visual Basic .NET, Visual C# и Visual J#	212
4.5. Система объектно-ориентированного программирования Turbo Delphi	216
4.6. Переменные	221
4.7. Графический интерфейс	226
4.8. Пространство имен .NET.	237
4.9. Процедуры и функции	243
4.9.1. Процедуры	243
4.9.2. Функции	248
4.10. Итерация и рекурсия	252
4.11. Делегаты.	257
4.12. Алгоритмы перевода чисел и их кодирование на языках объектно-ориентированного программирования	262
4.12.1. Алгоритм перевода целых чисел.	262
4.12.2. Алгоритм перевода дробных чисел	272
4.13. Графика в объектно-ориентированных языках программирования	281
4.13.1. Графика в языках программирования Visual Basic .NET, Visual C# и Visual J#	281
4.13.2. Графика в языке программирования Turbo Delphi	300

4.13.3. Компьютерная и математическая системы координат	311
4.13.4. Анимация.	316
4.14. Модульный принцип построения решений (групп) и проектов.	320
4.15. Чтение и запись данных в файлы	329
4.16. Массивы	332
4.16.1. Заполнение массивов	332
4.16.2. Поиск элемента в массивах	341
4.16.3. Сортировка числовых массивов	344
4.16.4. Сортировка строковых массивов.	354
Глава. 5. Практические задания для тематического и итогового контроля	359
Тема «Архитектура компьютера и защита информации»	359
Тема «Информация. Системы счисления»	367
Тема «Логика и логические основы компьютера»	368
Тема «Алгоритмизация и основы объектно-ориентированного программирования»	369
Ответы и решения	375
Плакаты	377
Таблица 1. Процессоры	377
Таблица 2. Элементы управления для ввода и вывода данных	379
Таблица 3. Элементы управления для организации интерактивного диалога	380
Таблица 4. Элементы управления, не отображаемые на форме в процессе выполнения проекта.	381
Таблица 5. Типы переменных	382
Таблица 6. Встроенные функции (методы).	383
Таблица 7. Кодирование алгоритмических структур «ветвление» и «выбор»	384
Таблица 8. Кодирование алгоритмической структуры «цикл»	385
Словарь компьютерных терминов	386


Рекомендации по использованию учебника

1. Учебник «Информатика и ИКТ-10. Профильный уровень» обеспечивает изучение профильного курса «Информатика и ИКТ» в 10 классе на профильном уровне в соответствии с образовательным стандартом. Учебник входит в состав учебно-программного комплекса, включающего:
 - учебники для основной школы: «Информатика и ИКТ-8» и «Информатика и ИКТ-9»;
 - учебники по профильному курсу для старшей школы на базовом уровне: «Информатика и ИКТ-10. Базовый уровень» и «Информатика и ИКТ-11. Базовый уровень»;
 - учебники по профильному курсу для старшей школы на профильном уровне: «Информатика и ИКТ-10. Профильный уровень» и «Информатика и ИКТ-11. Профильный уровень»;
 - учебное пособие и CD-ROM по элективному курсу для старшей школы «Исследование информационных моделей»;
 - методическое пособие для учителей «Преподавание курса «Информатика и ИКТ» в основной и старшей школе»;
 - Windows-CD, содержащий программную поддержку курса, готовые компьютерные проекты, рассмотренные в учебниках, тесты и методические материалы для учителей;
 - Visual Studio-CD (выпускается по лицензии корпорации Microsoft), содержащий дистрибутивы систем объектно-ориентированного программирования языков *Visual Basic .NET*, *Visual C#* и *Visual J#*;
 - Linux-DVD (выпускается по лицензии компании Alt-Linux), содержащий операционную систему Linux и программную поддержку курса;
 - TurboDelphi-CD (выпускается по лицензии компании Borland), содержащий систему объектно-ориентированного программирования TurboDelphi.
2. В практических работах указано необходимое для их выполнения программное обеспечение, которое обозначается значком приложения.
3. В учебнике используются ссылки на внешние источники информации (учебники, CD-диски и Интернет), а также на параграфы и пункты самого учебника.
4. В тексте пособия приняты следующие выделения:
 - Шрифтом Arial выделены имена программ, файлов и Интернет-адреса.

- Шрифтом Courier New выделены программы на языках программирования.
 - *Курсивом* выделены названия диалоговых окон, пунктов меню и элементов управления (текстовых полей, кнопок и т. д.) графического интерфейса операционных систем и приложений.
 - **Полужирным шрифтом** выделены важные термины и понятия.
 - Подчеркиванием выделены термины, краткое объяснение которых содержится в Словаре компьютерных терминов.
5. Важная информация выделена в тексте восклицательным знаком, а формулы — цифровым обозначением.
 6. Абзацы, содержащие дополнительную интересную информацию, выделены значком .
 7. Проекты создаются в системах объектно-ориентированного программирования Visual Basic .Net, Visual C#, Visual J# и Turbo Delphi и хранятся на CD-дисках Windows-CD и TurboDelphi-CD.
 8. Дополнительные материалы и интерактивные тесты для проверки усвоения материала находятся в Интернете по адресу: <http://iit.metodist.ru>

Глава 1

Архитектура компьютера и защита информации

Windows-CD 

В процессе изучения данной темы рекомендуется установить программное обеспечение в операционной системе Windows:

- программы тестирования компьютера SiSoft Sandra, CPU-Z, EasyTune, SIV;



- файловый менеджер Total Commander;



- архиватор 7-Zip;



- программу записи CD- и DVD-дисков: DeepBurner;



- браузеры Internet Explorer и SeaMonkey;



- антивирусные программы: avast! и Antivir Personal Edition;



- программу удаления рекламных и шпионских программ: Ad-Aware;



- программу восстановления системы: CCleaner;



- межсетевой экран: Outpost Firewall.



1.1. Магистрально-модульный принцип построения компьютера

В основу архитектуры современных персональных компьютеров положен магистрально-модульный принцип. Этот принцип предусматривает построение компьютера из функциональных блоков, взаимодействующих посредством общего канала (каналов) — шины. В сочетании с открытой (общезвестной) архитектурой это позволяет потребителю собирать машину нужной конфигурации.

Магистраль включает в себя три многопроводные шины: шину данных, шину адреса и шину управления, которые представляют собой многопроводные линии (рис. 1.1). К магистрали подключаются процессор и оперативная память, а также периферийные устройства ввода, вывода и хранения информации, которые обмениваются информацией в форме последовательностей нулей и единиц, реализованных в виде электрических импульсов.

Шина данных. По этой шине данные передаются между различными устройствами. Например, считанные из оперативной памяти данные могут быть переданы процессору для обработки, а затем полученные данные могут быть отправлены обратно в оперативную память для хранения. Таким образом, данные по шине данных могут передаваться от устройства к устройству через области оперативной памяти.

Разрядность шины данных определяется разрядностью процессора, т. е. количеством двоичных разрядов, которые процессор обрабатывает за один такт. Разрядность процессоров постоянно увеличивалась по мере развития компьютерной техники и в настоящее время составляет 64 бита.

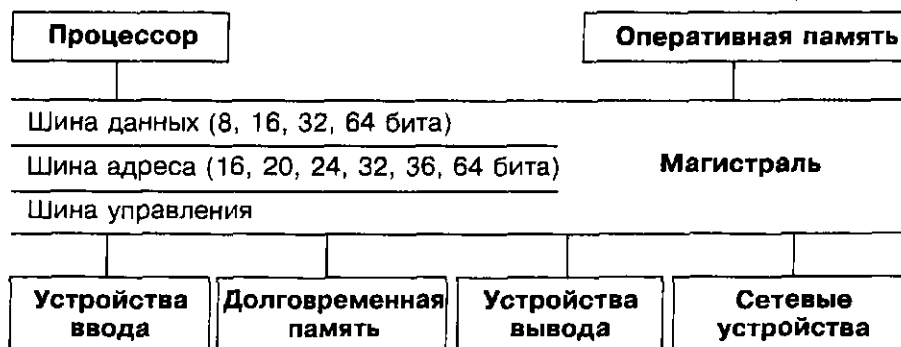


Рис. 1.1. Магистрально-модульное устройство компьютера

Шина адреса. Выбор устройства или ячейки памяти, куда пересылаются или откуда считываются данные по шине данных, производит процессор. Каждое устройство или ячейка оперативной памяти имеет свой адрес. Адрес передается по адресной шине, причем сигналы по ней передаются в одном направлении от процессора к оперативной памяти и устройствам (однонаправленная шина).

Разрядность шины адреса определяет объем адресуемой памяти, т. е. количество ячеек оперативной памяти, которые могут иметь уникальные адреса. Количество адресуемых ячеек памяти можно рассчитать по формуле:

$$N = 2^I, \text{ где } I \text{ — разрядность шины адреса.}$$

Разрядность шины адреса постоянно увеличивалась и в процессорах Pentium Extreme Edition составляет 64 бита. Таким образом, количество адресуемых ячеек памяти в таких процессорах равно:

$$N = 2^{64} \text{ ячеек.}$$

Шина управления. По шине управления передаются сигналы, определяющие характер обмена информацией по магистрали. Сигналы управления определяют, какую операцию — считывание или запись информации из памяти нужно производить, синхронизируют обмен информацией между устройствами и т. д.

Системная плата. Важнейшим аппаратным компонентом компьютера является системная плата (рис. 1.2, 1.3). На системной плате реализована магистраль обмена информацией, имеются разъемы для установки процессора, слоты для уста-

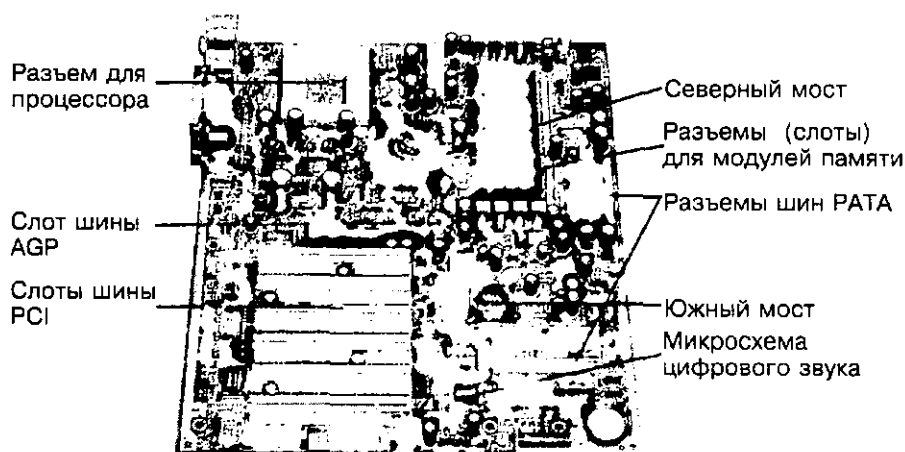


Рис. 1.2. Системная плата



Рис. 1.3. Логическая схема системной платы

новки оперативной памяти, а также контроллеров внешних устройств.

Пропускная способность. Быстродействие устройства зависит от тактовой частоты тактового генератора (обычно измеряется в мегагерцах — МГц) и разрядности, т. е. количества битов данных, которые устройство может обрабатывать или передавать одновременно (измеряется в битах). Дополнительно в устройствах используется внутреннее умножение частоты с разными коэффициентами.

Соответственно, скорость передачи данных (пропускная способность) соединяющих эти устройства шин также должна различаться. Пропускная способность шины данных (измеряется в бит/с) равна произведению разрядности шины (измеряется в битах) и частоты шины (измеряется в Гц = 1/с):

$$\begin{aligned} \text{Пропускная способность шины} &= \\ &= \text{Разрядность шины} \times \text{Частота шины}. \end{aligned}$$

Северный и южный мосты. Для согласования тактовой частоты и разрядности устройств на системной плате устанавливаются специальные микросхемы (их набор называется чипсетом), включающие в себя контроллер оперативной памяти и видеопамяти (так называемый северный мост) и контроллер периферийных устройств (южный мост).

Частота процессора. Северный мост обеспечивает обмен данными с процессором, оперативной памятью и видеопамятью. Частота процессора в несколько раз больше, чем базовая частота магистрали (иногда ее называют шиной FSB от англ. FrontSide Bus). Например, в наиболее быстрых компьютерах (2006 год) частота шины FSB составляет 266 МГц, коэффициент умножения частоты 14, следовательно, частота процессора $266 \text{ МГц} \times 14 \approx 3,7 \text{ ГГц}$.

Системная шина. Между северным мостом и процессором данные передаются по системной шине с частотой, которая в четыре раза больше частоты шины FSB. Таким образом, процессор может получать и передавать данные с частотой $266 \text{ МГц} \times 4 = 1064 \text{ МГц}$. Так как разрядность системной шины равна разрядности процессора и составляет 64 бита, то пропускная способность системной шины равна:

$$64 \text{ бит} \times 1064 \text{ МГц} = 68\,096 \text{ Мбит/с} \approx \\ \approx 66 \text{ Гбит/с} \approx 8 \text{ Гбайт/с.}$$

Шина памяти. Обмен данными между процессором и оперативной памятью производится по шине памяти, частота которой может быть меньше, чем частота шины процессора. Например, частота шины памяти может составлять 533 МГц, т. е. оперативная память получает данные в два раза реже, чем процессор. Так как разрядность шины памяти равна разрядности процессора и составляет 64 бита, то пропускная способность шины памяти равна:

$$64 \text{ бит} \times 533 \text{ МГц} = 34\,112 \text{ Мбит/с} \approx \\ \approx 33 \text{ Гбит/с} \approx 4 \text{ Гбайт/с.}$$

Шины AGP и PCI Express. По мере усложнения графики приложений требования к быстродействию шины, связывающей видеопамять с процессором и оперативной памятью, возрастают. Для подключения видеоплаты к северному мосту может использоваться 32-битовая шина AGP (Accelerated Graphic Port — ускоренный графический порт). Эта шина первоначально передавала данные с частотой 66 МГц, в настоящее время возможно использование шины AGP×8, частота которой $66 \text{ МГц} \times 8 = 528 \text{ МГц}$. В этом случае пропускная способность шины видеоданных составляет:

$$32 \text{ бит} \times 528 \text{ МГц} = 16\,896 \text{ Мбит/с} = \\ = 16,5 \text{ Гбит/с} \approx 2 \text{ Гбайт/с.}$$

В настоящее время для подключения видеоплаты к северному мосту все большее распространение получает шина PCI

Express (Peripheral Component Interconnect bus Express — ускоренная шина взаимодействия периферийных устройств). Пропускная способность этой шины значительно выше пропускной способности PCI и AGP.

К видеоплате с помощью аналогового разъема VGA (Video Graphics Array — графический видеоадаптер) или цифрового разъема DVI (Digital Visual Interface — цифровой видеоинтерфейс) подключается электронно-лучевой или жидко-кристаллический монитор или проектор.

Шина PCI. К северному мосту подключается по специальной шине южный мост, к которому, в свою очередь, подключаются периферийные устройства. Шина PCI (Peripheral Component Interconnect bus — шина взаимодействия периферийных устройств) обеспечивает обмен информацией с контроллерами периферийных устройств, которые устанавливаются в слоты расширения системной платы.

Наиболее часто эта шина используется для установки устройств доступа к локальной сети (сетевая карта), глобальной сети Интернет (встроенный модем) и беспроводной сети (сетевой адаптер Wi-Fi, произносится «вай-фай», сокр. от Wireless Fidelity — протокол и стандарт на оборудование для широкополосной радиосвязи).

Разрядность шины PCI может составлять 32 бита или 64 бита, а частота — 33 МГц или 66 МГц. Таким образом, максимальная пропускная способность шины PCI составляет:

$$64 \text{ бит} \times 66 \text{ МГц} = 4224 \text{ Мбит/с} = 528 \text{ Мбайт/с.}$$

Шина IEEE 1394 (другие названия FireWire, i-Link). Последовательная высокоскоростная шина, предназначенная для обмена цифровой информацией между компьютером и цифровыми устройствами (цифровыми видеокамерами, DVD-плеерами и др.) без потери качества изображения и звука. (Эту функцию может выполнять также контроллер IEEE 1394, который подключается к шине PCI.) Скорость передачи данных по этой шине может достигать 200 Мбайт/с и более.

Шина ATA. Устройства внешней памяти (жесткие диски, CD- и DVD-дисководы) подключаются к южному мосту по шине ATA (англ. Advanced Technology Attachment — шина подключения накопителей). Ранее использовалась параллельная шина PATA (англ. Parallel ATA), скорость передачи данных по которой может достигать 133 Мбайт/с. В настоящее время широкое распространение получила последовательная шина SATA (англ. Serial ATA), скорость передачи данных по которой может достигать 300 Мбайт/с.

Шина USB. Для подключения принтеров, сканеров, цифровых камер и других периферийных устройств обычно используется шина USB (Universal Serial Bus — универсальная последовательная шина). Эта шина обладает пропускной способностью до 60 Мбайт/с и обеспечивает подключение к компьютеру одновременно нескольких периферийных устройств (принтер, сканер, цифровая камера, Web-камера, модем и др.).

Клавиатура и мышь. Клавиатура и мышь подключаются с помощью порта PS/2 или шины USB (в том числе с помощью беспроводного адаптера).

Звук. К южному мосту может подключаться интегрированная в системную плату микросхема, которая обеспечивает обработку цифрового звука (эту функцию может выполнять также звуковая плата, которая подключается к шине PCI). С помощью аудиоразъемов к системной плате могут подключаться микрофон, колонки или наушники.

Практическое задание «Тестирование системной платы». С помощью программы тестирования EasyTune определить у вашего компьютера частоту шины FSB, коэффициент умножения частоты процессора, частоту процессора, частоту шины памяти, частоту шин AGP и PCI Express и частоту шины PCI. Вычислить пропускную способность шины памяти.



Тестирование системной платы

1. Запустить программу тестирования EasyTune5 и выбрать вкладку *Advanced Mode*.

На панели отобразятся параметры компьютера, в данном случае (рис. 1.4):

- частота шины FSB — 133 МГц;
- коэффициент умножения частоты процессора — 20;
- частота процессора — 2680 МГц;
- частота шины памяти — 266 МГц;
- частота шин AGP и PCI Express — 66 МГц;
- частота шины PCI — 33 МГц.

2. Пропускная способность шины памяти = 64 бита × × 266 МГц = 17024 Мбит/с = 2128 Мбайт/с ≈ 2 Гбайт/с.

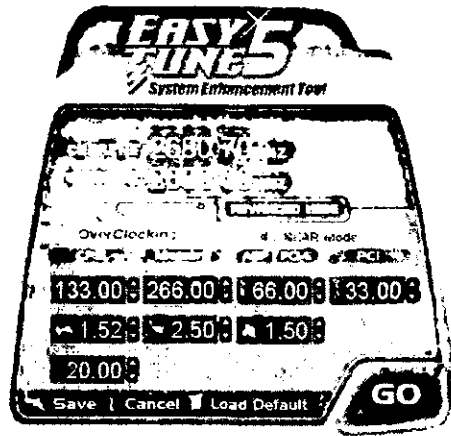


Рис. 1.4. Тестирование системной платы

Контрольные вопросы

1. Что определяют разрядности шины данных и шины адреса?
2. Почему различаются скорости передачи данных по шине FSB, системной шине, шине памяти, шинам AGP и PCI Express и шине PCI?

Компьютерный практикум

Windows-CD



- 1.1. С помощью программы тестирования компьютера EasyTune выполнить практическое задание «Тестирование системной платы».



1.2. Процессор и оперативная память

1.2.1. Процессор

Логическая схема процессора (рис. 1.5). Процессор является центральным устройством компьютера и выполняет команды программы, которая хранится в оперативной памяти. Команда программы поступает в процессор по шине данных и декодируется, т. е. определяется, какие действия необходимо выполнить и какие данные для этого требуются.

Данные запрашиваются из оперативной памяти, для этого по шине адреса передаются их адреса, а по шине управления — сигнал на считывание. Считанные данные передаются в процессор по шине данных.

Декодированная команда и данные передаются в АЛУ (арифметико-логическое устройство), где отдельно обрабатываются целочисленные данные, и отдельно — данные в форме чисел с плавающей запятой.

2.10. Представление чисел в компьютере

Результаты обработки передаются по шине данных в оперативную память, одновременно по шине адреса передаются адреса ячеек памяти, куда данные необходимо записать, а по шине управления передается сигнал на запись.

Быстродействие процессора существенно больше быстродействия оперативной памяти, поэтому процессор часть времени простаивает в ожидании данных. Чтобы этого не происходило, в современные процессоры встроена более быстрая, чем оперативная, кэш-память.

Кэш-память разделена на два уровня. В кэш-память 2-го уровня (большую по объему и менее быстродействующую) считывается из оперативной памяти очередная порция команд и данных. Кэш-память 1-го уровня (меньшая по объему, но более быстродействующая) разделена на две части, в одну часть считываются наиболее нужные процессору данные, а в другую часть — наиболее нужные процессору команды.

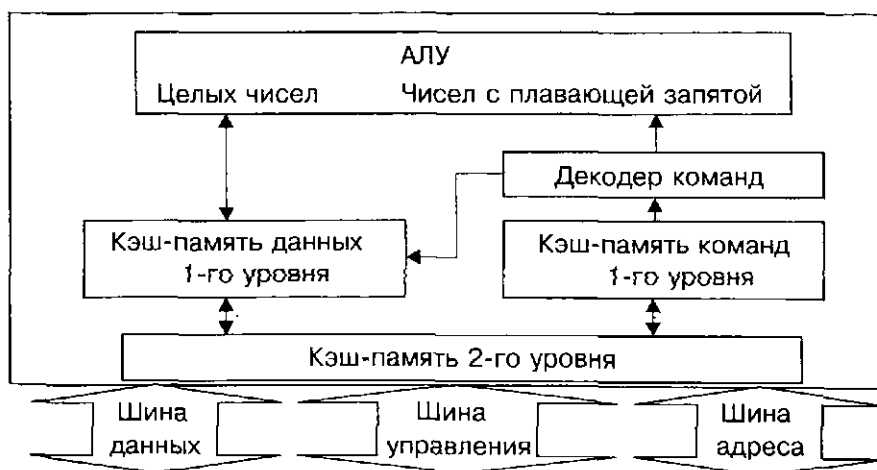


Рис. 1.5. Упрощенная логическая схема одноядерного процессора

Практическое задание «Определение объемов кэш-памяти процессора». С помощью программы тестирования CPU-Z определить объемы кэш-памяти 1-го и 2-го уровней вашего процессора.



Определение объемов кэш-памяти процессора

1. Запустить программу тестирования CPU-Z и выбрать вкладку *Cache* (рис. 1.6).

На панели отобразятся параметры кэш-памяти, в данном случае:

- кэш-память данных 1-го уровня — 8 Кбайт;
- кэш-память команд 1-го уровня — 12 Кбайт;
- кэш-память 2-го уровня — 128 Кбайт.

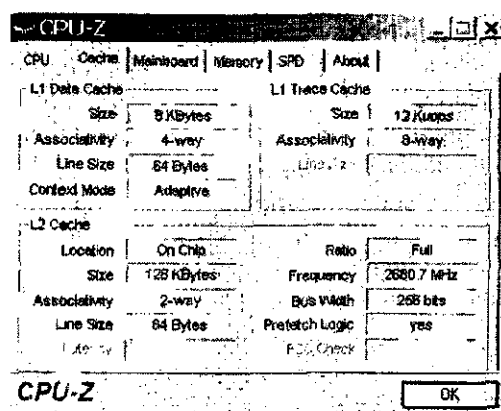


Рис. 1.6. Определение объемов кэш-памяти процессора

Технология. Процессор является большой (в смысле количества размещенных на ней элементов) полупроводниковой микросхемой. Процессоры создаются на основе кремниевых подложек, которые вырезаются из слитка чистого кремния. Кремний является полупроводником, который в разных условиях может вести себя и как проводник электрического тока, и как изолятор.

Физика-11

Сначала на кремниевой подложке под воздействием высокой температуры и кислорода формируется слой диоксида кремния. Этот процесс очень похож на возникновение ржавчины на железе, погруженном в воду. Разница заключается в том, что слой диоксида кремния формируется на подложке

гораздо быстрее и не виден невооруженным глазом (из-за того, что очень тонок).

Химия-9

Затем кремниевая подложка покрывается фотослоем. В процессе **фотолитографии** ультрафиолетовое излучение, проходя сквозь маску (которая выполняет функцию шаблона), формирует на подложке рисунок электрической схемы. Засвеченные участки фотослоя и диоксида кремния полностью удаляются с помощью растворителя. С помощью процесса ионной имплантации области кремниевой подложки, обработанные ультрафиолетом, бомбардируются ионами различных примесей. Ионы проникают в подложку, обеспечивая необходимую электрическую проводимость этих областей. В результате на кремниевой подложке создается электрическая схема.

После этого наносится слой поликристаллического кремния и еще один фотослой. Наложение новых слоев с последующим вытравливанием схемы осуществляется несколько раз, при этом для межслойных соединений в слоях оставляются «окна». Эти «окна» заполняются атомами металла (алюминия или меди). Так устанавливаются связи между несколькими десятками слоев, формирующими сложную трехмерную электрическую схему.

На конечном этапе процессор встраивается в защитный корпус (для электрического соединения кремниевой микросхемы с корпусом используется золото), который обеспечивает электрическое соединение процессора с системной платой.

В первом процессоре 4004 (1971 год) размеры каждого элемента составляли $10 \text{ мк} = 10^{-5} \text{ м}$ (были сравнимы с толщиной человеческого волоса), а их количество было равно 2300. В современных процессорах (2006 год) размеры каждого элемента составляют всего 65 нм (нанометров) $= 0,065 \text{ мк} = 6,5 \cdot 10^{-8} \text{ м}$ (сравнимы с размерами нескольких десятков атомов), а их количество равно 376 000 000.

Плакаты. Таблица 1 «Процессоры»

Производительность. Производительность процессора является его интегральной характеристикой и характеризует скорость выполнения приложений. Производительность процессора прямо пропорциональна разрядности процессора, его частоте, а также количеству команд, выполняемых за один такт:

$$\text{Производительность} \sim \text{Разрядность} \times \text{Частота} \times \\ \times \text{Кол-во команд за такт.}$$

Разрядность процессора определяется количеством двоичных разрядов, которые процессор обрабатывает за один такт. С момента появления первого процессора 4004 (за 35 лет) разрядность процессоров увеличилась в 16 раз (с 4 битов до 64 битов).

Однако в настоящее время увеличение разрядности процессоров (а следовательно, производительности) практически нецелесообразно, так как только сейчас происходит переход от 32-битовых операционных систем и приложений к 64-битовым.

Частота соответствует количеству тактов обработки данных, которые процессор производит за 1 секунду. С момента появления первого процессора частота процессоров увеличилась в 37 000 раз (с 0,1 МГц до 3700 МГц).

Однако увеличение производительности процессоров за счет увеличения частоты имеет свой предел из-за тепловыделения. Выделение процессором теплоты Q пропорционально потребляемой мощности P , которая, в свою очередь, пропорциональна квадрату частоты ν^2 :

$$Q \sim P \sim \nu^2.$$

Уже в настоящее время (2006 год) для отвода тепла от процессора используются массивные воздушные системы охлаждения (их называют кулерами) (рис. 1.7), состоящие из вентилятора и металлических теплоотводящих ребер. Разрабатываются и применяются также водяные системы охлаждения процессоров.



Рис. 1.7. Кулер для процессора

Практическое задание «Определение температуры процессора». С помощью программы тестирования SIV определить температуру процессора и скорость вращения кулера процессора.



Определение температуры процессора

1. Запустить программу тестирования SIV и выбрать вкладку *Hardware Monitor* (рис. 1.8).

На панели отобразится температура процессора и количество оборотов в минуту кулера процессора, в данном случае:

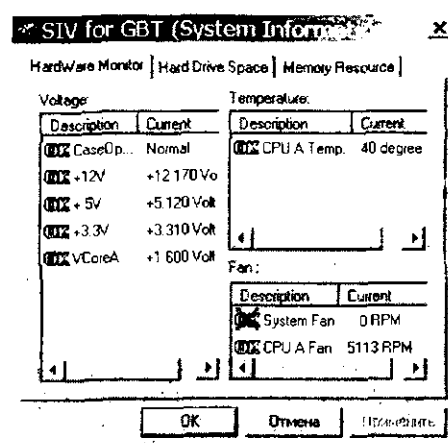


Рис. 1.8. Определение температуры процессора

- температура процессора — 40 градусов по Цельсию;
- количество оборотов в минуту кулера процессора — 5 113 об/мин.

В настоящее время производительность процессоров увеличивается путем совершенствования архитектуры процессора. Во-первых, в структуру процессора вводится кэш-память 1-го и 2-го уровней, которая позволяет ускорить выборку команд и данных и тем самым уменьшить время выполнения одной команды. Во-вторых, вместо одного ядра процессора (АЛУ и др.) используется два ядра, что позволяет повысить производительность процессора примерно на 80%.

Производительность процессора нельзя вычислить, она определяется в процессе тестирования, по скорости выполнения процессором определенных операций в какой-либо программной среде.

Практическое задание «Производительность процессора». С помощью программы тестирования SiSoft Sandra определить производительность вашего процессора и сравнить ее с производительностью:

- аналогичного процессора;
- высокопроизводительного двухъядерного процессора фирмы Intel;
- высокопроизводительного двухъядерного процессора фирмы AMD;
- процессора Pentium начального уровня.



Производительность процессора

1. Запустить программу тестирования SiSoft Sandra и выбрать тестирующий модуль *Арифметический тест процессора*.
2. Из списка *Эталонный процессор 1* выбрать аналогичный процессор.
3. Из списка *Эталонный процессор 2* выбрать высокопроизводительный двухъядерный процессор фирмы Intel.
4. Из списка *Эталонный процессор 3* выбрать высокопроизводительный двухъядерный процессор фирмы AMD.
5. Из списка *Эталонный процессор 4* выбрать процессор Pentium начального уровня.
6. Щелкнуть по кнопке *Обновить*, и через несколько секунд появится информация о производительности вашего процессора.
7. Для каждого процессора будут графически и численно выведены производительности (рис. 1.9):
 - количество целочисленных операций, выраженных в MIPS (Million Instructions Per Second — миллионах операций в секунду);
 - количество операций с плавающей точкой, выраженных в MFLOPS (Million FLoat Operations Per Second — миллионах операций с плавающей точкой в секунду).

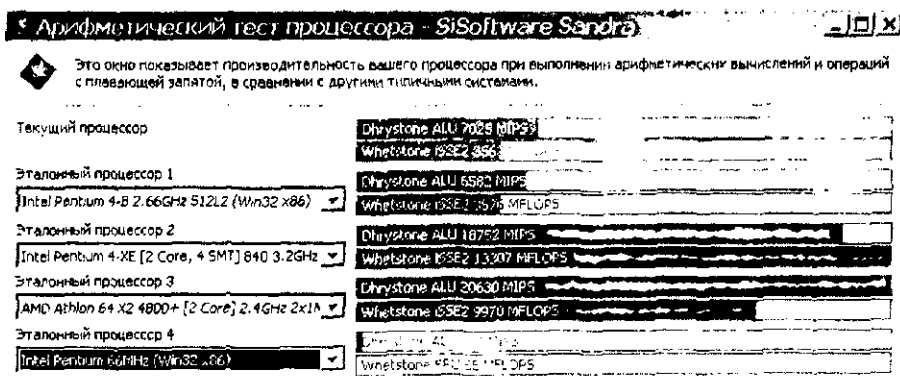






Рис. 1.9. Производительность процессора

Контрольные вопросы

1. Чем будет отличаться упрощенная логическая схема двухъядерного процессора от упрощенной логической схемы одноядерного процессора?
2. Опишите технологию производства процессоров.
3. Какие факторы влияют на производительность процессоров и как можно ее увеличить?

Компьютерный практикум

Windows-CD 

- 1.2. Ознакомиться с историей развития процессоров, используя материалы виртуального музея фирмы Intel. <http://www.intel.ru>
- 1.3. С помощью программы тестирования компьютера CPU-Z выполнить практическое задание «Определение объемов кэш-памяти процессора». 
- 1.4. С помощью программы тестирования компьютера SIV выполнить практическое задание «Определение температуры процессора». 
- 1.5. С помощью программы тестирования компьютера SiSoft Sandra выполнить практическое задание «Производительность процессора». 

1.2.2. Оперативная память

Логическая структура оперативной памяти. Оперативная память представляет собой множество ячеек, причем каждая ячейка имеет свой уникальный адрес (нумерация ячеек начинается с нуля).

Каждая ячейка памяти имеет объем 1 байт, следовательно, максимальный объем адресуемой памяти равен произведению количества ячеек N на 1 байт. Для процессоров Pentium 4, у которых разрядность шины адреса составляет 36 битов, максимальный объем адресуемой памяти равен (см. табл. 1.1):

$$\begin{aligned}
 N \times 1 \text{ байт} &= 2^l \times 1 \text{ байт} = 2^{36} \times 1 \text{ байт} = \\
 &= 68\ 719\ 476\ 736 \text{ байт} = 67\ 108\ 864 \text{ Кбайт} = \\
 &= 65\ 536 \text{ Мбайт} = 64 \text{ Гбайт}.
 \end{aligned}$$

Таблица 1.1. Оперативная память

Объем памяти	Ячейки	Десятичный адрес ячейки	Шестнадцатеричный адрес ячейки
64 Гбайт	10101010	68 719 476 735	FFFFFFFF
...
4 Гбайт	10101010	4 294 967 295	FFFFFFFF
...
	10101010	0	0

Модули оперативной памяти. Оперативная память предназначена для хранения информации. Изготавливается в виде модулей памяти. Модули памяти представляют собой пластины с рядами контактов, на которых размещаются микросхемы памяти. Модули памяти (DDR, DDR2 и др., рис. 1.10) уста-

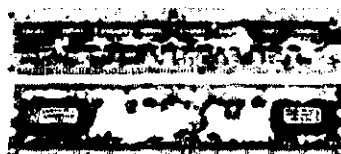


Рис. 1.10. Модули памяти DDR и DDR2

навливаются в специальные разъемы на системной плате и могут различаться между собой по количеству контактов, по быстродействию, по информационной емкости и т. д.

В персональных компьютерах величина адресного пространства процессора (объем адресуемой памяти) и величина фактически установленной оперативной памяти (модулей оперативной памяти) практически всегда различаются. Хотя объем адресуемой памяти у большинства современных процессоров равен 64 Гбайт, величина фактически установленной оперативной памяти может быть значительно меньше, например, может быть установлено 4 модуля памяти по 1 Гбайт («всего» 4 Гбайт).

Пропускная способность. Важнейшей характеристикой модулей оперативной памяти является пропускная способность, которая равна произведению разрядности шины данных и частоты операций записи или считывания информации из ячеек памяти:

$$\begin{aligned} \text{Пропускная способность} &= \\ &= \text{Разрядность шины данных} \times \text{Частота}. \end{aligned}$$

Разрядность шины данных составляет 64 бита, а максимально возможная в настоящее время (2006 год) частота шины данных совпадает с частотой системной шины и равна 1064 МГц. Тогда пропускная способность модулей памяти, подключенных к этой шине, равна:

$$\begin{aligned} \text{Пропускная способность} &= 64 \text{ бита} \times 1064 \text{ МГц} = \\ &= 68\,096 \text{ Мбит/с} = 8512 \text{ Мбайт/с} \approx 8 \text{ Гбайт/с}. \end{aligned}$$



Модули памяти маркируются своей пропускной способностью, выраженной в Мбайт/с: PC3200, PC4200, PC8500 и др.

Физическая и виртуальная память. Объем используемой программами памяти можно увеличить путем добавления к физической памяти (модулям оперативной памяти) виртуальной памяти. Виртуальная память выделяется в форме области жесткого диска. В ОС Windows это файл подкачки. По своей логической организации виртуальная память является частью оперативной памяти.

Размер файла подкачки и его размещение в иерархической файловой системе можно изменять. Однако необходимо учитывать, что быстродействие жесткого диска и, соответственно, виртуальной памяти существенно меньше быстродействия модулей оперативной памяти.

Замедление быстродействия виртуальной памяти может происходить в результате фрагментации данных в файле. Для того чтобы этого не происходило, рекомендуется произвести дефрагментацию диска и установить для файла подкачки постоянный размер.

Практическое задание «Виртуальная память». В операционной системе Windows и с использованием программы тестирования компьютера:

- установить объем виртуальной памяти и местоположение файла подкачки;
- определить объем виртуальной памяти и местоположение файла подкачки;
- определить загруженность процессора и объем используемой виртуальной памяти.



Виртуальная память

1. Щелкнуть правой кнопкой мыши по значку *Мой компьютер* и в контекстном меню активизировать пункт *Свойства*.
2. В диалоговом окне *Свойства системы* выбрать вкладку *Дополнительно* и в разделе *Быстродействие* щелкнуть по кнопке *Параметры*.
3. В появившемся диалоговом окне *Параметры быстродействия* выбрать вкладку *Дополнительно*. В разделе *Виртуальная память* щелкнуть по кнопке *Изменить*.

4. В появившемся диалоговом окне *Виртуальная память* (рис. 1.11) выбрать место размещения файла подкачки.

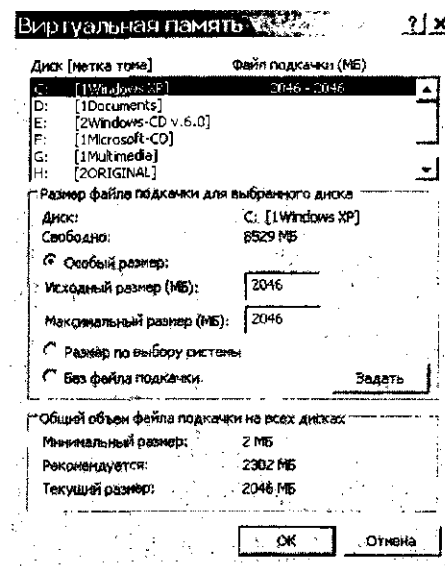


Рис. 1.11. Установка размера и местоположения виртуальной памяти

5. Для предотвращения фрагментации файла подкачки установить одинаковые значения для его исходного и максимального размеров.

Важное значение для производительности компьютера имеет объем оперативной памяти, причем не только объем установленных модулей оперативной памяти, но и общий объем системной оперативной памяти с учетом объема файла подкачки `C:\pagefile.sys`.



Виртуальная память

6. Запустить программу тестирования SiSoft Sandra и выбрать информационный модуль *Информация о памяти Windows*.

В окне появится информация об объемах памяти тестируемого компьютера, например (рис. 1.12):

всего физической памяти — 1,5 Гбайт;
 файл подкачки — `C:\pagefile.sys` 2046 2046.

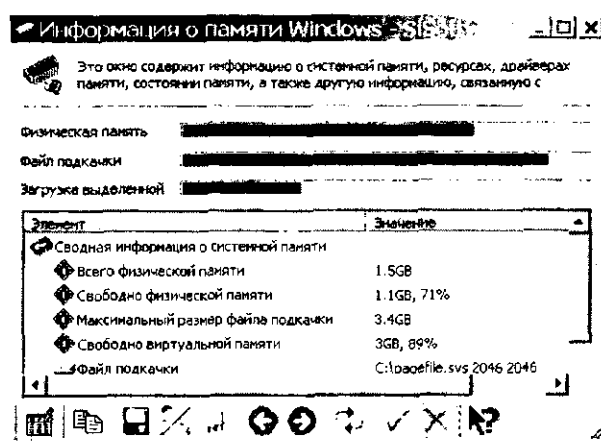


Рис. 1.12. Определение размера и местоположения виртуальной памяти

Производительность компьютера зависит не только от частоты процессора и объема системной оперативной памяти, но и от загрузки процессора и процента использования виртуальной памяти. Производительность компьютера уменьшается, если загрузка процессора достигает 100%, а также начинает использоваться более медленная, чем физическая, виртуальная память на жестком диске. Загрузку процессора и использование виртуальной памяти в операционной системе Windows 9x/Me можно наблюдать с помощью служебной программы Системный монитор, а в операционной системе Windows NT/2000/XP для этого используется Диспетчер задач Windows.



Виртуальная память

- Для вызова диспетчера задач одновременно нажать три клавиши $\{Ctrl\} + \{Alt\} + \{Del\}$.

На диалоговой панели *Диспетчер задач Windows* (рис. 1.13) выбрать вкладку *Быстродействие* и наблюдать с помощью гистограмм и графиков изменение во времени загрузки процессора и использования виртуальной памяти.

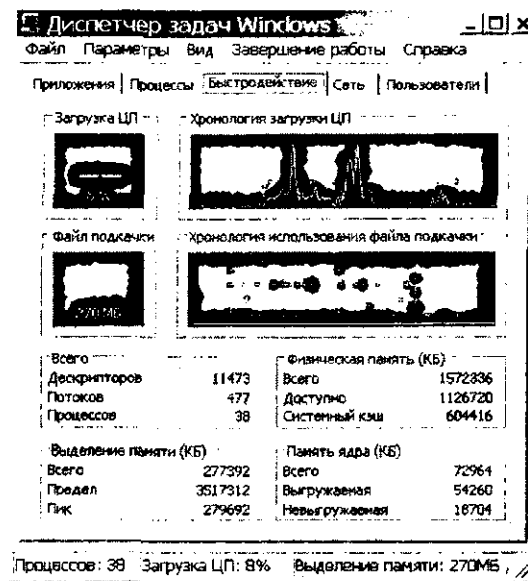


Рис. 1.13. Определение загруженности процессора и использования виртуальной памяти

Контрольные вопросы

1. В чем состоит различие между максимальным объемом адресуемой оперативной памяти и физически установленной оперативной памятью?
2. В чем состоит различие между физически установленной оперативной памятью и виртуальной памятью?
3. От каких факторов зависит пропускная способность модулей оперативной памяти?




Задачи

- 1.1. Вычислить частоту шины памяти, к которой подключены модули оперативной памяти PC3200.

Компьютерный практикум



Windows-CD

- 1.6. В операционной системе Windows выполнить первую часть практического задания «Виртуальная память» — установить объем виртуальной памяти и местоположение файла подкачки. 
- 1.7. С помощью программы тестирования компьютера SiSoft Sandra выполнить вторую часть практического задания «Виртуальная память» — определить объем виртуальной памяти и местоположение файла подкачки. 
- 1.8. В операционной системе Windows выполнить третью часть практического задания «Виртуальная память» — определить загруженность процессора и объем используемой виртуальной памяти. 

1.3. Внешняя (долговременная) память

1.3.1. Магнитная память

Основной функцией внешней памяти компьютера является долговременное хранение большого объема информации (программы, документы, аудио- и видеоклипы и т. д.). Устройство, которое обеспечивает запись/считывание информации, называется **накопителем** или **дисководом**, а хранится информация на **носителях** (например, дисках).

Магнитный принцип записи и считывания информации. В накопителях на гибких магнитных дисках (НГМД) и накопителях на жестких магнитных дисках (НЖМД), или «винчестерах», в основу записи информации положено намагничивание ферромагнетиков в магнитном поле, хранение информации основывается на сохранении намагниченности, а считывание информации базируется на явлении электромагнитной индукции.

Физика-10

В процессе записи информации на гибкие и жесткие магнитные диски головка дисковода с сердечником из магнитомягкого материала (малая остаточная намагниченность) перемещается вдоль магнитного слоя магнито жесткого носителя (большая остаточная намагниченность). На магнитную головку поступают последовательности электрических импу-

льсов (последовательности логических единиц и нулей), которые создают в головке магнитное поле. В результате последовательно намагничиваются (логическая единица) или не намагничиваются (логический нуль) элементы поверхности носителя.

При считывании информации при движении магнитной головки над поверхностью носителя намагниченные участки носителя вызывают в ней импульсы тока (явление электромагнитной индукции). Последовательности таких импульсов передаются по магистрали в оперативную память компьютера.

В отсутствие сильных магнитных полей и высоких температур элементы носителя могут сохранять свою намагниченность в течение долгого времени (лет и десятилетий).

Гибкие магнитные диски. Гибкие магнитные диски помещаются в пластмассовый корпус. Такой носитель информации называется дискетой. В центре дискеты имеется приспособление для захвата и обеспечения вращения диска внутри пластмассового корпуса. Дискета вставляется в дисковод, который вращает диск с постоянной угловой скоростью.

При этом магнитная головка дисковода устанавливается на определенную концентрическую дорожку диска, на которую и производится запись или с которой производится считывание информации. Информационная емкость современной дискеты невелика и составляет всего 1,44 Мбайт. Скорость записи и считывания информации также мала (составляет всего около 50 Кбайт/с) из-за медленного вращения диска (360 об/мин).

В целях сохранения информации гибкие магнитные диски необходимо предохранять от воздействия сильных магнитных полей и нагревания, так как такие физические воздействия могут привести к размагничиванию носителя и потере информации.

Жесткие магнитные диски. Жесткие магнитные диски (рис. 1.14) представляют собой несколько дисковых магнитных пластин, размещенных на одной оси, заключенных в металлический корпус и вращающихся с большой угловой скоростью. Обе стороны каждой пластины покрыты тонким слоем намагничиваемого материала.

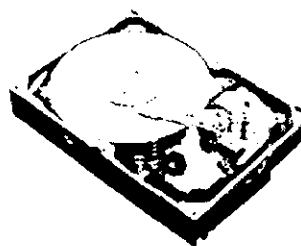


Рис. 1.14. Жесткий диск

У каждой магнитной стороны каждой пластины есть своя магнитная головка чтения/записи, головки также соединены вместе и движутся радиально по отношению к магнитным пластинам, обеспечивая, таким образом, доступ к любой дорожке любой пластины. Встроенная электроника предназначена для управления движением магнитных головок чтения/записи.

За счет использования нескольких дисковых магнитных пластин и гораздо большего количества дорожек на каждой стороне магнитных пластин информационная емкость жестких дисков может в сотни тысяч раз превышать информационную емкость дискет и достигать 500 Гбайт. Скорость записи и считывания информации на жестких дисках может достигать 300 Мбайт/с (по шине SATA) за счет быстрого позиционирования магнитной головки и высокой скорости вращения дисков (до 7200 об/мин).


В жестких дисках используются достаточно хрупкие и миниатюрные элементы (магнитные пластины носителей, магнитные головки и т. д.), поэтому в целях сохранения информации и работоспособности жесткие диски необходимо оберегать от ударов и резких изменений пространственной ориентации в процессе работы.

Контрольные вопросы

1. Почему сердечник магнитной головки изготавливается из магнитомягкого материала, а магнитный слой носителя — из магнитожесткого материала?
2. Как можно увеличить информационную емкость жестких дисков?

1.3.2. Оптическая память

Оптический принцип записи и считывания информации. В лазерных CD- и DVD-дисководах используется оптический принцип записи и считывания информации.

Физика-11 

В процессе записи информации на оптические диски для создания участков поверхности с различными коэффициентами отражения применяются различные технологии: от простой штамповки до изменения отражающей способности участков поверхности диска с помощью мощного лазера.

Информация на лазерном диске записывается на одну спиральную дорожку, начинающуюся от центра диска и содержащую чередующиеся участки с различной отражающей способностью.

В процессе считывания информации с оптических дисков луч лазера, установленного в дисковом устройстве, падает на поверхность вращающегося диска и отражается. Так как поверхность оптического диска имеет участки с различными коэффициентами отражения, то отраженный луч также меняет свою интенсивность (логический 0 или 1). Затем отраженные световые импульсы преобразуются с помощью фотоэлементов в электрические импульсы и по магистрали передаются в оперативную память.

При соблюдении правил хранения (хранения в футлярах в вертикальном положении) и эксплуатации (без нанесения царапин и загрязнений) оптические носители могут сохранять информацию в течение десятков лет.

Оптические диски (рис. 1.15). Оптические CD-диски рассчитаны на использование инфракрасного лазера с длиной волны 780 нм и имеют информационную емкость 700 Мбайт.

Оптические DVD-диски рассчитаны на использование красного лазера с длиной волны 650 нм. Они имеют большую информационную емкость по сравнению с CD-дисками (4,7 Гбайт) за счет меньшей ширины и более плотного размещения оптических дорожек. DVD-диски могут быть двухслойными (емкость 8,5 Гбайт), при этом оба слоя имеют отражающую поверхность, несущую информацию.



Рис. 1.15. CD- и DVD-диски

В настоящее время (2006 год) на рынок поступили оптические диски (HD DVD и Blu-Ray), информационная емкость которых в 3–5 раз превосходит информационную емкость DVD-дисков за счет использования синего лазера с длиной волны 405 нанометров.

На дисках CD-ROM и DVD-ROM хранится информация, которая была записана на них в процессе изготовления. Запись на них новой информации невозможна, что отражено во второй части их названий: ROM (Read Only Memory — память только для чтения). Производятся такие диски путем штамповки на дорожке микроскопических физических углублений (участков с плохой отражающей способностью).

На дисках CD-R и DVD±R (R — recordable, записываемый) информация может быть записана, но только один раз. Данные записываются на диск лучом лазера повышенной мощности, который разрушает органический краситель записывающего слоя и меняет его отражательные свойства. Управляя мощностью лазера, на записывающем слое получают чередование темных и светлых пятен, которые при чтении интерпретируются как логические 0 и 1.

На дисках CD-RW и DVD±RW (RW — ReWritable, перезаписываемый) информация может быть записана и стерта многократно. Записывающий слой изготавливается из специального сплава, который можно нагреванием приводить в два различных устойчивых агрегатных состояния — аморфное и кристаллическое. При записи (или стирании) луч лазера нагревает участок дорожки и переводит его в одно из устойчивых агрегатных состояний, которые характеризуются различной степенью прозрачности. При чтении луч лазера имеет меньшую мощность и не изменяет состояние записывающего слоя, а чередующиеся участки с различной прозрачностью интерпретируются как логические 0 и 1.

Оптические дисководы. Оптические CD- и DVD-дисководы используют лазер для чтения или записи информации. Скорость чтения/записи информации в оптических дисководах зависит от скорости вращения диска.

Первые CD-дисководы были односкоростными и обеспечивали скорость считывания информации 150 Кбайт/с. В настоящее время широкое распространение получили CD-дисководы, которые обеспечивают в 52 раза большую скорость чтения и записи дисков CD-R (до 7,8 Мбайт/с). Запись CD-RW дисков производится на меньшей скорости (например, 32-кратной), поэтому CD-дисководы маркируются тремя числами «скорость чтения × скорость записи CD-R × скорость записи CD-RW» (например, «52×52×32»).

Первое поколение DVD-накопителей обеспечивало скорость считывания информации примерно 1,3 Мбайт/с. В настоящее время широкое распространение получили DVD-дис-

ководы, которые обеспечивают в 16 раз большую скорость чтения (примерно 21 Мбайт/с), в 8 раз большую скорость записи DVD±R дисков и в 6 раз большую скорость записи DVD±RW дисков. DVD-дисководы маркируются тремя числами (например, «16×8×6»).

Контрольные вопросы

1. Почему в CD-дисководах используется инфракрасный лазер, в DVD-дисководах — красный лазер, а в HD DVD- и Blu-Ray-дисководах — синий лазер?
2. В чем состоит различие между дисками CD-ROM, CD-R и CD-RW?
3. Что означают числа маркировки DVD-дисководов?

1.3.3. Флэш-память


Флэш-память — особый вид полупроводниковой энерго-независимой перезаписываемой памяти:

- полупроводниковая (твердотельная) — не содержащая механически движущихся частей (как обычные жесткие диски или CD), построенная на основе полупроводниковых микросхем;
- энергонезависимая — не требующая дополнительной энергии для хранения данных (энергия требуется только для записи);
- перезаписываемая — допускающая изменение (перезапись) хранимых в ней данных.



Название флэш-памяти было дано во время разработки первых микросхем (в начале 1980-х годов) как характеристика скорости стирания флэш-памяти (от англ. «in a flash» — в мгновение ока).

Принцип записи и считывания информации на картах флэш-памяти. Во флэш-памяти для записи и считывания информации используются электрические сигналы.

Физика-11 

В простейшем случае каждая ячейка флэш-памяти хранит один бит информации и состоит из одного полевого транзистора со специальной электрически изолированной областью («плавающим» затвором) (рис. 1.16).

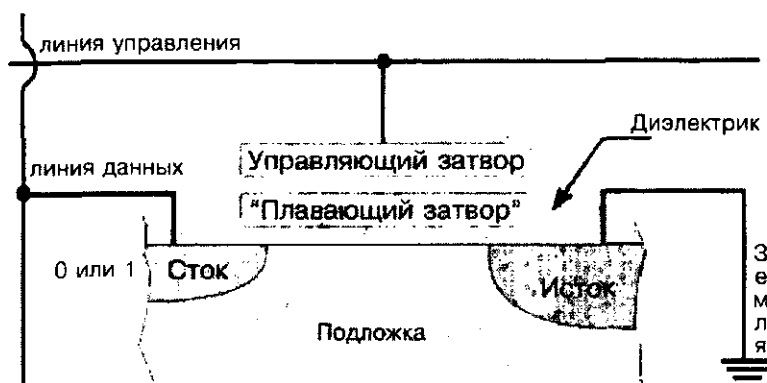


Рис. 1.16. Ячейка флэш-памяти на полевом транзисторе

При отсутствии сигнала на линии управления (на управляющем затворе) ячейка памяти хранит один бит информации (0 или 1) на стоке полевого транзистора. Между стоком и истоком ток не идет.

При записи данных на линию управления подается положительное напряжение и электроны в результате эффекта туннелирования попадают на плавающий затвор. Между стоком и истоком возникает электрический ток и в результате на стоке полевого транзистора записывается один бит данных.



Эффект туннелирования — один из квантомеханических эффектов, использующих волновые свойства электрона. В полевом транзисторе этот эффект заключается в «перескакивании» электрона через слой диэлектрика между стоком и «плавающим» затвором под действием электрического поля.

Карты флэш-памяти (рис. 1.17). Флэш-память представляет собой микросхему, помещенную в миниатюрный плоский корпус. Микросхемы флэш-памяти могут содержать миллиарды ячеек, каждая из которых хранит 1 бит информации. Информационная емкость карт флэш-памяти может достигать 4 Гбайт (2006 год).

Информация, записанная на флэш-память, может храниться очень длительное время (от 20 до 100 лет) и способна выдерживать значительные механические нагрузки (в 5–10 раз превышающие предельно допустимые для жестких дисков). Флэш-память компактнее и потребляет значительно меньше энергии (примерно в 10–20 раз), чем магнитные и оптические дисководы.

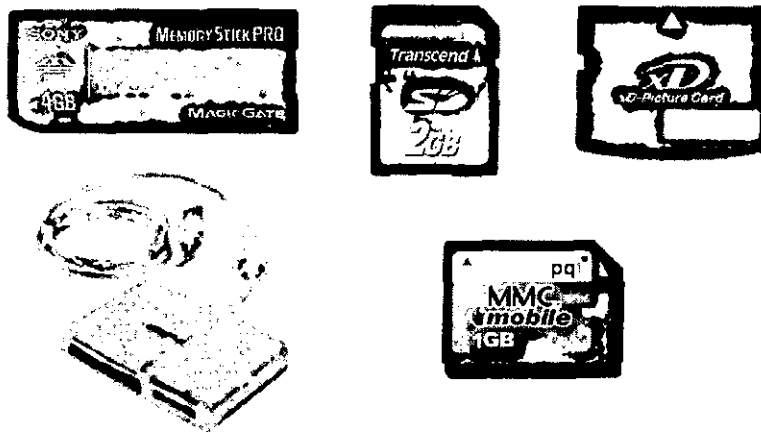


Рис. 1.17. Карты флэш-памяти и картридер

Благодаря низкому энергопотреблению, компактности, долговечности и относительно высокому быстродействию, флэш-память идеально подходит для использования в качестве носителя информации в портативных устройствах: цифровых фото- и видеокамерах, сотовых телефонах, портативных компьютерах, MP3-плеерах и цифровых диктофонах.

К недостаткам флэш-памяти следует отнести то, что не существует единого стандарта и различные производители изготавливают несовместимые друг с другом по размерам и электрическим параметрам карты памяти (Memory Stick, MultiMedia Card, Secure Digital, xD-Picture и др.). Для считывания или записи информации карта памяти вставляется в специальные накопители (картридеры), встроенные в мобильные устройства или подключаемые к компьютеру через USB-порт.

USB флэш-диски (рис. 1.18). Накопители на флэш-памяти представляет собой микросхему флэш-памяти, дополненную контроллером USB, и подключаются к последовательно-

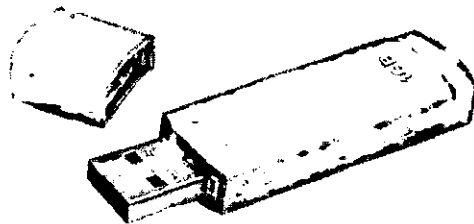


Рис. 1.18. USB флэш-диск

му порту USB. USB флэш-диски могут использоваться в качестве внешнего сменного носителя информации.

USB флэш-диски могут содержать переключатель защиты от записи, поддерживать парольную защиту, а также могут быть загрузочными. Некоторые USB флэш-диски имеют кроме индикатора работы жидкокристаллический экранчик, на котором отображается, сколько свободного места осталось на диске.

Контрольные вопросы

1. В чем состоит отличие микросхем флэш-памяти от микросхем оперативной памяти?
2. В чем состоит преимущество флэш-памяти перед магнитной и оптической долговременной памятью?

1.4. Файл и файловые системы

1.4.1. Логическая структура носителя информации

2.3.2. Файловая система

Информатика-8



Логическая структура носителя информации в распространенной файловой системе FAT имеет следующие разделы:

- загрузочный кластер;
- таблицу размещения файлов, которая содержит в своих ячейках цепочку номеров кластеров для каждого файла;
- корневой каталог;
- файлы.

В файловой системе FAT используется следующая организация хранения.

Минимальным адресуемым элементом носителя информации является кластер, который может включать в себя несколько секторов (объем сектора составляет 512 байтов). Размер кластера (от 512 байтов до 64 Кбайт) зависит от типа используемой файловой системы. Кластеры нумеруются в линейной последовательности (на магнитных дисках от первого кластера нулевой дорожки до последнего кластера последней дорожки).

Файловая система организует кластеры в файлы и каталоги (реально являющиеся файлами, содержащими список файлов в этом каталоге). Файловая система отслеживает, ка-

кие из кластеров в настоящее время используются, какие свободны, какие помечены как неисправные.

При записи файлов будет занято всегда целое количество кластеров, соответственно минимальный размер файла равен размеру одного кластера, а максимальный соответствует общему количеству кластеров на диске.

Файл записывается в произвольные свободные кластеры. Например, Файл_1 может занимать кластеры 34, 35 и 47, 48, а Файл_2 — кластеры 36 и 49 (табл. 1.2).

Таблица 1.2. Логическая структура носителя информации

№ кластера																	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
...																	

Таблица размещения файлов. Полная информация о кластерах, которые занимают файлы, содержится в таблице размещения файлов FAT. В целях более надежного сохранения информации о размещении файлов таблица FAT хранится на носителе информации в двух идентичных копиях. Количество ячеек FAT соответствует количеству кластеров на диске, а значениями ячеек являются цепочки размещения файлов, т. е. последовательности адресов кластеров, в которых хранятся файлы.

Например, для двух рассмотренных выше файлов таблица FAT с 1-й по 54-ю ячейку принимает следующий вид (табл. 1.3).

Таблица 1.3. Фрагмент FAT

															35	47	49
										48	К	К					

Цепочка размещения для файла Файл_1 выглядит следующим образом: в начальной 34-й ячейке FAT хранится адрес следующего кластера (35), соответственно, в следующей 35-й ячейке хранится 47, в 47-й — 48, в 48-й — знак конца файла (К).

FAT12. Файловая система для операционной системы Windows. Выделяет 12 битов для хранения адреса кластера,

соответственно, она может адресовать $2^{12} = 4096$ кластеров. Объем кластера по умолчанию равен размеру одного сектора (512 байтов), и поэтому FAT12 не может использоваться для носителей информации объемом более:

$$\begin{aligned} 512 \text{ байт} \times 4096 &= 2\,097\,152 \text{ байт} = \\ &= 2048 \text{ Кбайт} = 2 \text{ Мбайт}. \end{aligned}$$

Такой объем имеют дискеты, поэтому FAT12 используется именно для дискет.

FAT16. Файловая система для операционной системы Windows. Выделяет 16 битов для хранения адреса кластера, соответственно, она может адресовать $2^{16} = 65\,536$ кластеров. Объем кластера не может быть более 128 секторов (64 Кбайт), и поэтому FAT16 не может использоваться для носителей информации объемом более:

$$\begin{aligned} 64 \text{ Кбайт} \times 65\,536 &= 4\,194\,304 \text{ Кбайт} = \\ &= 4096 \text{ Мбайт} = 4 \text{ Гбайт}. \end{aligned}$$

В настоящее время такой объем имеет флэш-память, поэтому FAT16 используется именно для флэш-памяти.

FAT32. Файловая система для операционной системы Windows. Выделяет 32 бита для хранения адреса кластера, соответственно, она может адресовать $2^{32} = 4\,294\,967\,296$ кластеров. Объем кластера по умолчанию составляет 8 секторов (4 Кбайт), и поэтому FAT32 может использоваться для носителей информации объемом:

$$\begin{aligned} 4 \text{ Кбайт} \times 4\,294\,967\,296 &= 17\,179\,869\,184 \text{ Кбайт} = \\ &= 16\,777\,216 \text{ Мбайт} = 16\,384 \text{ Гбайт} = 16 \text{ Тбайт}. \end{aligned}$$

Таким образом, FAT32 может использоваться для жестких дисков самого большого объема.

NTFS. Файловая система для операционной системы Windows. Позволяет устанавливать различный объем кластера (от 512 байтов до 64 Кбайт, по умолчанию 4 Кбайт). NTFS по сравнению с FAT32 увеличивает надежность и эффективность использования дискового пространства.

NTFS использует систему журналирования для повышения надежности файловой системы. Журналируемая файловая система сохраняет список изменений, которые она будет проводить с файловой системой, перед фактической записью изменений. Эти записи хранятся в отдельной части файловой системы, называемой «журналом» или «логом». Как только изменения файловой системы будут внесены в журнал, журналируемая файловая система применит эти изменения к файлам.

ext3 и ReiserFS. Журналируемые файловые системы для операционных систем Unix. ReiserFS — высоконадежная файловая система, хорошо приспособленная для хранения большого количества маленьких файлов. Блок (кластер) ext3 может иметь размер от 1 до 8 Кбайт, а в ReiserFS в одном блоке могут быть размещены данные нескольких файлов. С файлами большого размера файловая система ReiserFS также справляется весьма уверенно, максимальный размер файловой системы составляет 16 Тбайт.

HFS. Иерархическая журналируемая файловая система, разработанная Apple Computer для использования на компьютерах, работающих под управлением операционной системы Mac OS.

CDFS. Файловая система для работы с оптическими CD- и DVD-дисками, базирующаяся на стандарте ISO 9660, согласно которому имя файла не может превышать 32 символа и глубина вложения папок — не более 8 уровней.

UDF. Мультисистемная файловая система для работы с файлами на оптических дисках позволяет на перезаписываемых CD-RW и DVD±RW дисках удалять, копировать и сохранять отдельные файлы.

Практическое задание «Объем файла в различных файловых системах». В операционной системе Windows создать текстовый файл и последовательно сохранить его на гибком диске, на флэш-диске и на жестком диске. Ознакомиться с объемом текстового файла и занимаемым им объемом на дисках с различными файловыми системами.



Объем файла в различных файловых системах

1. Запустить Блокнот командой [*Программы-Стандартные-Блокнот*]. Ввести текст в файл, например «информатика».
2. Сохранить файл на гибком диске, на флэш-диске и на жестком диске.

На каждом диске определить объем, занимаемый файлом.

3. Последовательно в контекстном меню дисков активизировать пункт *Свойства*. В появившихся диалоговых окнах ознакомиться с объемом текстового файла и занимаемым им объемом на дисках (рис. 1.19).

Информационный объем текста равен:

$1 \text{ байт} \times 11 = 11 \text{ байт}$.

На гибком диске используется файловая система FAT12, и этот файл будет занимать один сектор, т. е. 512 байтов.

На флэш-диске используется файловая система FAT16, и этот файл будет занимать один кластер, объем которого равен 32 Кбайт.

На жестком диске используется файловая система FAT32 или NTFS, и этот файл будет занимать один кластер, объем которого равен 4 Кбайт.

Размещение:	A:\
Размер:	11 байт (11 байт)
На диске:	512 байт (512 байт)
Размещение:	N:\
Размер:	11 байт (11 байт)
На диске:	32,0 КБ (32 768 байт)
Размещение:	D:\
Размер:	11 байт (11 байт)
На диске:	4,00 КБ (4 096 байт)

Рис. 1.19. Объем файла на дисках с различными файловыми системами

Форматирование носителей информации. Для того чтобы на носителе можно было хранить информацию, он должен быть отформатирован, т. е. создана логическая структура в соответствии с определенной файловой системой.



Жесткие диски должны быть предварительно физически отформатированы (это делается на заводе-изготовителе). Формирование физической структуры диска состоит в создании на диске концентрических дорожек, которые, в свою очередь, делятся на секторы (в определенных местах диска расставляются метки дорожек и секторов).

Большие по объему жесткие диски рекомендуется разбивать на **разделы**, т. е. независимые области на диске. Разделы могут быть отформатированы в различных файловых системах, и, таким образом, созданы логические диски. Логическим дискам ставятся в соответствие заглавные буквы латинского алфавита (C:, D: и т. д.).

На один физический диск в разные разделы могут быть установлены различные операционные системы.

Интерфейс командной строки. Давным-давно (в 80-е годы XX века), когда еще не был разработан графический интерфейс операционных систем, форматирование дисков и другие операции с файлами проводились с использованием командной строки операционной системы MS-DOS. В операционной

системе Windows предусмотрен режим работы с интерфейсом командной строки.

В ответ на приглашение системы можно вводить команды с клавиатуры, в том числе:

- команды работы с дисками (format, defrag и др.);
- команды работы с файлами (copy, del, rename и др.);
- команды работы с каталогами (cd, dir и др.).

Каждая команда имеет свой формат, параметры которого запомнить достаточно трудно. Для того чтобы получить справочную информацию о команде, необходимо после имени команды ввести ключ */?*.

Практическое задание «Форматирование из командной строки». В операционной системе Windows в командной строке с помощью команды `format` отформатировать дискету с нестандартным размером кластера (например, 1024 байта). Предварительно получить справку о формате команды `format`.



Форматирование из командной строки

1. Ввести команду [Программы-Стандартные-Командная строка]. Появится окно командной строки.
2. В ответ на приглашение системы перейти в каталог C:\Windows с помощью команды CD:
C:\Documents and Settings\НДУ>CD C:\Windows
3. Получить справку о формате команды `format` (рис. 1.20):
C:\Windows>format /?

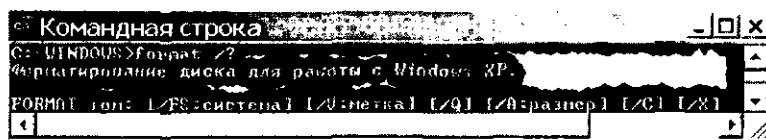


Рис. 1.20. Получение справки по формату команды

4. Отформатировать дискету с нестандартным размером кластера 1024 байта:
C:\Windows>format A:/ A:1024
После окончания форматирования можно узнать количество кластеров, количество байтов в кластере, разрядность ячейки FAT и другие параметры форматирования дискеты (рис. 1.21).

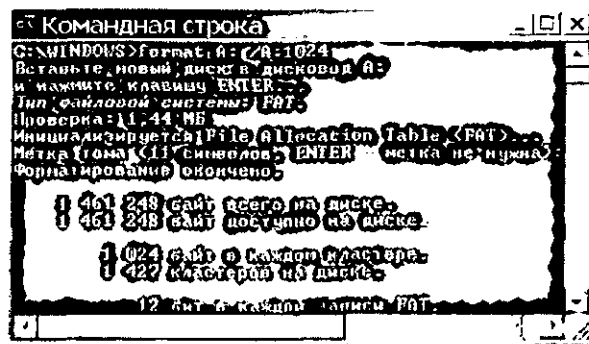





Рис. 1.21. Форматирование дискеты в командной строке

Контрольные вопросы

1. Какую последовательность кластеров занимает файл `Файл_2` из приведенного примера на носителе информации?
2. Чем различаются таблицы размещения файлов FAT12, FAT16 и FAT32?
3. Почему журналируемые файловые системы обеспечивают большую надежность хранения файлов?

Компьютерный практикум

Windows-CD

- 1.9. С помощью средств операционной системы Windows определить для диска C: тип файловой системы, размер кластера и количество кластеров. 
- 1.10. В операционной системе Windows выполнить практическое задание «Объем файла в различных файловых системах». 
- 1.11. В операционной системе Windows выполнить практическое задание «Форматирование из командной строки». 

1.4.2. Файл

Все программы и данные хранятся в долговременной памяти компьютера в виде файлов.

2.3.1. Файл Информатика-8

Файл обязательно имеет имя, формат которого определяется используемой файловой системой. Имя файла состоит из двух частей, разделенных точкой: собственно имени файла и расширения, определяющего тип файла (табл. 1.4). Расширение файла обычно дописывается к имени приложением, в котором этот файл создается. Приложение, с которым связан файл, можно изменить (например, файлы архивов с расширением zip можно связать с различными архиваторами).

Таблица 1.4. Некоторые типы файлов и расширений

Тип файла	Расширения
Исполнимые файлы	exe
Текстовые файлы	txt, doc
Web-страницы	htm, html
Графические файлы	bmp, gif, png, jpg
Звуковые файлы	wav, mid
Видеофайлы	avi, wmf
Архивы	zip, 7z

Файл в каталоге описывается записью, в которой указывается его имя, размер, дата и время создания, а также набор атрибутов:

- «только для чтения» (от англ. «read only») — файл с таким атрибутом доступен операционной системе и приложениям только для чтения, т. е. в нем нельзя сохранить какие-либо исправления;
- «скрытый» (от англ. «hidden») — файл с данным атрибутом не выводится при показе содержимого папок;
- «системный» (от англ. «system») — файлы с таким атрибутом относятся к системным файлам операционной системы, которые нельзя удалять или изменять;
- «архивный» (от англ. «archive») — этот атрибут указывает программам архивации файлы, предназначенные для резервного копирования.



Файл — это некоторая информация (программа или данные), имеющая имя и хранящаяся в долговременной (внешней) памяти.

Практическое задание «Расширение и атрибуты файла». В операционной системе Windows файлы архивов с расширением zip связать с файловым менеджером Total Commander, а затем с помощью этого файлового менеджера изменить атрибуты и дату/время создания файла архива.



Расширение и атрибуты файла

1. Щелкнуть правой кнопкой мыши по имени файла и в контекстном меню активизировать пункт *Открыть с помощью*.
2. Если в появившемся меню нужное приложение отсутствует, выбрать пункт *Выбрать программу...*
3. В появившемся диалоговом окне *Выбор программы* (рис. 1.22) выбрать программу из списка.

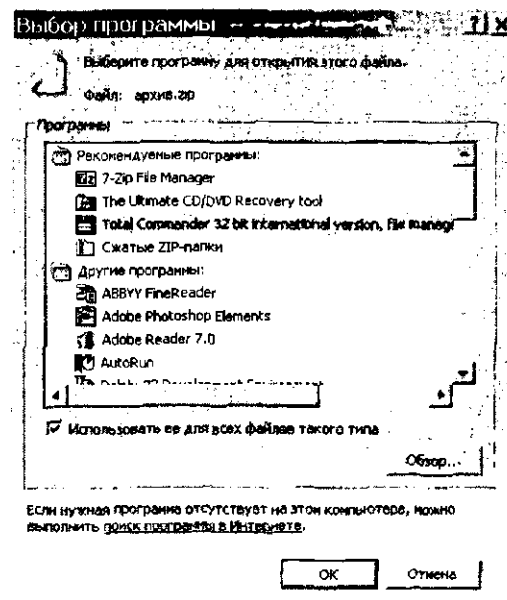


Рис. 1.22. Выбор приложения для открытия файла

4. В случае отсутствия нужного приложения в списке щелкнуть по кнопке *Обзор* и выбрать приложение в иерархической файловой системе.

5. Установить флажок *Использовать ее для всех файлов такого типа*.



Расширение и атрибуты файла

1. Запустить файловый менеджер Total Commander.
2. На панели выбрать файл, например архив.zip.
3. В меню ввести команду [*Файл- Изменить атрибуты...*].
4. В появившемся диалоговом окне *Изменение атрибутов* (рис. 1.23) установить флажок *Скрытый*. Файл перестанет отображаться на панели.

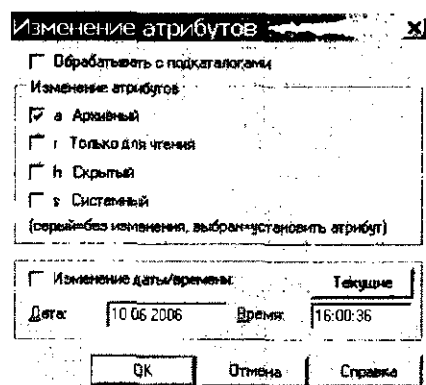


Рис. 1.23. Изменение атрибутов и даты/времени создания файла

5. Установить текущие дату и время создания файла, для этого щелкнуть по кнопке *Текущие*.

Архивация файлов. Для уменьшения места, занимаемого файлами, с целью долговременного хранения или передачи по компьютерным сетям файлы архивируются (сжимаются) с помощью файловых менеджеров или специализированных приложений — **архиваторов**.

Существуют различные алгоритмы архивации данных без потери информации, при которых при разархивации данные будут восстановлены в исходном виде. Самый простой алгоритм сжатия данных основан на замене повторяющихся битов (в тексте может иметься последовательность одинаковых символов, в графическом файле — закрашенная одним цветом область и т. д.) более короткой последовательностью битов. Например, в тексте подряд идут 10 пробелов, которые кодируются 10 байтами. При архивации они заменяются 3 байтами (первый байт кодирует заменяемый символ; второй байт — это

специальный байт «флажка» архивации, который указывает на необходимость развернуть первый байт в последовательность байтов; третий байт указывает количество повторяющихся байтов).

Алгоритм кодирования одинаковых последовательностей символов ищет в текстовых файлах одинаковые подстроки, а в графических — близкие по цвету последовательности пикселей. Каждый такой фрагмент файла представляется более коротким кодом (последовательностью битов) и в процессе архивации при повторных появлениях заменяется ссылкой на первичный код.

Лучше всего сжимаются текстовые и графические файлы, практически не сжимаются файлы архивов и исполнимые файлы.

Существуют различные архиваторы файлов (ZIP, 7-Zip, RAR и др.), которые используют вышеописанные и другие алгоритмы архивации. Архиваторы различаются степенью сжатия файлов, скоростью выполнения и другими параметрами.

При создании нового архива нужно задать параметры архивации:

- задать имя архивного файла и место его сохранения на диске;
- выбрать формат архивации ZIP, 7z, RAR или др.;
- выбрать степень сжатия файлов (от *Без сжатия* — файлы не сжимаются, до *Максимальный* для получения наилучшего сжатия);
- выбрать размер словаря и слова (чем больше их размер, тем больше, но медленнее сжатие);
- можно создать многотомный архив, т. е. архив, состоящий из нескольких частей, которые используются для сохранения большого архива на нескольких дискетах или других сменных носителях;
- можно создать непрерывный (от англ. solid) архив (в архиваторе RAR), что позволяет добиться максимальной степени сжатия нескольких файлов;
- можно создать самораспаковывающийся архив SFX, который является исполнимым файлом и имеет расширение exe. Для разархивации такого архива не нужен архиватор, достаточно запустить файл архива на выполнение.

Для разархивации файлов необходимо открыть файл архива в архиваторе и ввести соответствующую команду.

Практическое задание «Архивация файлов». С помощью архиватора 7-Zip осуществить архивацию файлов.



Архивация файлов

1. Запустить архиватор 7-Zip.
Выделить файлы и в контекстном меню ввести команду [7-Zip-Добавить в архив...].
2. В окне *Добавить к архиву* (рис. 1.24) задать параметры архивации:
 - имя архивного файла;
 - место хранения архива на диске;
 - формат архива;
 - метод сжатия;
 - размер словаря и слова;
 - размер тома для многотомного архива;
 - метод обновления архива;
 - непрерывность архива.

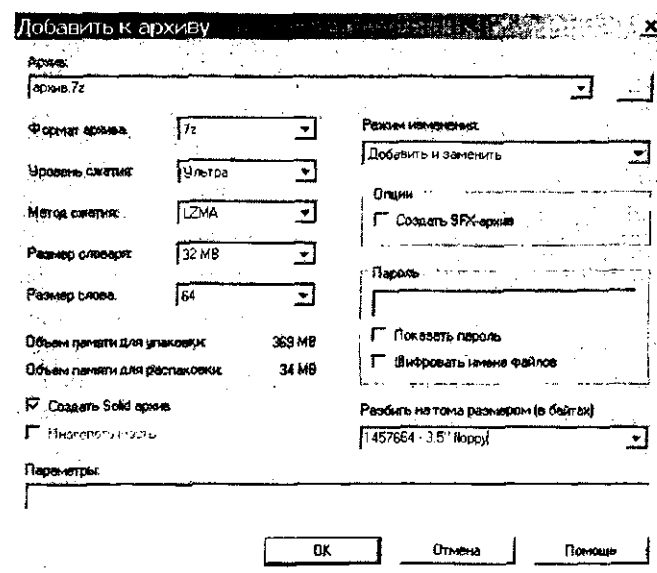


Рис. 1.24. Параметры архивации

3. Для извлечения файлов из архива необходимо выбрать файл архива и в контекстном меню ввести команду [7-Zip-Распаковать].

Контрольные вопросы

1. Как связаны между собой расширение файла и приложение, в котором этот файл был создан?
2. Какие параметры архивации влияют на степень сжатия файлов?

Компьютерный практикум

Windows-CD

- 1.12. В операционной системе Windows с помощью файлового менеджера Total Commander выполнить практическое задание «Расширение и атрибуты файла».



- 1.13. С помощью архиватора 7-Zip выполнить практическое задание «Архивация файлов».



1.4.3. Иерархическая файловая система

На каждом носителе информации (магнитном или оптическом диске, карте флэш-памяти) может храниться большое количество файлов. Для того чтобы можно было найти файл по его имени, на диске создают разделы (тома), а в каждом томе — каталоги.

Информатика-11

Каталог. Каждый том носителя информации содержит корневой каталог (базовый, основной), в котором перечислены хранящиеся на носителе файлы и папки. В простейшем случае, когда файлов мало, каталог представляет собой линейную последовательность записей о файлах.

В разных файловых системах файлы описываются по-разному. Далее рассмотрим файловую систему FAT.

Запись о файле в каталоге содержит (табл. 1.5) имя файла, адрес первого кластера, с которого начинается файл, размер файла, дату и время его создания, а также атрибуты файла (a — архивный, r — только для чтения, h — скрытый, s — системный).

Таблица 1.5. Структура записей в каталоге

Имя файла	Адрес первого кластера	Объем файла (Кбайт)	Дата создания	Время создания	Атрибуты
Файл_1	34	2	14.01.2006	14.29	ar
Файл_2	36	1	20.03.2006	19.45	hs

Если на диске хранятся сотни и тысячи файлов, то для удобства поиска файлы хранятся в многоуровневой иерархической файловой системе, которая имеет древоподобную структуру (рис. 1.25).

Корневая папка содержит вложенные папки 1-го уровня (например, папки Архив и Документы), в свою очередь, каждая из них может содержать вложенные папки 2-го уровня (например, папка Документы содержит папки Web-страницы, Видео, Звук и Изображения) и т. д. Необходимо отметить, что в папках всех уровней могут храниться файлы.

Путь к файлу. Для того чтобы найти файл в иерархии каталогов, необходимо указать путь к файлу. Путь к файлу начинается с логического имени диска, затем записывается последовательность имен вложенных друг в друга папок, в последней из которых содержится нужный файл. Имена диска и папок записываются через разделитель «\». Например, путь к файлу image.bmp можно записать следующим образом:

A:\Документы\Изображения\image.bmp

Восстановление файлов и файловой системы. В процессе работы компьютера случаются сбои («зависание» программ, внезапное отключение питания и др.), в результате происходит неправильное завершение работы приложений и операционной системы, что может приводить к повреждению отдельных кластеров и файлов. Могут появиться сбойные

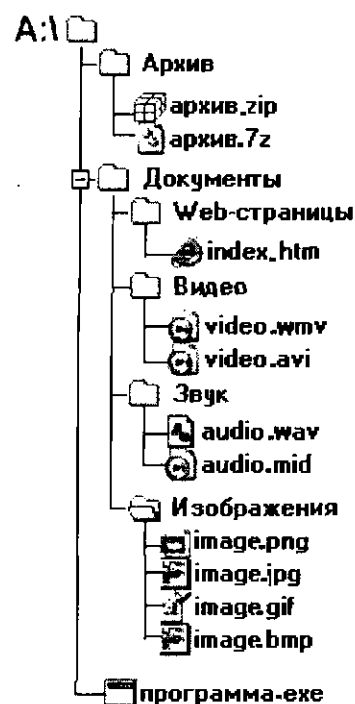


Рис. 1.25. Пример иерархической файловой системы

(нечитаемые) кластеры, в каталогах могут быть изменены имена файлов, а в FAT могут появиться нарушения в цепочках размещения файлов (некоторые цепочки могут быть оборваны, один и тот же кластер может принадлежать различным файлам и др.).

Для восстановления файловой системы используются специальные программы. В операционной системе Windows такой программой является служебная программа Проверка диска, которая автоматически запускается при загрузке Windows после неправильного завершения работы или может быть при необходимости запущена пользователем в произвольный момент.

Практическое задание «Проверка файловой системы диска». В операционной системе Windows осуществить проверку файловой системы диска.



Проверка файловой системы диска

1. В контекстном меню проверяемого диска (например, C:) выбрать пункт *Свойства*.
В появившемся диалоговом окне *Свойства* выбрать вкладку *Сервис* (рис. 1.26) и щелкнуть по кнопке *Выполнить проверку...*

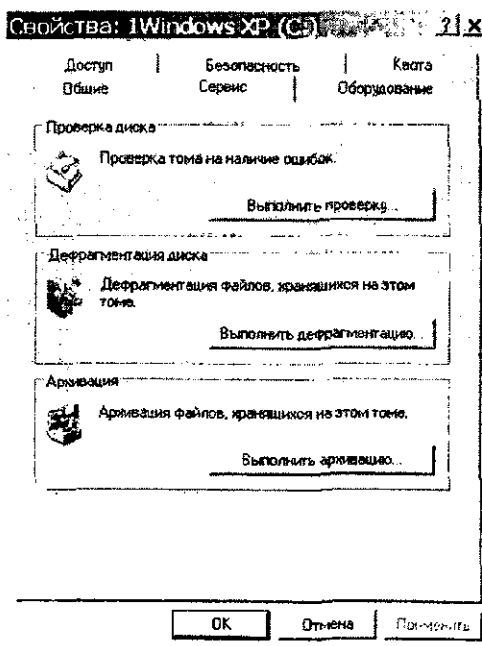


Рис. 1.26. Вкладка *Сервис*

2. В появившемся диалоговом окне *Проверка диска* (рис.1.27) с помощью флажков установить требуемые параметры и щелкнуть по кнопке *Запуск*.

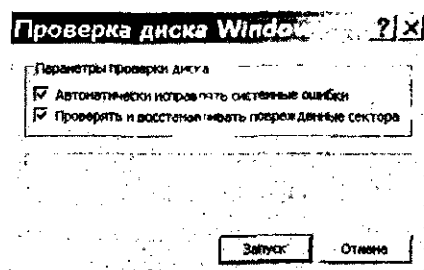


Рис. 1.27. Процесс проверки диска

3. Проверка диска будет выполнена в три этапа. После окончания проверки и восстановления сбойных кластеров и файлов появится окно с информацией о состоянии диска после проверки.

Дефрагментация дисков. Замедление скорости обмена данными с носителем информации может происходить в результате **фрагментации** файлов. Если запись производится на незаполненный диск, то кластеры, принадлежащие одному файлу, записываются подряд. Если диск переполнен, на нем может не быть непрерывной области, достаточной для размещения файла, и файл запишется в виде нескольких фрагментов. Фрагментация файлов (фрагменты файлов хранятся в различных, удаленных друг от друга кластерах) возрастает с течением времени, в процессе удаления одних файлов и записи других.

Так как на диске могут храниться сотни и тысячи файлов в сотнях тысяч кластеров, то фрагментированность файлов будет существенно замедлять доступ к ним (магнитным головкам придется постоянно перемещаться с дорожки на дорожку) и в конечном итоге приводить к преждевременному износу жесткого диска. Рекомендуется периодически проводить дефрагментацию диска, в процессе которой файлы записываются в кластеры, последовательно идущие друг за другом.

Практическое задание «Дефрагментация диска». В операционной системе Windows осуществить дефрагментацию диска.

5.4 Дефрагментация диска

1. Для запуска программы Дефрагментация диска необходимо ввести команду [Стандартные-Служебные-Дефрагментация диска].
2. Диалоговое окно Дефрагментация диска (рис. 1.28) позволяет выбрать диск, нуждающийся в процедуре дефрагментации.

После нажатия кнопки Дефрагментация начнется процесс дефрагментации, который можно визуальнo наблюдать.

Вертикальные полосы красного цвета отображают фрагментированные файлы, синего цвета — нефрагментированные файлы, зеленого цвета — непереключаемые файлы и, наконец, белого цвета — свободные кластеры.

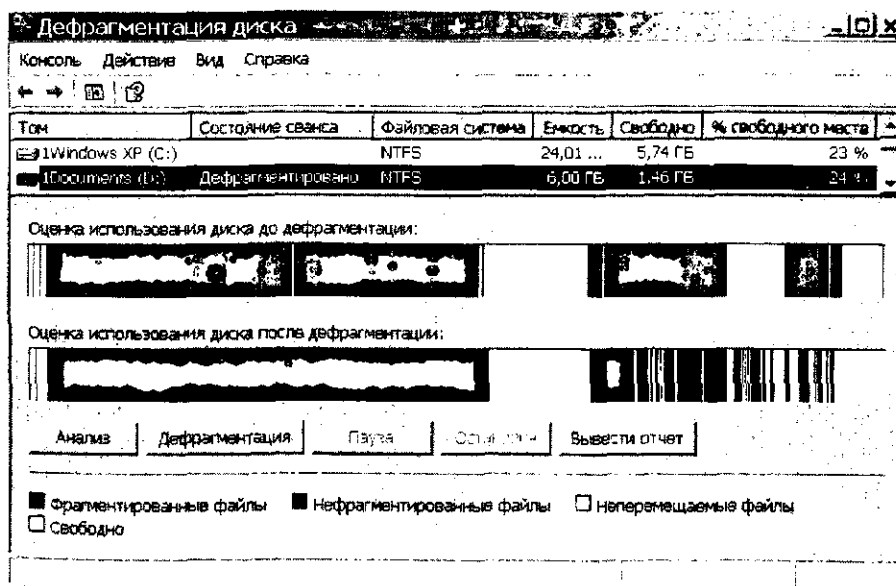


Рис. 1.28. Дефрагментация диска


Контрольные вопросы

1. В чем состоит различие между папками и каталогами?
2. Почему происходит фрагментация файлов?



- 1.2. Записать пути к файлам иерархической файловой системы, изображенной на рис. 1.25.



Windows-CD 

- 1.14. В операционной системе Windows выполнить практическое задание «Проверка файловой системы диска».
- 1.15. В операционной системе Windows выполнить практическое задание «Дефрагментация диска».



1.5. Операционная система

1.5.1. Назначение и состав операционной системы

Операционная система является базовой и необходимой составляющей программного обеспечения компьютера.



Операционная система обеспечивает совместное функционирование всех устройств компьютера и предоставляет пользователю доступ к его ресурсам.

Современные операционные системы имеют сложную структуру, каждый элемент которой выполняет определенные функции по управлению компьютером.

Управление файловой системой. В операционной системе имеются программные модули, управляющие файловой системой.

В процессе работы на компьютере наиболее часто над файлами производятся следующие операции:

- копирование (копия файла помещается в другую папку);
- перемещение (сам файл перемещается в другую папку);
- удаление (запись о файле удаляется из папки);
- переименование (изменяется имя файла).

Командный процессор. В состав операционной системы входит специальная программа — командный процессор, которая запрашивает у пользователя команды и выполняет их.

Пользователь может дать команду запуска программы, выполнения какой-либо операции над файлами (копирование, удаление, переименование), вывода документа на печать и т. д. Операционная система должна эту команду выполнить.

Практическое задание «Копирование файлов». В операционной системе Windows осуществить копирование файлов с использованием графического интерфейса и с помощью командной строки.



Копирование файлов

1. Скопировать файл с использованием графического интерфейса (копированием из папки в папку или с помощью Проводника).
2. Ввести команду [*Программы-Стандартные-Командная строка*]. Появится окно командной строки.
3. В ответ на приглашение системы перейти в каталог C:\Windows с помощью команды CD:
C:\Documents and Setting\НДУ>CD
C:\Windows
4. Получить справку о формате команды copy:
C:\Windows>copy /?
5. Ввести команду (рис. 1.29):
C:\Windows>copy A:\проба.doc D:



Рис. 1.29. Копирование файла в командной строке

Драйверы устройств. К магистрали компьютера подключаются различные устройства (дискетоды, монитор, клавиатура, мышь, принтер и др.). Каждое устройство выполняет определенную функцию (ввод информации, хранение информации, вывод информации), при этом техническая реализация устройств существенно различается.

Драйверы устройств — специальные программы, которые обеспечивают управление работой устройств и согласование информационного обмена с другими устройствами, а также позволяют производить настройку некоторых их параметров. Каждому типу устройств соответствует свой драйвер.


В процессе установки операционная система определяет тип и конкретную модель установленного устройства и подключает необходимые для их функционирования драйверы. При включении компьютера производится загрузка драйверов в оперативную память. Пользователь имеет возможность вручную установить или переустановить драйверы.

Графический интерфейс. Для упрощения работы пользователя в состав современных операционных систем входят программные модули, создающие графический пользовательский интерфейс. В операционных системах с графическим интерфейсом пользователь может вводить команды с помощью диалоговых окон, тогда как в режиме командной строки необходимо вводить команды с помощью клавиатуры.

Диалоговые окна могут включать в себя разнообразные элементы управления:

- вкладки, которые позволяют использовать «страницы» внутри диалогового окна;
- кнопки, которые обеспечивают выполнение тех или иных действий;
- текстовые поля, которые позволяют ввести информацию;
- списки, которые представляют собой наборы предлагаемых на выбор значений;
- переключатели, которые служат для выбора одного из взаимоисключающих вариантов;
- флажки, которые обеспечивают присваивание каким-либо параметрам определенных значений;
- счетчики, которые представляют собой пару стрелок и позволяют увеличивать или уменьшать значение в связанном с ними поле;
- ползунки, которые позволяют плавно изменять значения каких-либо параметров;
- контекстные меню, которые позволяют ознакомиться со свойствами объектов, а также выполнить над ними разрешенные операции.

Такие же элементы управления используются в системах объектно-ориентированного программирования, что позволяет создавать в них приложения с использованием графического интерфейса операционной системы.

 Глава 4. Алгоритмизация и объектно-ориентированное программирование

Сервисные программы. В состав операционной системы входят также сервисные программы, или утилиты. Такие программы позволяют обслуживать диски (проверять, сжимать, дефрагментировать и т. д.), выполнять операции с файлами (архивировать и т. д.), работать в компьютерных сетях и т. д.

Системный реестр Windows. Системный реестр операционной системы Windows является иерархической базой данных, в которой хранится информация о конфигурации Windows. В реестре содержатся сведения об оборудовании системы, установленных программах и параметрах настройки, к которым операционная система постоянно обращается во время работы. В операционную систему Windows входит редактор реестра — программа `regedit.exe`.

Практическое задание «Ознакомление с системным реестром Windows». Ознакомиться с реестром на примере установки в редакторе реестра цвета надписей на кнопках графического интерфейса.



Ознакомление с системным реестром Windows

1. Ввести команду [Выполнить].

В появившемся диалоговом окне *Запуск программы* (рис. 1.30) ввести в список имя программы — редактора реестра и щелкнуть по кнопке *ОК*.

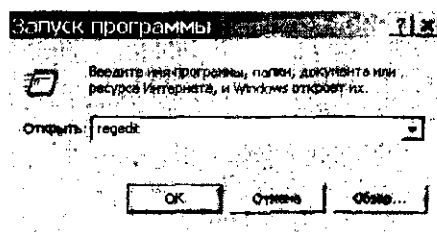


Рис. 1.30. Запуск редактора реестра

2. В появившемся диалоговом окне *Редактор реестра* (рис. 1.31) в иерархическом дереве реестра открыть пункт

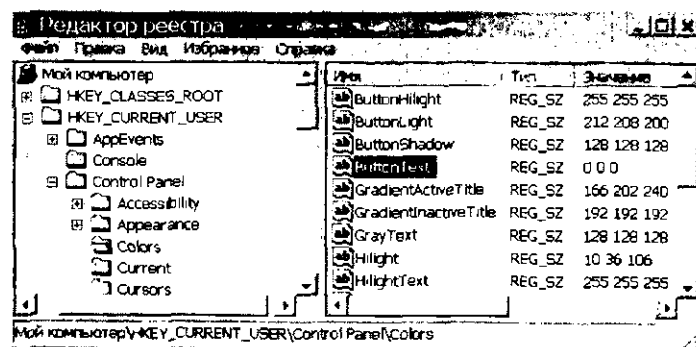


Рис. 1.31. Редактор реестра

HKEY_CURRENT_USER, а затем открыть вложенный пункт *Control Panel* и в нем активизировать пункт *Colors*. В правой части окна редактора реестра в списке параметров выделить параметр *ButtonText*.

3. В диалоговом окне *Изменение строкового параметра* (рис. 1.32) ввести новое значение параметра *ButtonText* (например, *255 0 0* — красный цвет надписей на кнопках).

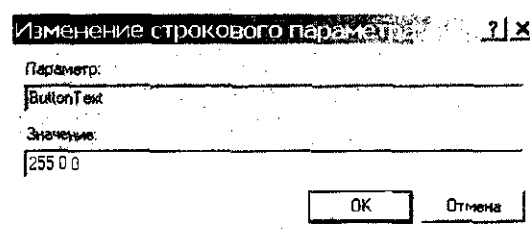


Рис. 1.32. Изменение значения параметра

4. Для того чтобы изменение вступило в силу, выйти из редактора реестра командой [*Файл-Выход*] и перезагрузить компьютер.

Категорически не рекомендуется изменять параметры реестра без четкого понимания производимых действий, так как ошибка при изменении реестра может серьезно повредить системе.

Если повреждена информация об устройствах, то соответствующий раздел реестра можно исправить или восстановить в том виде, который он имел в момент последнего удачного запуска компьютера. Для этого необходимо перезагрузить компьютер, в процессе загрузки нажать клавишу {F8} и выбрать вариант загрузки *Загрузка последней удачной конфигурации*. Современные версии операционной системы Windows содержат средство восстановления системы, которое может восстановить системный реестр, существовавший на определенную дату, если ранее была сохранена резервная копия этих данных.


Справочная система. Для удобства пользователя в состав операционной системы обычно входит также справочная система. Справочная система позволяет оперативно получить необходимую информацию как о функционировании операционной системы в целом, так и о работе ее отдельных модулей.

Контрольные вопросы

1. Каковы основные компоненты операционной системы и каково их назначение?

Компьютерный практикум




Windows-CD 

- 1.16. В операционной системе Windows выполнить практическое задание «Копирование файлов».
- 1.17. В операционной системе Windows выполнить практическое задание «Ознакомление с системным реестром Windows».



1.5.2. Загрузка операционной системы

Самотестирование компьютера. После включения компьютера или перезагрузки операционной системы процессор начинает считывать и выполнять микропрограммы тестирования компьютера и первого этапа загрузки операционной системы, которые хранятся в микросхеме **BIOS**.

-  Включение компьютера осуществляется включением питания (кнопка *Power* на системном блоке). Перезагрузка операционной системы производится нажатием кнопки *Reset* на системном блоке компьютера или одновременным нажатием комбинации клавиш $\{Ctrl\} + \{Alt\} + \{Del\}$ на клавиатуре.

Прежде всего начинает выполняться программа тестирования **POST**, которая проверяет работоспособность основных устройств компьютера: процессора, видеоадаптера, оперативной памяти, дисководов, контроллеров жестких дисков и клавиатуры. В случае обнаружения неисправностей выдаются диагностические сообщения в виде различных последовательностей коротких и длинных звуковых сигналов (например, в AWARD BIOS: 1 длинный и 3 коротких сигнала — не подключен монитор, 5 коротких — ошибка процессора и т. д.) или в виде текстовых сообщений. После успешной инициализации видеоплаты краткие диагностические сообщения выводятся на экран монитора.

BIOS Setup. Пользователь может установить новые параметры конфигурации компьютера и запомнить их в специальной микросхеме памяти, которая при выключенном компьютере питается от батарейки, установленной на системной плате. Для этого в процессе выполнения самотестирования обычно необходимо нажать клавишу *{Del}*. Загрузится системная утилита BIOS Setup, имеющая интерфейс в виде системы иерархических меню. В случае выхода из строя батарейки конфигурационные параметры теряются, и компьютер перестает нормально загружаться.

Загрузка операционной системы (рис. 1.33). Файлы операционной системы хранятся во внешней, долговременной памяти (на магнитных, оптических или флэш-дисках). Однако программы могут выполняться, только если они находятся в оперативной памяти, поэтому файлы операционной системы необходимо загрузить в оперативную память. Разрешение этого противоречия состоит в последовательной, поэтапной загрузке операционной системы.



Диск, на котором находятся файлы операционной системы и с которого производится ее загрузка, называется **системным**.

После проведения самотестирования специальная программа, содержащаяся в BIOS, начинает поиск загрузчика операционной системы. Современные версии BIOS позволяют загружать операционную систему не только с магнитных и оптических дисков, но и с USB флэш-дисков.

Если диск системный и программа-загрузчик оказывается на месте, то она загружается в оперативную память и ей передается управление работой компьютера. Программа ищет файлы операционной системы на системном диске и загружает их в оперативную память в качестве программных модулей.

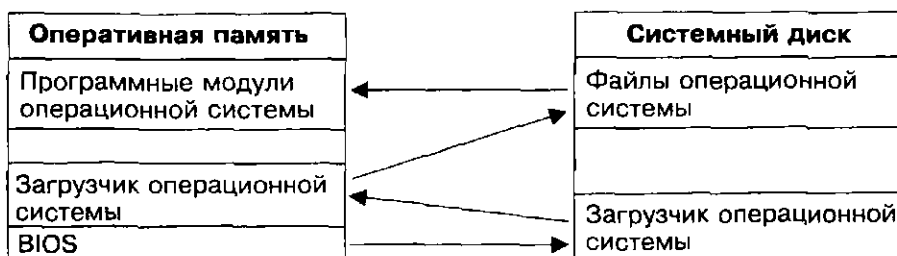


Рис. 1.33. Процесс загрузки операционной системы

Если системные диски в компьютере отсутствуют, на экране монитора появляется сообщение «Non system disk», и компьютер «зависает», т. е. загрузка операционной системы прекращается и компьютер остается неработоспособным.

В процессе загрузки можно выбрать в том числе вариант загрузки без графического интерфейса (для вывода меню вариантов загрузки нужно нажать клавишу {F8}). В случае использования интерфейса командной строки на экране появляется приглашение системы к вводу команд. Приглашение представляет собой последовательность символов, сообщающую о текущем диске и папке. Например, если загрузка операционной системы была произведена с диска C:, а операционная система была установлена в папку C:\WINDOWS, то появится приглашение:

```
C:\WINDOWS>
```

В случае загрузки графического интерфейса операционной системы команды могут вводиться с использованием элементов управления.

Контрольные вопросы


1. Каковы основные этапы загрузки операционной системы?

1.6. Защита информации от вредоносных программ

1.6.1. Вредоносные программы и антивирусные программы

Типы вредоносных программ. Вредоносными программами являются программы, наносящие вред данным и программам, хранящимся на компьютере. Основными типами вредоносных программ являются:

- компьютерные вирусы;
- сетевые черви;
- троянские программы;
- программы показа рекламы (от англ. adware) и программы-шпионы, занимающиеся сбором персональной информации о компьютере и пользователе (от англ. spyware);
- хакерские утилиты.

 Прообразом вредоносных программ была компьютерная игра «Дарвин», созданная в 1961 году в научно-исследовательских целях. Несколько компьютерных программ, названных «организмами», загружались в память ЭВМ, причем организмы, созданные одним игроком (т. е. принадлежащие к одному виду), должны были уничтожать представителей другого вида и захватывать оперативную память.

За создание, использование и распространение вредоносных программ в России и большинстве стран предусмотрена уголовная ответственность.

Антивирусные программы. Принцип работы антивирусных программ основан на проверке файлов, загрузочных секторов дисков и оперативной памяти и поиске в них известных и новых вирусов.

Для поиска известных вирусов используются сигнатуры, т. е. некоторые постоянные последовательности двоичного кода, специфичные для этого конкретного вируса. Если антивирусная программа обнаружит такую последовательность в каком-либо файле, то файл считается зараженным вирусом и подлежит лечению.

Для поиска новых вирусов используются алгоритмы эвристического сканирования, т. е. анализ последовательности команд в проверяемом объекте. Если «подозрительная» последовательность команд обнаруживается, то антивирусная программа выдает сообщение о возможном заражении объекта.

Большинство антивирусных программ сочетает в себе функции постоянной защиты (антивирусный монитор) и функции защиты по требованию пользователя (антивирусный сканер).

Антивирусный монитор запускается автоматически при старте операционной системы и работает в качестве фонового системного процесса, проверяя на вредоносность совершаемые другими программами действия. Основная задача антивирусного монитора состоит в обеспечении максимальной защиты от вредоносных программ при минимальном замедлении работы компьютера.

Антивирусный сканер запускается по заранее выбранному расписанию или в произвольный момент пользователем. Антивирусный сканер производит поиск вредоносных программ в оперативной памяти, а также на жестких и сетевых дисках компьютера.

К недостаткам антивирусных программ можно отнести большие размеры используемых ими антивирусных баз данных, которые должны содержать информацию о максимально возможном количестве вирусов (в настоящее время десятках тысяч), что, в свою очередь, приводит к относительно небольшой скорости поиска вирусов.

Признаки заражения компьютера. Есть ряд признаков, свидетельствующих о проникновении на компьютер вредоносных программ:

- вывод на экран непредусмотренных сообщений или изображений;
- подача непредусмотренных звуковых сигналов;
- неожиданное открытие и закрытие лотка CD/DVD дисковода;
- произвольный запуск на компьютере каких-либо программ;
- частые «зависания» и сбои в работе компьютера;
- медленная работа компьютера при запуске программ;
- исчезновение или изменение файлов и папок;
- частое обращение к жесткому диску (часто мигает лампочка на системном блоке);
- «зависание» или неожиданное поведение браузера (например, окно программы невозможно закрыть).

Кроме того, есть некоторые характерные признаки поражения сетевым вирусом через электронную почту:

- друзья или знакомые говорят о полученных от вас сообщениях, которые вы не отправляли;
- в вашем почтовом ящике находится большое количество сообщений без обратного адреса и заголовка.

Действия при наличии признаков заражения компьютера. Прежде чем предпринимать какие-либо действия, необходимо сохранить результаты работы на внешнем носителе (дискете, CD- или DVD-диске, флэш-карте и пр.). Далее необходимо:

- отключить компьютер от локальной сети и Интернета, если он к ним был подключен;
- если симптом заражения состоит в том, что невозможно загрузиться с жесткого диска компьютера (компьютер выдает ошибку, когда вы его включаете), попробовать загрузиться в режиме защиты от сбоев или с диска аварийной загрузки Windows;
- запустить антивирусную программу.

Контрольные вопросы

1. К каким последствиям может привести заражение компьютерными вирусами?
2. Какие существуют признаки заражения компьютера вирусом?
3. Что необходимо сделать в первую очередь в случае заражения компьютера вирусом?

1.6.2. Компьютерные вирусы и защита от них

Обязательным свойством компьютерного вируса является способность к размножению (самокопированию). Вирусы могут также незаметно для пользователя внедряться в исполняемые файлы, загрузочные секторы дисков и документы. Название «вирус» по отношению к компьютерным программам пришло из биологии именно по признаку способности к саморазмножению.

После заражения компьютера вирус может начать выполнение вредоносных действий и распространение своих копий, а также заставить компьютер выполнять какие-либо действия. Активизация вируса может быть связана с различными событиями (наступлением определенной даты или дня недели, запуском программы, открытием документа и т. д.).



Компьютерные вирусы являются вредоносными программами, которые могут «размножаться» и скрытно внедрять свои копии в исполнимые файлы, загрузочные секторы дисков и документы. Активизация компьютерного вируса может вызывать уничтожение программ и данных.

Разнообразны последствия действия вирусов. По величине вредных воздействий вирусы можно разделить на:

- неопасные, влияние которых ограничивается уменьшением свободной памяти на диске, графическими, звуковыми и другими внешними эффектами;
- опасные, которые могут привести к сбоям и «зависаниям» при работе компьютера;
- очень опасные, активизация которых может привести к потере программ и данных (изменению или удалению файлов и каталогов), форматированию винчестера и т. д.

В настоящее время известно несколько десятков тысяч вирусов, заражающих компьютеры различных операционных

систем. По способу сохранения и исполнения своего кода вирусы можно разделить на загрузочные, файловые, макро-вирусы и скрипт-вирусы.

Загрузочные вирусы. Загрузочные вирусы заражают загрузочный сектор гибкого или жесткого диска. Принцип действия загрузочных вирусов основан на алгоритмах запуска операционной системы при включении или перезагрузке компьютера. После необходимых тестов установленного оборудования программа системной загрузки считывает первый физический сектор загрузочного диска (гибкого, жесткого, оптического или флэш-диска в зависимости от параметров, установленных в BIOS Setup) и передает на него управление.

При заражении дисков загрузочные вирусы «подставляют» свой код вместо программы, получающей управление при загрузке системы, и отдают управление не оригинальному коду загрузчика, а коду вируса. При инфицировании диска вирус в большинстве случаев переносит оригинальный загрузочный сектор в какой-либо другой сектор диска (например, в первый свободный).



Первая эпидемия загрузочного компьютерного вируса произошла в 1986 году, когда вирус «Brain» «заражал» загрузочный сектор дискет для персональных компьютеров. Вирус «Brain» являлся также и первым вирусом-невидимкой, так как при попытке обнаружения зараженного загрузочного сектора вирус незаметно «подставлял» его незараженный оригинал.

Профилактическая защита от таких вирусов состоит в отказе от загрузки операционной системы с гибких дисков и установке в BIOS вашего компьютера защиты загрузочного сектора от изменений. С помощью программы BIOS Setup можно провести настройку BIOS таким образом, что будет запрещена (заблокирована) любая запись в загрузочный сектор диска и компьютер будет защищен от заражения загрузочными вирусами.

Файловые вирусы. Файловые вирусы различными способами внедряются в исполнимые файлы (командные файлы *.bat, программы *.exe, системные файлы *.com и *.sys, программные библиотеки *.dll и др.) и обычно активизируются при их запуске. После запуска зараженного файла вирус находится в оперативной памяти компьютера и является активным (т. е. может заражать другие файлы) вплоть до момента выключения компьютера или перезагрузки операционной системы.

По способу заражения файловые вирусы разделяют на:

- перезаписывающие вирусы, которые записывают свой код вместо кода программы, не изменяя названия исполнимого файла. При запуске программы выполняется код вируса, а не сама программа;
- вирусы-компаньоны, которые, как и перезаписывающие вирусы, создают свою копию на месте заражаемой программы, но в отличие от перезаписываемых не уничтожают оригинальный файл, а переименовывают или перемещают его. При запуске программы вначале выполняется код вируса, а затем управление передается оригинальной программе;
- паразитические вирусы — это файловые вирусы, изменяющие содержимое файла, добавляя в него свой код. Код может внедряться в начало, середину или конец программы и выполняется перед, вместе или после программы. При этом зараженная программа сохраняет полную или частичную работоспособность.



В 1999 году произошла эпидемия очень опасного файлового вируса Win95.CIH, названного «Чернобыль» из-за даты активации 26 апреля. Вирус уничтожил данные на жестком диске, а на системных платах стирал содержимое BIOS, что приводило к необходимости их замены.

Практически все загрузочные и файловые вирусы резидентны, т. е. они находятся в оперативной памяти компьютера, и в процессе работы пользователя могут осуществлять опасные действия (стирать данные на дисках, изменять названия и другие атрибуты файлов и т. д.). Лечение от резидентных вирусов затруднено, так как даже после удаления зараженных файлов с дисков вирус остается в оперативной памяти и возможно повторное заражение файлов.

Профилактическая защита от файловых вирусов состоит в том, что не рекомендуется запускать на исполнение файлы, полученные из сомнительного источника и предварительно не проверенные антивирусными программами.

Макро-вирусы. Наибольшее распространение получили макро-вирусы для интегрированного офисного приложения Microsoft Office (Word, Excel, PowerPoint и Access). Макро-вирусы фактически являются макрокомандами (макросами) на встроенном языке программирования Visual Basic for Applications (VBA), которые помещаются в документ.

При работе с документом пользователь выполняет различные действия: открывает документ, сохраняет, печатает, за-

крывает и т. д. При этом приложение ищет и выполняет соответствующие стандартные макросы. Макро-вирусы содержат стандартные макросы, вызываются вместо них и заражают каждый открываемый или сохраняемый документ. Вредные действия макро-вирусов реализуются с помощью встроенных макросов (вставки текстов, запрета выполнения команд меню приложения и т. д.).

Макро-вирусы являются **ограниченно резидентными**, т. е. они находятся в оперативной памяти и заражают документы, пока открыто приложение. Кроме того, макро-вирусы заражают шаблоны документов и поэтому активизируются уже при запуске зараженного приложения.



В августе 1995 года началась эпидемия первого макро-вируса «Concept» для текстового процессора Microsoft Word. Макро-вирус «Concept» до сих пор имеет широкое распространение, и на сегодняшний момент известно около 100 его модификаций.

Профилактическая защита от макро-вирусов состоит в предотвращении запуска вируса. При открытии документа в приложениях Microsoft Office сообщается о присутствии в них макросов (потенциальных вирусов) и предлагается запретить их загрузку. Выбор запрета на загрузку макросов надежно защитит ваш компьютер от заражения макро-вирусами, однако отключит и полезные макросы, содержащиеся в документе.

Скрипт-вирусы. Особой разновидностью вирусов являются активные элементы (программы) на языках JavaScript или VBScript, которые могут содержаться в файлах Web-страниц. Заражение локального компьютера происходит при их передаче по Всемирной паутине с серверов Интернета в браузер локального компьютера.



В ноябре 1998 года появился первый скрипт-вирус VBScript.Rabbit, заражающий скрипты Web-страниц, а через полтора года, в мае 2000 года грянула глобальная эпидемия скрипт-вируса «LoveLetter». Сейчас этот тип вирусов прочно удерживает первое место в списке наиболее распространенных и опасных вирусов.

Профилактическая защита от скрипт-вирусов состоит в том, что в браузере можно запретить получение активных элементов на локальный компьютер.

Практическое задание «Защита от компьютерных вирусов». С помощью антивирусной программы (например, AntiVir Personal Edition) проверить компьютер на наличие вирусов и при их обнаружении вылечить или удалить зараженные файлы.



Защита от компьютерных вирусов

Прежде всего, необходимо через Интернет обновить саму антивирусную программу и вирусную базу данных.

1. Соединиться с Интернетом.

На панели задач в контекстном меню значка антивирусной программы AntiVir Personal Edition выбрать пункт *Start Update*.

В появившемся диалоговом окне (рис. 1.34) будет отображаться процесс обновления антивирусной программы и вирусной базы данных.

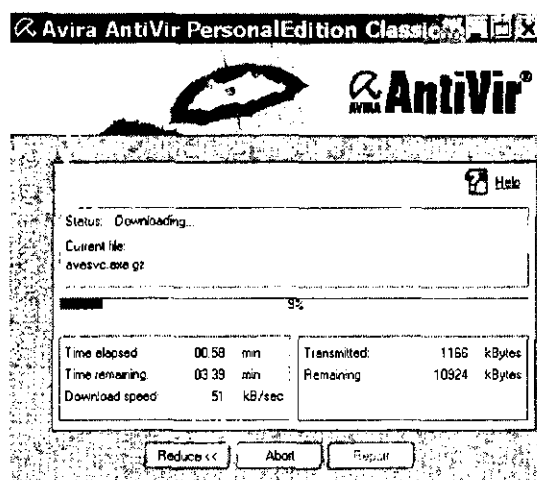


Рис. 1.34. Обновление антивирусной программы

Настроим параметры антивирусного монитора (Guard) и сканера (Scanner).

2. В контекстном меню значка антивирусной программы выбрать пункт *Configure AntiVir*.

В появившемся диалоговом окне (рис. 1.35) выбрать экспертный режим, установив флажок *Expert mode*.

В левой части окна в иерархическом меню выбрать пункт *Guard*, а в правой установить необходимые параметры антивирусного монитора.

Для выбора типов проверяемых файлов щелкнуть по кнопке *File extensions* и в появившемся раскрывающемся списке добавить или удалить типы файлов.

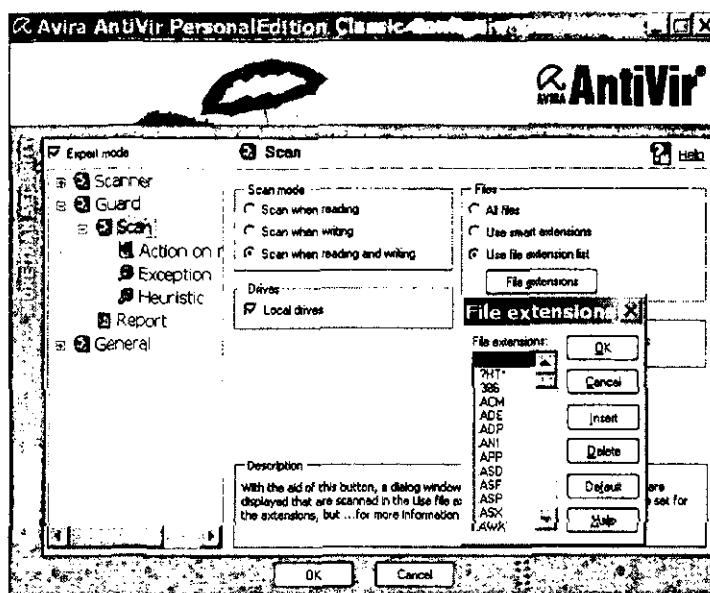


Рис. 1.35. Настройка параметров антивирусного монитора (Guard)

3. В иерархическом меню выбрать пункт *Scan* (рис. 1.36), а в правой части окна установить необходимые параметры антивирусного сканера.

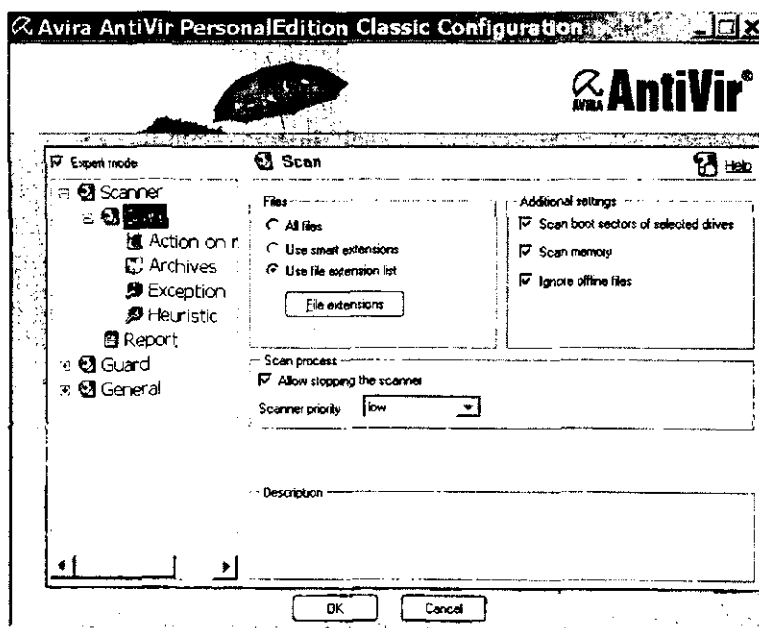


Рис. 1.36. Настройка параметров антивирусного сканера (Scanner)

Проведем проверку на вирусы выбранных дисков.

4. В контекстном меню значка антивирусной программы выбрать пункт *Start AntiVir*.

В появившемся диалоговом окне (рис. 1.37) выбрать вкладку *Scanner*.

В иерархической файловой системе отметить флажком диски, выбранные для проверки.

Щелкнуть по кнопке *Start scan*.

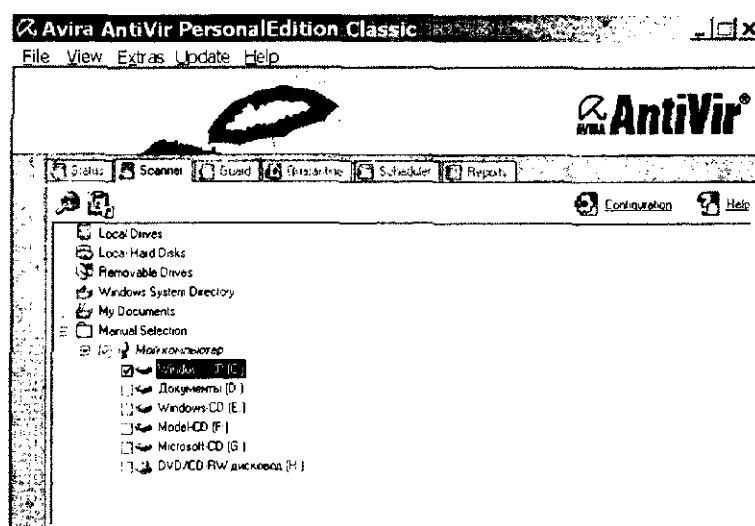


Рис. 1.37. Проверка на вирусы выбранных дисков


5. Появится диалоговое окно, в котором будет отображаться процесс антивирусной проверки файлов на выбранном диске.


Для получения результатов проверки щелкнуть по кнопке *Report*.

Контрольные вопросы

1. Какие типы компьютерных вирусов существуют, чем они отличаются друг от друга и какова должна быть профилактика заражения?
2. Почему даже чистая отформатированная дискета может стать источником заражения вирусом?

Компьютерный практикум

Windows-CD 

- 1.18. С помощью антивирусной программы (например, AntiVir Personal Edition) выполнить практическое задание «Защита от компьютерных вирусов». 
- 1.19. С использованием Вирусной энциклопедии ознакомиться с классификацией вирусов и методами антивирусной защиты.

<http://www.viruslist.com>

1.6.3. Сетевые черви и защита от них

К сетевым червям (от англ. worm) относятся вредоносные программы, распространяющие свои копии по локальным и/или глобальным сетям. Для своего распространения сетевые черви используют разнообразные сервисы глобальных и локальных компьютерных сетей: Всемирную паутину, электронную почту, интерактивное общение, файлообменные сети и т. д.

Основным признаком, по которому типы червей различаются между собой, является способ распространения червя — каким способом он передает свою копию на удаленные компьютеры. Однако многие сетевые черви используют более одного способа распространения своих копий по компьютерам локальных и глобальных сетей.



Сетевые черви являются вредоносными программами, которые проникают на компьютер, используя сервисы компьютерных сетей. Активизация сетевого червя может вызывать уничтожение программ и данных, а также похищение персональных данных пользователя.

Почтовые черви. Почтовые черви для своего распространения используют электронную почту. Червь отправляет либо свою копию в виде вложения в электронное письмо, либо ссылку на свой файл, расположенный на каком-либо сетевом ресурсе. В первом случае код червя активизируется при открытии (запуске) зараженного вложения, во втором — при открытии ссылки на зараженный файл. В обоих случаях эффект одинаков — активизируется код червя.

Лавинообразная цепная реакция распространения почтового червя базируется на том, что червь после заражения компьютера начинает рассылать себя по всем адресам электронной почты, которые имеются в адресной книге пользователя.

Профилактическая защита от почтовых червей состоит в том, что не рекомендуется открывать вложенные в почтовые сообщения файлы, полученные из сомнительных источников.

Черви, использующие «уязвимости» программного обеспечения. Червь ищет в сети компьютеры, на которых используются операционная система и приложения, содержащие критические уязвимости. Для заражения уязвимых компьютеров червь посылает специально оформленный сетевой пакет или запрос, в результате чего код (или часть кода) червя проникает на компьютер-жертву. Если сетевой пакет содержит только часть кода червя, он затем скачивает основной файл и запускает его на исполнение на зараженном компьютере.

Профилактическая защита от таких червей состоит в том, что рекомендуется своевременно скачивать из Интернета и устанавливать обновления системы безопасности операционной системы и приложений.

Черви, использующие файлообменные сети. Механизм работы подобных червей достаточно прост — для внедрения в файлообменную сеть червя достаточно скопировать себя в папку обмена файлами на одном из компьютеров. Всю остальную работу по распространению червя файлообменная сеть берет на себя — при поиске файлов в сети она сообщит удаленным пользователям о данном файле-черве и предоставит его для скачивания.



В сентябре 2001 года началась стремительное «расползание» сетевого червя «Nimda», который мог атаковать компьютеры сразу несколькими способами: через сообщения электронной почты, через открытые ресурсы локальных сетей, а также используя уязвимости в системе безопасности операционной системы серверов Интернета.

Сетевые черви кроме вредоносных действий, которыми обладают и классические компьютерные вирусы, могут выполнять шпионскую функцию троянских программ.



1.6.4. Троянские программы и защита от них

Практическое задание «Защита от сетевых червей». С помощью антивирусной программы (например, avast!) проверить компьютер на заражение сетевыми червями и при их обнаружении вылечить или удалить зараженные файлы.



Защита от сетевых червей

Прежде всего необходимо через Интернет обновить саму антивирусную программу и вирусную базу данных.

1. Соединиться с Интернетом.

На панели задач в контекстном меню значка антивирусной программы avast! выбрать сначала пункт [Обновление-Обновление программы], а затем пункт [Обновление-Обновление базы данных iAVS].

В появляющихся диалоговых окнах будет отображаться процесс обновления антивирусной программы и вирусной базы данных.

Настроим параметры резидентного антивирусного монитора (авторы программы называют его сканером доступа). Можно отдельно настроить параметры сканера доступа на наличие сетевых червей в электронной почте, Web-страницах, мгновенных сообщениях интерактивного общения, локальных и файлообменных сетях (авторы программы называют эти сервисы компьютерных сетей провайдерами).

2. В контекстном меню значка антивирусной программы выбрать пункт *Настройка сканера доступа*.

В левой части появившегося диалогового окна (рис. 1.38) выбрать провайдера, например *Электронная почта*.

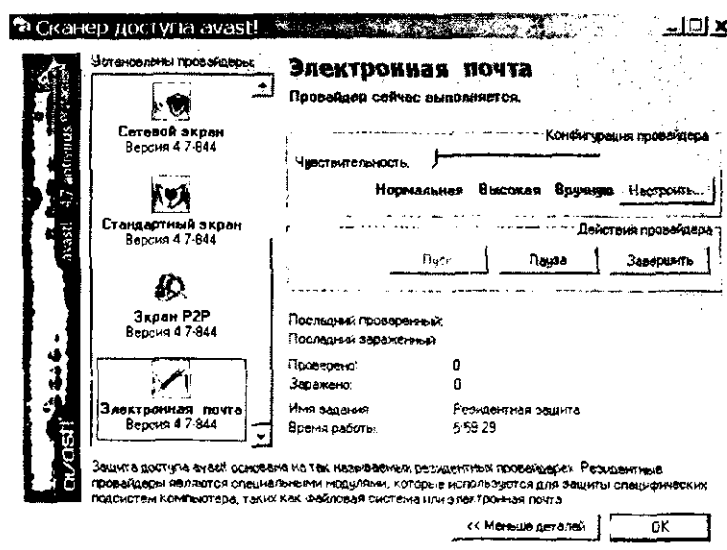


Рис. 1.38. Настройка параметров антивирусного монитора (сканера доступа)

В правой части окна установить необходимые параметры проверки входящей почты.

Проверим входящую почту на наличие почтовых червей.

3. Запустить почтовую программу и начать получение входящих сообщений.

В случае обнаружения в сообщении почтового червя раздастся голосовое предупреждение и будет выведено диалоговое окно *Обнаружен вирус!* (рис. 1.39).

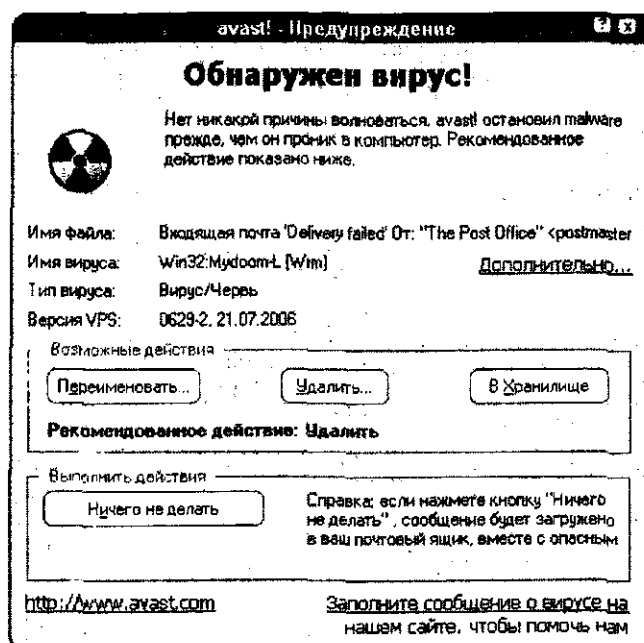


Рис. 1.39. Обнаружение почтового червя

В окне будет указано, какое сообщение инфицировано, имя вируса (например, *Win32.Mydoom-L*) и тип вируса (например, *Вирус/Червь*).

Пользователь может выбрать одно из следующих действий щелчком по соответствующей кнопке: *Переименовать*, *Удалить*, *В хранилище*, *Ничего не делать*.

Проведем проверку на вирусы выбранных дисков.

4. В контекстном меню значка антивирусной программы выбрать пункт *Запустить антивирус avast!*.

В появившемся диалоговом окне с помощью кнопок справа выбрать диски для сканирования. Щелкнуть по кнопке *Запустить*.

В диалоговом окне будет отображаться процесс антивирусной проверки файлов на выбранном диске (рис. 1.40).

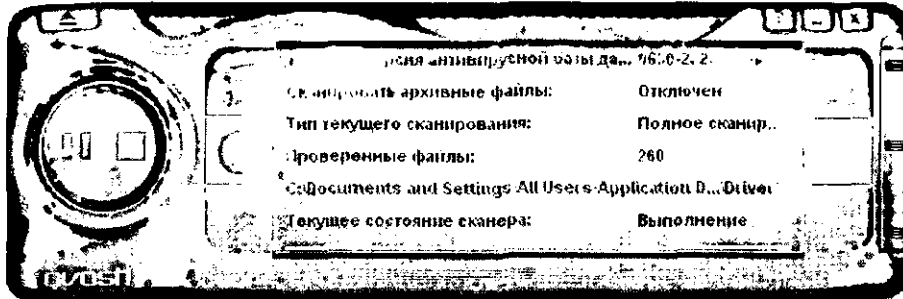



Рис. 1.40. Проверка на вирусы выбранных дисков

5. Для ознакомления с результатами в контекстном меню значка антивирусной программы выбрать пункт *Просмотр журнала avast!*.

Контрольные вопросы

1. Какие типы сетевых червей существуют, чем они отличаются друг от друга и какова должна быть профилактика заражения?

Компьютерный практикум

Windows-CD 

- 1.20. С помощью антивирусной программы (например, avast!) выполнить практическое задание «Защита от сетевых червей».



1.6.4. Троянские программы и защита от них

Троянские программы осуществляют не санкционированные пользователем действия по сбору и передаче информации злоумышленнику, а также ее разрушение или злонамеренную модификацию. Кроме того, троянские программы могут вызывать нарушение работоспособности компьютера или незаметно для пользователя использовать ресурсы компьютера в целях злоумышленника.

Троянские программы обычно проникают на компьютер как сетевые черви, а различаются между собой по тем действиям, которые они производят на зараженном компьютере.



Троянская программа, троянец (от англ. trojan) — вредоносная программа, которая выполняет несанкционированную пользователем передачу управления компьютером удаленному пользователю, а также действия по удалению, модификации, сбору и пересылке информации третьим лицам.

Троянские утилиты удаленного администрирования. Троянские программы этого класса являются утилитами удаленного администрирования компьютеров в сети. Утилиты скрытого управления позволяют принимать или отсылать файлы, запускать и уничтожать их, выводить сообщения, стирать информацию, перезагружать компьютер и т. д.

При запуске троянец устанавливает себя в системе и затем следит за ней, при этом пользователю не выдается никаких сообщений о действиях троянской программы в системе. В результате «пользователь» этой троянской программы может и не знать о ее присутствии в системе, в то время как его компьютер открыт для удаленного управления.

Троянские программы данного типа являются одним из самых опасных видов вредоносного программного обеспечения, поскольку в них заложена возможность самых разнообразных злоумышленных действий, в том числе они могут быть использованы для обнаружения и передачи конфиденциальной информации.



В 2003 году широкое распространение получила троянская программа Backdoor.Win32.BO, которая осуществляет следующие действия:

- высылает имена компьютера, пользователя и информацию о системе: тип процессора, размер памяти, версию системы, информацию об установленных устройствах;
- посылает/принимает, уничтожает, копирует, переименовывает, исполняет любой файл;
- отключает пользователя от сети;
- «завешивает» компьютер;
- читает или модифицирует системный реестр.

Троянские программы, ворующие информацию. Такие троянские программы воруют различную информацию с зараженного компьютера. При запуске они ищут файлы, хранящие конфиденциальную информацию о пользователе (банковские реквизиты, пароли доступа к Интернету и др.) и отправляют ее по указанному в коде троянца электронному адресу или адресам.

Троянцы данного типа также сообщают информацию о зараженном компьютере (размер памяти и дискового пространства, версию операционной системы, IP-адрес и т. п.). Некоторые троянцы воруют регистрационную информацию к программному обеспечению.

Троянские программы — инсталляторы вредоносных программ. Троянские программы этого класса скрытно инсталлируют другие вредоносные программы и используются для «подсовывания» на компьютер-жертву вирусов или других троянских программ. Загруженные без ведома пользователя из Интернета программы затем либо запускаются на выполнение, либо включаются троянцем в автозагрузку операционной системы.

Троянские программы — шпионы. Данные троянцы осуществляют электронный шпионаж за пользователем зараженного компьютера: вводимая с клавиатуры информация, снимки экрана, список активных приложений и действия пользователя с ними сохраняются в каком-либо файле на диске и периодически отправляются злоумышленнику.

Троянские программы этого типа часто используются для кражи информации пользователей различных систем онлайн-новых платежей и банковских систем.

Троянские программы часто изменяют записи системного реестра операционной системы, поэтому для их удаления необходимо в том числе восстановление системного реестра.

Практическое задание «Защита от троянских программ». С помощью программы восстановления системы (например, CCleaner) исправить ошибки системного реестра.



Защита от троянских программ

Исправим записи системного реестра, в которые троянские программы могли внести изменения.

1. Запустить программу восстановления системы CCleaner. В левой части окна приложения (рис. 1.41) щелкнуть по кнопке *Неполадки*.

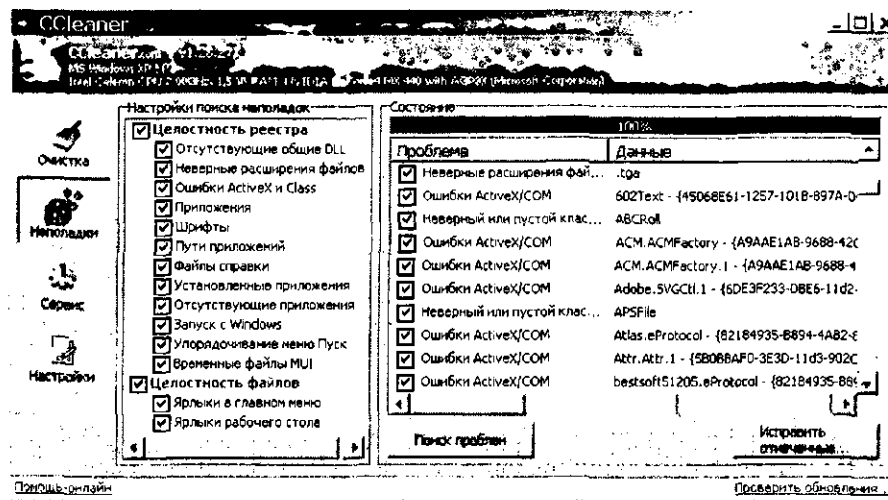


Рис. 1.41. Исправление записей системного реестра

В списке *Настройка поиска неполадок* отметить флажками нужные пункты.

Щелкнуть по кнопке *Поиск проблем*, в правой части окна будут показаны проблемы реестра и соответствующие записи реестра.

Отметить флажками нужные записи и щелкнуть по кнопке *Исправить отмеченные...*


Произойдет исправление выбранных записей системного реестра.

Контрольные вопросы

1. Какие типы троянских программ существуют и каковы их вредоносные действия?

Компьютерный практикум



Windows-CD 

- 1.21. С помощью программы восстановления системы (например, CCleaner) выполнить практическое задание «Защита от троянских программ».



1.6.5. Рекламные и шпионские программы и защита от них

Рекламные программы. Рекламные программы (от англ. **Adware: Advertisement** — реклама и **Software** — программное обеспечение) встраивают рекламу в основную полезную программу. Часто рекламные программы входят в состав официально поставляемых условно бесплатных версий программного обеспечения.

Реклама демонстрируется пользователю в процессе работы основной программы в виде графических баннеров или бегущей строки. Обычно после покупки и/или регистрации основной программы рекламная вставка удаляется и показ рекламы прекращается.

Шпионские программы. Шпионские программы (от англ. **Spyware: Spy** — шпион и **Software** — программное обеспечение) скрытно собирают различную информацию о пользователе компьютера и затем отправляют ее злоумышленнику.

Эти программы иногда проникают на компьютер под видом рекламных программ и не имеют возможности деинсталляции пользователем без нарушения функционирования использующей их программы. Иногда шпионские программы обнаруживаются в распространенных программных продуктах известных на рынке производителей.



В марте 2005 года под видом поисковой панели для браузера Internet Explorer начала распространяться рекламно-шпионская программа «mwsbar». Программа регистрирует себя в системном реестре и добавляет в автозагрузку, что приводит к изменению настроек браузера и перенаправлению результатов поиска в Интернете на сайт злоумышленника.

Практическое задание «Защита от рекламных и шпионских программ». С помощью программы удаления рекламных и шпионских программ (например, Ad-Adware) очистить компьютер от adware и spyware программ.



Защита от рекламных и шпионских программ

1. Запустить программу Ad-Adware, которая удаляет рекламные и шпионские программы.

В окне приложения (рис. 1.42) щелкнуть по кнопке *Start*.

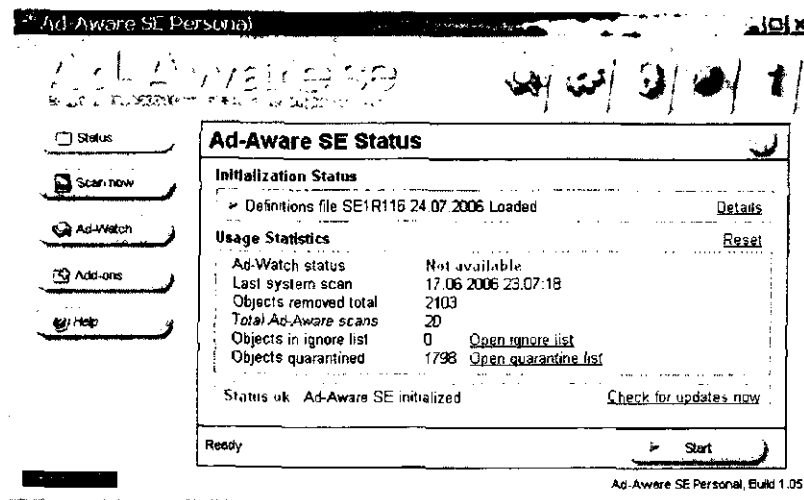


Рис. 1.42. Окно программы Ad-Adware

- В появившемся диалоговом окне (рис. 1.43) выбрать тип сканирования (например, *Perform smart system scan* — *Интеллектуальное сканирование*). Щелкнуть по кнопке *Next*.

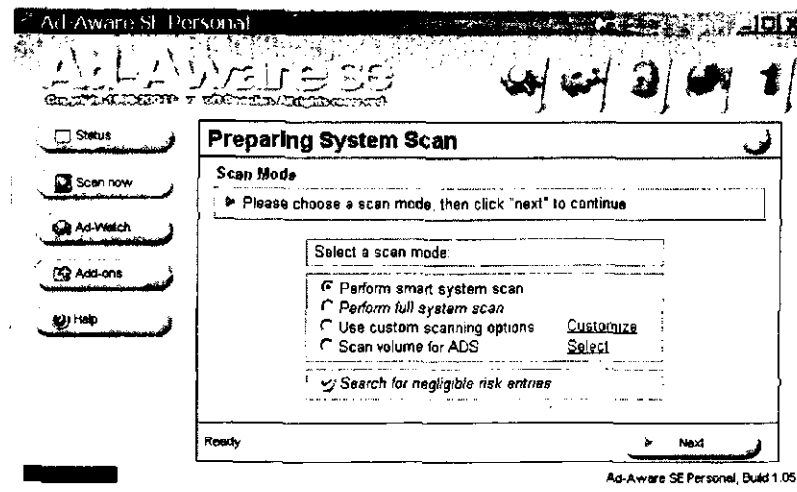


Рис. 1.43. Выбор типа сканирования

- В течение некоторого времени будет происходить сканирование, процесс которого будет отображаться в диалоговом окне.

4. С результатами сканирования можно ознакомиться в следующем диалоговом окне с использованием вкладок. Активизировать вкладку *Critical Objects* (рис. 1.44) и в контекстном меню выбрать пункт *Select all objects — Выделить все объекты*. В данном случае будут выделены файлы cookies (см. ниже).

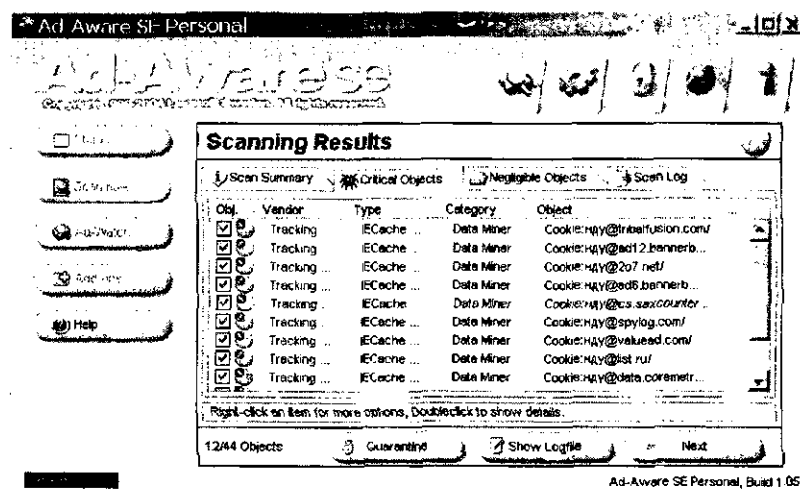


Рис. 1.44. Список найденных критических объектов

5. Активизировать вкладку *Negligible Objects* (рис. 1.45) и в контекстном меню выбрать пункт *Select all objects — Выделить все объекты*.

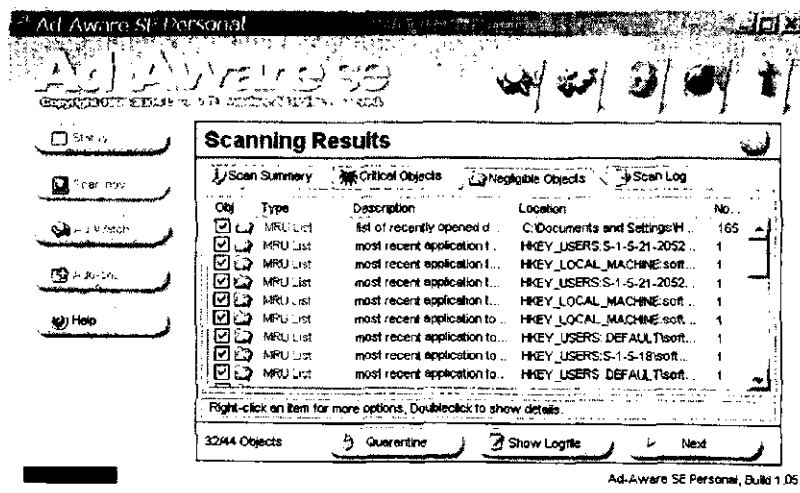


Рис. 1.45. Список найденных подозрительных объектов

В данном случае будут выделены файлы с подозрением на *adware* и *sruware*.

Щелкнуть по кнопке *Next*.

6. В появившемся диалоговом окне можно подтвердить удаление найденных критических и подозрительных объектов (*adware*, *sruware* и *cookies*).

Куки. Куки (от англ. *cookies* — домашнее печенье) — небольшой текстовый файл, помещаемый Web-сервером на локальный компьютер пользователя. Файлы *cookies* могут храниться в оперативной памяти (сеансовые файлы *cookies*) или записываться на жесткий диск (постоянные файлы *cookies*). Файлы *cookies* не могут быть использованы для запуска программного кода (запуска программ) или для заражения компьютера вирусами.

Cookies применяются для сохранения данных, специфичных для данного пользователя. При вводе регистрационных данных файлы *cookies* помогают серверу упростить процесс сохранения персональных данных, связанных с текущим пользователем. Если пользователь Интернет-магазина ранее указывал адрес для доставки счетов или товара, вместо повторного ввода этих данных можно указать пароль, позволяющий автоматически заполнить соответствующие поля в форме заказа.

Браузеры позволяют включать и отключать использование файлов *cookies*, а также выполнять прием файлов *cookies* только после подтверждения со стороны пользователя.

Практическое задание «Упорядочение использования файлов *cookies*». В браузере установить уровень защиты локального компьютера от файлов *cookies*.



Упорядочение использования файлов *cookies*

1. Запустить браузер Internet Explorer и ввести команду [*Сервис-Свойства обозревателя...*].
В появившемся диалоговом окне *Свойства обозревателя* выбрать вкладку *Конфиденциальность* (рис. 1.46).
С помощью вертикального ползунка установить уровень конфиденциальности от значения *Принимать все cookies* до уровня *Блокировать все cookies*.
2. Запустить браузер SeaMonkey и ввести команду [*Инструменты-Менеджер Cookies ...*].
С помощью пунктов меню установить уровень защиты от *cookies*.

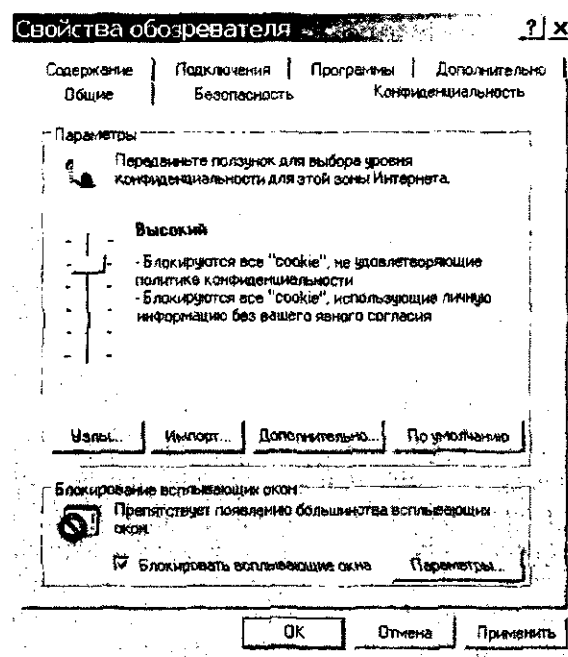



Рис. 1.46. Защита от cookies в браузере Internet Explorer

Контрольные вопросы

1. Каковы вредоносные действия рекламных и шпионских программ?
2. Что такое куки (cookies)? В чем состоит их польза и вред?

Компьютерный практикум

Windows-CD 

1.22. С помощью программы удаления рекламных и шпионских программ (например, Ad-Adware) выполнить практическое задание «Защита от рекламных и шпионских программ».



1.23. В браузерах выполнить практическое задание «Упорядочение использования файлов cookies».



1.6.6. Спам и защита от него

Спам (от англ. spam) — это массовая автоматическая рассылка рекламных электронных сообщений, со скрытым или фальсифицированным обратным адресом. Спам распространяется по компьютерным сетям с использованием электронной почты и систем интерактивного общения (типа ICQ), а также по мобильным сетям с использованием службы SMS-сообщений.



Спам — массово рассылаемая корреспонденция рекламного или иного характера, отправляемая людям, не выразившим желание ее получить. В первую очередь термин «спам» относится к рекламным электронным письмам.

Спам приходит потому, что электронный адрес получателя стал известен спамерам (рассыльщикам спама). Чаще всего владелец почтового ящика сам указывает электронный почтовый адрес при регистрации на каком-либо сайте и его обнаруживает специальный робот, «бродящий» по сайтам наподобие индексирующего робота поисковых систем.

Спамеры стремятся получить подтверждение, что почтовый адрес действительно используется (в этом случае поток спама может увеличиться многократно). Чтобы убедиться, что спамовое сообщение получено и прочитано, спамеры применяют различные уловки:

- требуется подтверждение о получении сообщения;
- предлагается активизировать ссылку на Web-страницу, на которой предлагается получить дополнительную информацию;
- предлагается отменить подписку на эту рассылку, по-слав письмо по указанному адресу.

Рекламный спам. Некоторые компании, занимающиеся легальным бизнесом, рекламируют свои товары или услуги с помощью спама. Они могут осуществлять его рассылку самостоятельно, но чаще заказывают ее тем компаниям (или лицам), которые на этом специализируются. Привлекательность такой рекламы заключается в ее сравнительно низкой стоимости и большом охвате потенциальных клиентов.

С помощью спама часто рекламируют продукцию, о которой нельзя сообщить другими способами, например оружие, порнографию, лекарственные средства с ограничениями по обороту, ворованную информацию (базы данных), контрафактное программное обеспечение и т. п.

«Нигерийские письма». Иногда спам используется для того, чтобы выманить деньги у получателя письма. Наиболее распространенный способ получил название «нигерийские письма», потому что большое количество таких писем приходило из Нигерии. Такое письмо содержит сообщение о том, что получатель письма может получить большую сумму денег, а отправитель может ему в этом помочь. Затем отправитель письма просит перевести ему немного денег под предлогом, например, оформления документов или открытия счета. Выманивание этой суммы и является целью мошенников.

Фишинг. Фишинг (от англ. fishing — рыбалка) — еще один способ мошенничества путем обмана пользователей. Он представляет собой попытку выманить у получателя письма данные, которые можно использовать для получения выгоды: номера его кредитных карточек или пароли доступа к системам онлайн-платежей. Такое письмо обычно маскируется под официальное сообщение от администрации банка. В нем говорится, что получатель должен подтвердить сведения о себе, иначе его счет будет заблокирован, и приводится адрес сайта (принадлежащего спамерам) с формой, которую надо заполнить. Среди данных, которые требуется сообщить, присутствуют и те, которые нужны мошенникам. Для того чтобы жертва не догадалась об обмане, оформление этого сайта имитирует оформление официального сайта банка.

Защита от спама. В силу массового характера спамовые почтовые рассылки затрудняют работу информационных систем и ресурсов, создавая для них бесполезную нагрузку. Пользователи сети вынуждены ежедневно тратить время на обработку бесполезных рекламных сообщений, а провайдером спам приносит неудобства вследствие повышения нагрузки на почтовые серверы и каналы связи.

Для борьбы со спамом используются антиспамовые фильтры, которые могут быть установлены как на локальных компьютерах пользователей, так и на почтовых серверах провайдеров. Антиспамовые фильтры анализируют содержание письма или пытаются опознать спамера по электронному адресу. Если письмо классифицировано как спам, оно может быть помечено, перемещено в другую папку или даже удалено.

Для затруднения автоматической фильтрации спамовые сообщения часто искажаются, вместо букв используются похожие по начертанию цифры, русские буквы заменяются на латинские, а в случайных местах добавляются пробелы.

Практическое задание «Защита от спама». В почтовых программах (например, Outlook Express и SeaMonkey) создать антиспамовый фильтр (правила для работы с почтой), который будет защищать от получения рекламных сообщений:

- от конкретных отправителей;
- при наличии определенных слов в сообщении.



Защита от спама

Создадим антиспамовый фильтр в почтовой программе Outlook Express.

1. Запустить почтовую программу Outlook Express. Ввести команду [*Сервис-Правила для сообщений-Почта...*].
2. В появившемся диалоговом окне *Правила для сообщений* выбрать вкладку *Правила для почты* и щелкнуть по кнопке *Создать*.
3. Появившееся диалоговое окно *Создать правило для почты* (рис. 1.47) содержит три списка и текстовое поле. В первом списке необходимо выбрать условия применения правила, в данном случае *Искать сообщения, содержащие адресатов в поле «От»* и *Искать сообщения, содержащие заданные слова*.

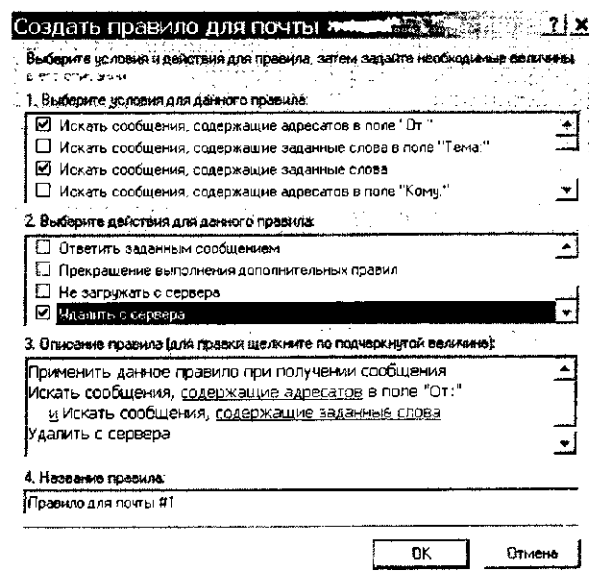


Рис. 1.47. Создание антиспамового фильтра в почтовой программе Outlook Express

4. Во втором списке необходимо выбрать действие над сообщениями для данного правила, в данном случае *Удалить с сервера*.
5. В третьем списке необходимо уточнить порядок действия данного правила.
Щелчком по ссылке *содержащие адресатов* вызвать диалоговое окно *Выбор получателей* и ввести почтовые адреса спамеров, от которых почтовые сообщения не будут получаться.
Щелчком по ссылке *содержащие заданные слова* вызвать диалоговое окно *Ввод ключевых слов* и ввести ключевые слова, при наличии которых в почтовых сообщениях эти сообщения будут получаться.
6. Текстовое поле содержит название и номер созданного правила. После щелчка по кнопке *ОК* правило вступает в действие.

Создать антиспамовый фильтр в почтовой программе SeaMonkey.

1. Запустить почтовую программу SeaMonkey. Ввести команду [*Инструменты-Фильтры сообщений...*].
2. В появившемся диалоговом окне *Правила фильтрации* (рис. 1.48) в текстовое поле ввести имя фильтра (например, *Антиспамовый фильтр*).

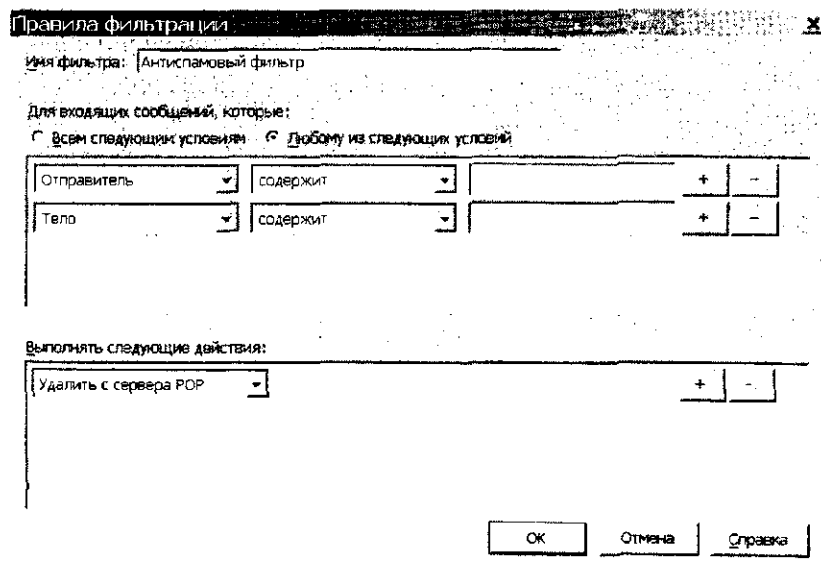


Рис. 1.48. Создание антиспамового фильтра в почтовой программе SeaMonkey


3. Во втором поле добавить условия применения фильтра щелчком по кнопке +.
В первом условии ввести почтовые адреса спамеров, от которых почтовые сообщения не будут получаться.
Во втором условии ввести ключевые слова, при наличии которых в почтовых сообщениях эти сообщения не будут получаться.
4. В третьем поле необходимо выбрать действие над сообщениями для данного фильтра, в данном случае *Удалить с сервера POP*.

Контрольные вопросы

1. Каковы вредные последствия спама?
2. Какие существуют основные типы спамовых сообщений?

Компьютерный практикум



Windows-CD 

- 1.24. В почтовых программах (например, Outlook Express и SeaMonkey) выполнить практическое задание «Защита от спама».



1.6.7. Хакерские утилиты и защита от них

В тех случаях, когда затруднения в работе или утере данных возникает в результате направленных действий, говорят о сетевых атаках.

Сетевые атаки. Сетевые атаки на удаленные серверы реализуются с помощью специальных программ, которые посылают на них специфические запросы. Это приводит к отказу в обслуживании («зависанию» сервера), если ресурсы атакуемого сервера недостаточны для обработки всех поступающих запросов.

DoS-программы (от англ. Denial of Service — отказ в обслуживании) реализуют атаку с одного компьютера с ведома пользователя. DoS-программы обычно наносят ущерб удаленным компьютерам и сетям, не нарушая работоспособность зараженного компьютера.

DDoS-программы (от англ. Distributed DoS — распределенный DoS) реализуют распределенные атаки с разных компьютеров, причем без ведома пользователей зараженных компьютеров. Для этого DDoS-программа засылается на

компьютеры «жертв-посредников» и после запуска в зависимости от текущей даты или по команде от хакера начинает сетевую атаку на указанный сервер в сети.



Некоторые сетевые черви содержат в себе DoS-процедуры, атакующие сайты, которые по каким-либо причинам «невзлюбил» автор червя. Так, червь «Codedred» 20 августа 2001 года организовал успешную атаку на официальный сайт президента США, а червь «Mydoom» 1 февраля 2004 года «выключил» сайт компании — производителя дистрибутивов UNIX.

Некоторые хакерские утилиты реализуют фатальные сетевые атаки. Такие утилиты используют уязвимости в операционных системах и приложениях и отправляют специально оформленные запросы на атакуемые компьютеры в сети. В результате сетевой запрос специального вида вызывает критическую ошибку в атакуемом приложении, и система прекращает работу.

Утилиты «взлома» удаленных компьютеров. Утилиты «взлома» удаленных компьютеров предназначены для проникновения в удаленные компьютеры с целью дальнейшего управления ими (используя методы троянских программ типа утилит удаленного администрирования) или для внедрения во «взломанную» систему других вредоносных программ.

Утилиты «взлома» удаленных компьютеров обычно используют уязвимости в операционных системах или приложениях, установленных на атакуемом компьютере. Профилактическая защита от таких хакерских утилит состоит в своевременной загрузке из Интернета обновлений системы безопасности операционной системы и приложений.

Руткиты. Руткит (от англ. root kit — «набор для получения прав root») — программа или набор программ для скрытого взятия под контроль «взломанной» системы.

Термин «rootkit» исторически пришел из мира операционной системы UNIX, и под этим термином понимается набор утилит, которые хакер устанавливает на «взломанном» им компьютере после получения первоначального доступа.

В операционной системе Windows под rootkit принято подразумевать программу, которая внедряется в систему и перехватывает системные функции. Перехват и модификация низкоуровневых функций, в первую очередь, позволяет такой программе достаточно качественно маскировать свое присутствие в системе. Кроме того, как правило, rootkit может маскировать присутствие в системе любых описанных в его кон-

фигурации процессов, каталогов и файлов на диске, ключей в реестре. Многие rootkit устанавливают в систему свои драйверы и службы (они, естественно, также являются «невидимыми»).

Защита от хакерских атак и сетевых червей. Защита компьютерных сетей или отдельных компьютеров от несанкционированного доступа может осуществляться с помощью межсетевого экрана, или брандмауэра (от англ. firewall). Межсетевой экран может быть реализован как аппаратно, так и программно.

Межсетевой экран позволяет:

- блокировать хакерские DoS-атаки, не пропуская на защищаемый компьютер сетевые пакеты с определенных серверов (определенных IP-адресов или доменных имен);
- не допускать проникновение на защищаемый компьютер сетевых червей (почтовых, Web и др.);
- препятствовать троянским программам отправлять конфиденциальную информацию о пользователе и компьютере.

Практическое задание «Настройка межсетевого экрана». Установить и настроить межсетевой экран (например, Agnitum Outpost).



Настройка межсетевого экрана

Установим политику защиты сетевого экрана, которая определяет, какие сетевые соединения разрешены или запрещены:

- *Блокировать все* — запрет любых сетевых взаимодействий;
- *Режим блокировки* — запрет всех сетевых взаимодействий, кроме явно разрешенных;
- *Режим обучения* — пользователь может принять решение о возможности или запрете выхода в сеть для приложения. Межсетевой экран выдает диалоговое окно предупреждения о первом сетевом взаимодействии для данного приложения, в котором можно настроить правила доступа данного приложения в сеть (эта политика задается по умолчанию);
- *Режим бездействия* — разрешение любых сетевых взаимодействий;
- *Режим разрешения* — разрешение всех сетевых взаимодействий, кроме явно запрещенных.

1. Запустить межсетевой экран Agnitum Outpost.
Ввести команду [*Параметры-Политики...*].

2. В появившемся диалоговом окне *Параметры* активизировать вкладку *Политики* (рис. 1.49) и выбрать политику защиты сетевого экрана. Щелкнуть по кнопке *ОК*.

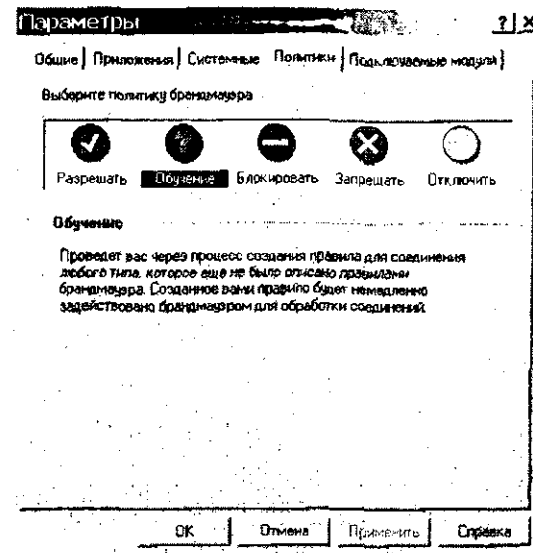


Рис. 1.49. Установка политики межсетевого экрана

При установленной политике *Режим обучения* каждое приложение будет запрашивать установление индивидуальной политики доступа к Интернету.

3. При попытке получения электронной почты появится диалоговое окно, запрашивающее разрешение данному приложению на доступ к Интернету (рис. 1.50).

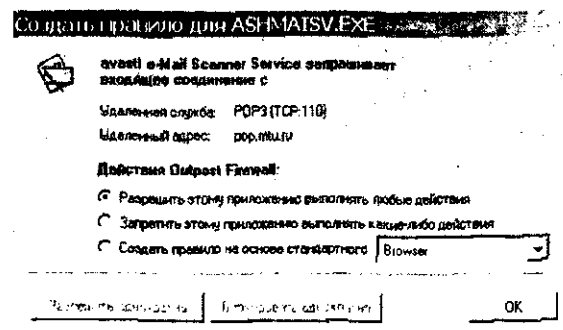


Рис. 1.50. Запрос на разрешение доступа к Интернету

Можно получить информацию о разрешенных и заблокированных приложениях.

4. Открыть окно межсетевое экрана (рис. 1.51) и в левой его части в иерархическом меню выбрать интересующий тип приложений. В правой части окна межсетевое экрана будет выведен список приложений с дополнительной информацией о типе соединения.

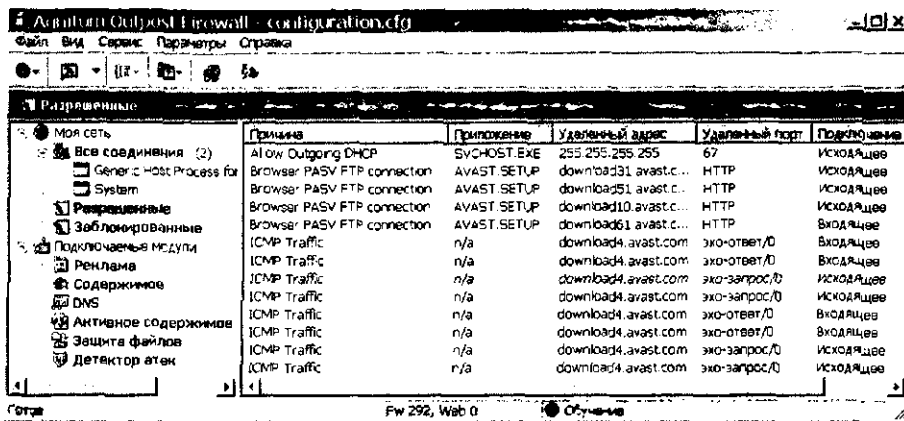


Рис. 1.51. Разрешенные приложения

С помощью межсетевое экрана можно установить уровень тревоги и заблокировать хакерские DoS-атаки.

5. Ввести команду [Параметры-Подключаемые модули...]. В появившемся диалоговом окне выбрать подключаемый модуль *Детектор атак* и щелкнуть по кнопке *Параметры...* В появившемся диалоговом окне *Параметры* (рис. 1.52) с

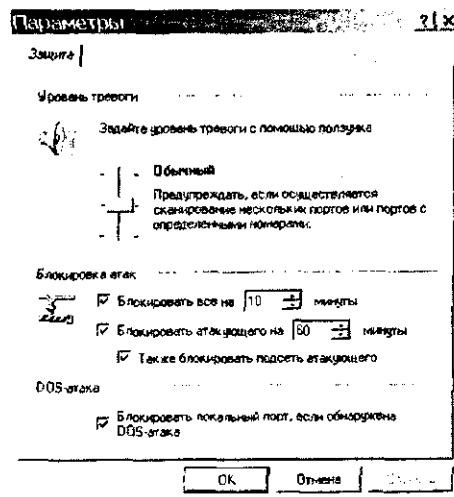


Рис. 1.52. Установка параметров защиты от DoS-атак

помощью вертикального ползунка выбрать уровень тревоги. Установить флажки в разделе *Блокировка атак* и с помощью счетчиков установить время блокировки.

- Открыть окно межсетевого экрана и в левой его части в иерархическом меню выбрать пункт *Детектор атак* (рис 1.53).

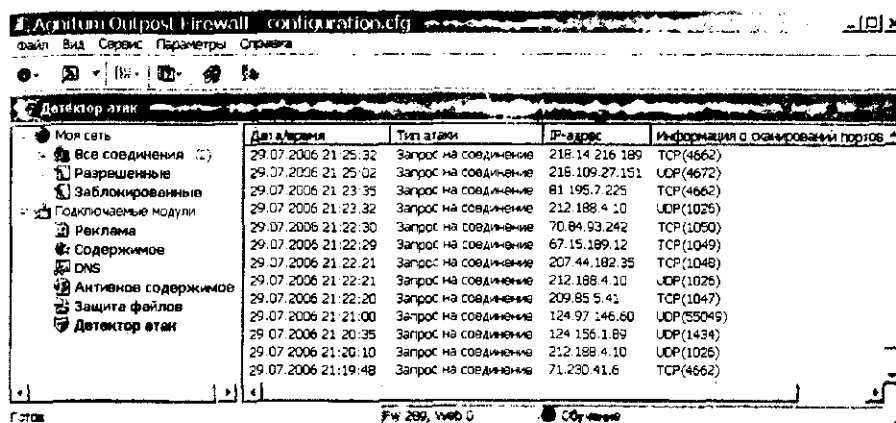



Рис. 1.53. Детектор DoS-атак

В правой части окна межсетевого экрана будет выведен список атак с дополнительной информацией о каждой атаке.

Контрольные вопросы

- Каковы основные хакерские утилиты и их злонамеренное использование?
- В каких целях используются межсетевые экраны?

Компьютерный практикум


Windows-CD 

- С помощью межсетевого экрана (например, Agnitum Outpost) выполнить практическое задание «Настройка межсетевого экрана».



Глава 2

Информация. Системы счисления

Windows-CD 

В процессе изучения данной темы рекомендуется установить программное обеспечение в операционной системе Windows:

- компьютерные калькуляторы *Wise Calculator* и *NumLock Calculator*;
- программа перевода единиц измерения различных величин, в том числе информации.



2.1. Понятие «информация» в науках о неживой и живой природе, обществе и технике

Слово «информация» происходит от латинского слова «informatio», что в переводе означает сведение, разъяснение, ознакомление. Понятие «информация» является базовым в курсе информатики, невозможно дать его определение через другие, более «простые» понятия. В геометрии, например, невозможно выразить содержание базовых понятий «точка», «луч», «плоскость» через более простые понятия. Содержание основных, базовых понятий в любой науке должно быть пояснено на примерах или выявлено путем их сопоставления с содержанием других понятий.

В случае с понятием «информация» проблема его определения еще более сложная, так как оно является общенаучным понятием. Понятие «информация» используется в различных науках (информатике, кибернетике, биологии, физике и др.), при этом в каждой науке понятие «информация» связано с различными системами понятий.

2.1.1. Информация в физике

В физике информация рассматривается как антиэнтропия, или энтропия с обратным знаком. Энтропия системы яв-

Информация в открытых системах. Современная физика рассматривает открытые системы, которые обмениваются энергией или веществом с окружающей средой и увеличивают свою организованность. По мере увеличения организованности системы величина энтропии уменьшается и величина информации увеличивается.

В открытых системах в результате самоорганизации могут возникнуть пространственные или временные структуры, хранящие информацию. Примерами могут служить лазер, создающий когерентное излучение, жидкости, образующие пространственные или временные структуры при нагревании, или химические реакции, в которых наблюдаются периодические пространственные спирали или концентрические волны. Процессы самоорганизации в открытых системах изучаются синергетикой.

Информация в микро- и мегамире. В конце XIX века в классической физике рассматривали нашу Вселенную как замкнутую систему и предсказывали, что ее ждет «тепловая смерть», когда молекулы и атомы равномерно распределятся в пространстве и какие-либо изменения и развитие прекратятся.

Однако современная наука установила, что некоторые законы классической физики, справедливые для макротел, нельзя применять для микро- и мегамира. Согласно современным научным представлениям, наша Вселенная является динамически развивающейся системой, в которой постоянно происходят процессы усложнения структуры.

Астрономия-11

Согласно теории Большого взрыва, Вселенная образовалась около 15 миллиардов лет назад в результате взрыва «первоматерии» (рис. 2.2). В первые мгновения материя существовала фактически в форме энергии, а затем на протяжении долей секунды начало образовываться вещество в форме элементарных частиц (электронов, протонов, нейтронов и др.).

В следующий миллион лет основные события развивались в микромире. Из разлетающихся во все стороны элементарных частиц образовывались атомы, то есть из хаоса возникали системы с более сложной структурой. Сначала образовывались атомы самых легких химических элементов (водорода и гелия), а затем и более тяжелых элементов (рис. 2.3).

В мегамире в течение последующих миллиардов лет под действием сил гравитационного притяжения из хаоса гигант-

ских облаков пыли и газа формировались сложные структуры — галактики (рис. 2.4).

Краткая история вселенной

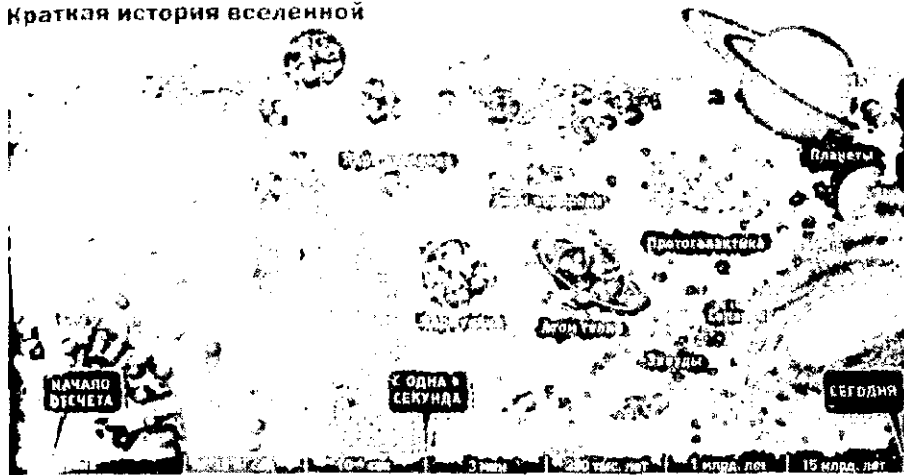


Рис. 2.2. Развитие Вселенной



Рис. 2.3. Переход от хаоса к порядку в микромире

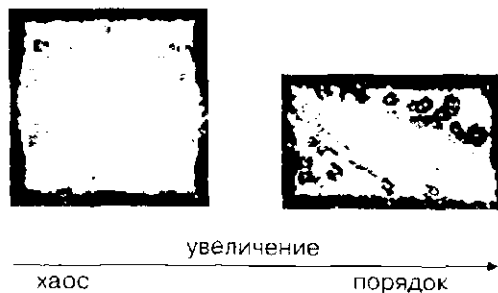


Рис. 2.4. Переход от хаоса к порядку в макромире

Наша солнечная система, в которую входит планета Земля, образовалась около 5 миллиардов лет назад и вместе с сотнями миллионов других звезд образует нашу галактику Млечный Путь.

На поверхности планет стали происходить химические реакции, в результате которых из атомов образовывались более сложные системы — молекулы веществ. В том числе молекула воды, которая состоит из двух атомов водорода и одного атома кислорода.

Химия 9

Таким образом, с одной стороны, в неживой природе в замкнутых системах идут процессы в направлении от порядка к хаосу (в них информация уменьшается). С другой стороны, в открытых системах может происходить усложнение структуры и, следовательно, информация, являющаяся мерой упорядоченности элементов системы, возрастает.



Мы живем в макромире, т. е. в мире, который состоит из объектов, по своим размерам сравнимых с человеком (рис. 2.5). Обычно макрообъекты разделяют на неживые (камень, льдина и т. д.), живые (растения, животные, сам человек) и искусственные (здания, средства транспорта, станки и механизмы, компьютеры и т. д.).

Макрообъекты состоят из молекул и атомов, которые, в свою очередь, состоят из элементарных частиц, размеры которых чрезвычайно малы. Этот мир называется микромиром (рис. 2.6).



Рис. 2.5. Макромир.
Гулливер в стране
лилипутов

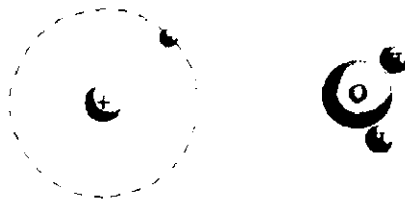


Рис. 2.6. Микромир. Атом водорода и молекула воды

Мы живем на планете Земля, которая входит в солнечную систему, Солнце вместе с сотнями миллионов других звезд образует нашу галактику Млечный Путь, а миллиарды галактик образуют Вселенную. Все эти объекты имеют громадные размеры и образуют **мегамир** (рис. 2.7).



Рис. 2.7. Мегамир. Солнечная система

Контрольные вопросы

1. Приведите примеры перехода от хаоса к порядку (увеличения информации) в окружающем мире.
2. Приведите примеры перехода от порядка к хаосу (уменьшения информации) в окружающем мире.

2.1.2. Информация в биологии

Процесс увеличения информации характерен для открытых, обменивающихся веществом и энергией с окружающей средой, саморазвивающихся систем живой природы (белковых молекул, организмов, популяций животных и т. д.). В процессе эволюции живой природы идет саморазвитие, т. е. повышение сложности и разнообразия живых организмов, а следовательно, увеличение информации, понимаемой как мера упорядоченности элементов системы.

Информационные сигналы. В биологии, которая изучает живую природу, понятие «информация» связывается с целесообразным поведением живых организмов. Такое поведение строится на основе получения и использования организмом информации об окружающей среде в форме информационных сигналов. Информационные сигналы могут иметь различную физическую или химическую природу: звук, свет, запах и другие.

Простейшие (например, амеба) могут получать информацию лишь о химическом составе и температуре окружающей среды. Причем информация может быть получена только о ближайших областях окружающей среды путем непосредственного контакта простейшего со средой.

Человек воспринимает информацию об окружающем мире с помощью органов чувств (зрения, слуха, обоняния, осязания и вкуса), причем может это делать на большом расстоянии. Чувствительные нервные окончания органов чувств (рецепторы) воспринимают воздействие (например, на глазном дне колбочки и палочки реагируют на воздействие световых лучей) и передают его нейронам (нервным клеткам), цепи которых составляют нервную систему.

Нейрон может находиться в двух состояниях: невозбужденном или возбужденном. Возбужденный нейрон генерирует электрический импульс, который передается по нервной системе. В нервной системе происходит кодирование и передача информации с помощью двух состояний нейрона: нет импульса, есть импульс.

В этом случае сами состояния нейрона можно рассматривать как знаки некоторого алфавита нервной системы, с помощью которого происходит передача информации.

Генетическая информация. Понятие «информация» в биологии используется также в связи с исследованиями механизмов наследственности. Генетическая информация передается по наследству и хранится во всех клетках живых организмов. Гены представляют собой сложные молекулярные структуры, содержащие информацию о строении живых организмов. Последнее обстоятельство позволило проводить научные эксперименты по клонированию, т. е. созданию точных копий организмов из одной клетки.

Общая биология 10–11

Генетическая информация определяет строение и развитие живых организмов и передается по наследству.

Хранится генетическая информация в клетках организмов в структуре молекул ДНК (дезоксирибонуклеиновой кислоты). Молекула ДНК (рис. 2.8) состоит из двух скрученных друг с другом в спираль цепей, построенных из четырех нуклеотидов: А, G, Т и С, которые образуют генетический код.

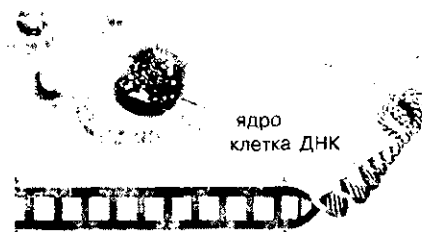


Рис. 2.8. Молекула ДНК

Молекула ДНК человека включает в себя около 3 миллиардов пар нуклеотидов. В ней закодирована вся информация об организме человека: его внешность, предрасположенность к болезням, способности и т. д.



В живых организмах информация передается и хранится с помощью объектов различной физической природы (состояние нейрона, нуклеотиды в молекуле ДНК), которые могут рассматриваться как знаки биологических алфавитов.

Контрольные вопросы

1. Какова физическая природа знака при представлении информации в нервной системе? В генетическом коде?

2.1.3. Информация в общественных науках

Социально-значимые свойства информации. Человек — существо социальное, для общения с другими людьми он должен обмениваться с ними информацией, причем обмен информацией всегда производится на определенном языке: русском, английском и т. д. Участники дискуссии должны владеть тем языком, на котором ведется общение, тогда информация будет понятной всем участникам обмена информацией.

Информация должна быть **полезной**, тогда дискуссия приобретает практическую ценность. Бесполезная информация создает информационный шум, который затрудняет восприятие полезной информации. Примерами передачи и получения бесполезной информации могут служить некоторые конференции и чаты в Интернете.

Широко известен термин «**средства массовой информации**» (газеты, радио, телевидение), которые доводят информацию до каждого члена общества. Такая информация должна быть **достоверной и актуальной**. Недостоверная информация вводит членов общества в заблуждение и может быть причиной возникновения социальных потрясений. Неактуальная информация бесполезна и поэтому никто, кроме историков, не читает прошлогодних газет.

Для того чтобы человек мог правильно ориентироваться в окружающем мире, информация должна быть **полной и точной**. Задача получения полной и точной информации стоит перед наукой. Владение научными знаниями в процессе обучения позволяет человеку получать полную и точную информацию о природе, обществе и технике.

Информация и знания. Человек получает информацию из окружающего мира с помощью органов чувств, анализирует ее и выявляет существенные закономерности с помощью мышления, хранит полученную информацию в памяти. Процесс систематического научного познания окружающего мира приводит к накоплению информации в форме знаний (фактов, научных теорий и т. д.). Таким образом, с точки зрения процесса познания информация может рассматриваться как **знания**.

Процесс познания можно наглядно изобразить в виде расширяющегося круга знания (такой способ придумали еще древние греки). Вне этого круга лежит область незнания, а окружность является границей между знанием и незнанием (рис. 2.9). Парадокс состоит в том, что чем большим объемом знаний обладает человек, чем шире круг знаний, тем больше



Рис. 2.9. Знание и незнание

он ощущает недостаток знаний, и тем больше граница нашего незнания, мерой которого в этой модели является длина окружности.

Так, объем знаний выпускника школы гораздо больше, чем объем знаний первоклассника, однако и граница его незнания существенно больше. Действительно, первоклассник ничего не знает о законах физики и его это не очень волнует, тогда как выпускник школы при подготовке к экзаменам по физике может обнаружить, что существуют физические законы, которые он не знает или не понимает.

Информацию, которую получает человек, можно считать мерой уменьшения неопределенности знания. Если некоторое сообщение приводит к уменьшению неопределенности нашего знания, то можно говорить, что такое сообщение содержит информацию.

Например, после сдачи экзамена по информатике вы мучаетесь неопределенностью, вы не знаете, какую оценку получили. Наконец, экзаменационная комиссия объявляет результаты экзамена и вы получаете сообщение, которое приносит полную определенность, теперь вы знаете свою оценку. Происходит переход от незнания к полному знанию, значит сообщение экзаменационной комиссии содержит информацию.

Контрольные вопросы

1. Какие социально-значимые свойства информации можно выделить?
2. Как можно представить процесс познания?

2.1.4. Информация в кибернетике

В кибернетике (науке об управлении) понятие «информация» используется для описания процессов управления в сложных системах (живых организмах или технических устройствах). Жизнедеятельность любого организма или нормальное функционирование технического устройства связано с процессами управления, благодаря которым поддерживаются в необходимых пределах значения его параметров. Так для поддержания своей жизнедеятельности любой живой организм постоянно получает информацию из внешнего мира с помощью органов чувств, обрабатывает ее и управляет своим поведением (например, перемещаясь в пространстве, избегает опасности).



Процессы управления включают в себя получение, хранение, преобразование и передачу информации.

В процессе управления полетом самолета в режиме автопилота бортовой компьютер получает информацию от датчиков (скорости, высоты и т. д.), обрабатывает ее и передает команды на исполнительные механизмы, изменяющие режим полета (закрылки, клапаны, регулирующие работу двигателей, и т. д.).

В любом процессе управления всегда происходит взаимодействие двух объектов — управляющего и управляемого, которые соединены каналами прямой и обратной связи. По каналу прямой связи передаются управляющие сигналы, а по каналу обратной связи — информация о состоянии управляемого объекта.

Разомкнутые системы управления. Если в процессе управления не учитывается состояние управляемого объекта и обеспечивается управление только по прямому каналу (от управляющего объекта к управляемому), то такие системы управления называются разомкнутыми. Информационную модель разомкнутой системы управления можно наглядно представить с помощью следующей схемы (рис. 2.10).

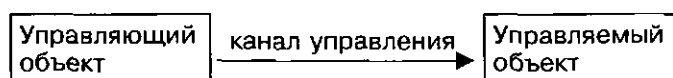


Рис. 2.10. Разомкнутая система управления

В качестве примера разомкнутой системы управления рассмотрим процесс записи информации на гибкий диск, в котором управляющий объект *Контроллер дисковод* изменяет состояние управляемого объекта *Магнитной головки*.

Для того чтобы информация могла быть записана, необходимо установить магнитную головку дисковода над определенной концентрической дорожкой диска. При записи информации на гибкие диски не требуется особой точности установки (имеется всего 80 дорожек) и можно не учитывать возможные (например, от нагревания) механические деформации носителя, поэтому управляющий объект (*Контроллер дисковода*) просто перемещает управляемый объект (*Магнитную головку*) на определенное расстояние вдоль радиуса дискеты.

Замкнутые системы управления. В замкнутых системах управления управляющий объект по прямому каналу управления производит необходимые действия над объектом управления, а по каналу обратной связи получает информацию о его реальных параметрах (рис. 2.11). Это позволяет осуществлять управление с гораздо большей точностью.

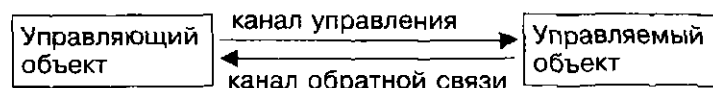


Рис. 2.11. Замкнутая система управления

Информационную модель замкнутой системы управления можно наглядно представить с помощью следующей схемы:

Примером использования замкнутой системы управления являются жесткие диски. При записи информации на жесткие диски требуется особая точность установки магнитных головок, так как на рабочей поверхности носителя имеются десятки тысяч дорожек и необходимо учитывать механические деформации магнитного носителя (например, в результате изменения температуры). Управляющий объект (*Контроллер жесткого диска*) постоянно получает информацию о реальном положении управляемого объекта (*Магнитных головок*) по каналу обратной связи, а по прямому каналу выставляет головки над поверхностью носителя с большой точностью.

Контрольные вопросы

1. В чем состоит различие между разомкнутыми и замкнутыми системами управления? Приведите примеры.

2.2. Количество информации как мера уменьшения неопределенности знания

1.3.1. Количество информации как мера уменьшения неопределенности знания

Информатика-8



Уменьшение неопределенности знания. Получение информации можно связать с уменьшением неопределенности

знания. Это позволяет количественно измерять информацию, что чрезвычайно важно для информатики. Рассмотрим вопрос об определении количества информации более подробно на конкретных примерах.

Пусть у нас имеется монета, которую мы бросаем на ровную поверхность. С равной вероятностью произойдет одно из двух возможных событий — монета окажется в одном из двух положений: «орел» или «решка».

i Можно говорить, что события равновероятны, если при возрастающем числе опытов количества выпадений «орла» и «решки» постепенно сближаются. Например, если мы бросим монету 10 раз, то «орел» может выпасть 7 раз, а решка — 3 раза, если бросим монету 100 раз, то «орел» может выпасть 60 раз, а «решка» — 40 раз, если бросим монету 1000 раз, то «орел» выпадет 520 раз, а «решка» — 480 и т. д. В итоге при очень большой серии опытов количества выпадений «орла» и «решки» практически сравниваются.

Перед броском существует неопределенность нашего знания (возможны два события), и как упадет монета, предсказать невозможно. После броска наступает полная определенность, так как мы видим (получаем зрительное сообщение), что монета в данный момент находится в определенном положении (например, «орел»). Это сообщение приводит к уменьшению неопределенности нашего знания в два раза, так как из двух возможных равновероятных событий реализовалось одно (рис. 2.12).

Возможные события	Произошедшее событие
	
	

Рис. 2.12. Возможные и произошедшее события

В окружающей действительности достаточно часто встречаются ситуации, когда может произойти некоторое количество

равновероятных событий. Так, при бросании равносторонней четырехгранной пирамиды существуют 4 равновероятных события, а при бросании шестигранного игрального кубика — 6 равновероятных событий.

Чем больше количество возможных событий, тем больше начальная неопределенность нашего знания и соответственно тем большее количество информации будет содержать сообщение о результатах опыта.

Единицы измерения количества информации. Для количественного выражения любой величины необходимо определить единицу измерения. Так, для измерения длины в качестве единицы выбран метр, для измерения массы — килограмм и т. д. Аналогично, для определения количества информации необходимо ввести единицу измерения.

1.3.2. Определение количества информации

Информатика и ИКТ-8



За единицу количества информации принимается такое количество информации, которое содержит сообщение, уменьшающее неопределенность знания в два раза. Такая единица названа бит.

Если вернуться к опыту с бросанием монеты, то здесь неопределенность как раз уменьшается в два раза (из двух возможных событий реализуется одно) и, следовательно, количество полученной информации равно 1 биту.

Минимальной единицей измерения количества информации является бит, а следующей по величине единицей — байт, причем

$$1 \text{ байт} = 2^3 \text{ битов} = 8 \text{ битов.}$$

В информатике система образования кратных единиц измерения количества информации несколько отличается от принятых в большинстве наук. Традиционные метрические системы единиц, например Международная система единиц СИ, в качестве множителей кратных единиц использует коэффициент 10^n , где $n = 3, 6, 9$ и т. д., что соответствует десятичным приставкам «кило» (10^3), «мега» (10^6), «гига» (10^9) и т. д.

Компьютер оперирует числами не в десятичной, а в двоичной системе счисления, поэтому в кратных единицах измерения количества информации используется коэффициент 2^n .

Так, кратные байту единицы измерения количества информации вводятся следующим образом:

- 1 Кбайт = 2^{10} байт = 1024 байт;
- 1 Мбайт = 2^{10} Кбайт = 1024 Кбайт;
- 1 Гбайт = 2^{10} Мбайт = 1024 Мбайт.

Практическое задание «Перевод единиц измерения количества информации». Произвести перевод количества информации с помощью специализированной программы перевода единиц измерения (например, VersaVerter).



Перевод единиц измерения количества информации

1. Запустить программу VersaVerter (Перевод единиц измерения).
2. В открывшемся диалоговом окне программы выбрать вкладку *Data*.
3. В левом окне диалогового окна выбрать исходную единицу измерения (например, *gigabyte*) и в текстовом поле ввести числовое значение (например, 1).

После выбора в правом окне конечной единицы измерения (например, *byte*) в текстовом поле появится числовое значение (в данном случае, 1 073 741 824) (рис. 2.13).

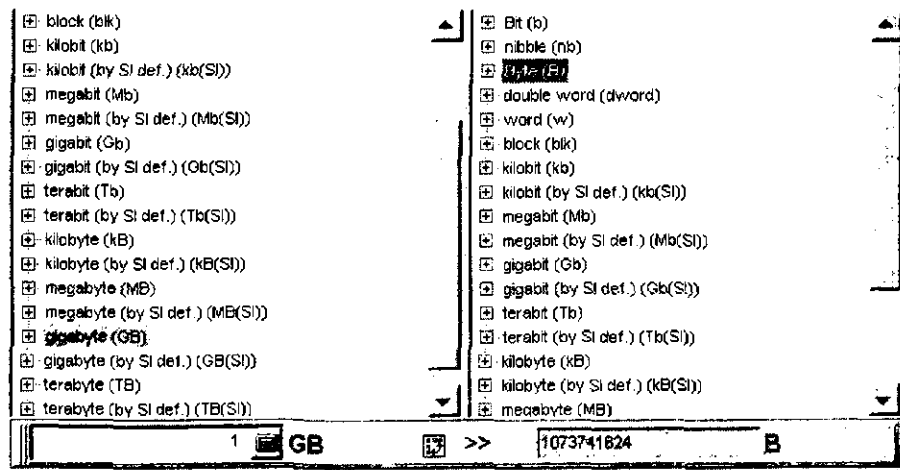


Рис. 2.13. Перевод единиц измерения количества информации

Количество возможных событий и количество информации. Существует формула, которая связывает между собой количество возможных событий N и количество информации I :

$$N = 2^I. \quad (2.1)$$

По этой формуле можно легко определить количество возможных событий, если известно количество информации. Например, если мы получили 4 бита информации, то количество возможных событий составляло

$$N = 2^4 = 16.$$

Наоборот, для определения количества информации, если известно количество событий, необходимо решить показательное уравнение относительно I .

Задание «Определение количества информации». Определить количество информации, полученной вторым игроком после первого хода первого игрока, в игре «крестики-нолики» на поле размером 8×8 клеток.

Перед первым ходом существует 64 возможных события (64 различных вариантов расположения «крестика»), тогда уравнение (2.1) принимает вид:

$$64 = 2^I.$$

Так как $64 = 2^6$, то

$$2^6 = 2^I.$$

Таким образом, $I = 6$ битов, т. е. количество информации, полученной вторым игроком после первого хода первого игрока, составляет 6 битов.

Контрольные вопросы

1. Приведите примеры уменьшения неопределенности знания после получения информации о произошедшем событии.
2. Как зависит количество информации от количества возможных событий?

Задания

- 2.1. В рулетке общее количество лунок равно 128. Какое количество информации мы получаем в зрительном сообщении об остановке шарика в одной из лунок?

Компьютерный практикум



Windows-CD 

- 2.1. С помощью программы перевода единиц измерения выполнить практическое задание «Перевод единиц измерения количества информации».



2.2. Заполнить пропуски числами и проверить правильность вычислений с помощью программы перевода единиц измерения:



- а) 5 Кбайт = ___ байт = ___ бит;
 б) ___ Кбайт = ___ байт = 12 288 бит;
 в) ___ Кбайт = ___ байт = 2^{13} бит;
 г) ___ Гбайт = 1536 Мбайт = ___ Кбайт;
 д) 512 Кбайт = 2___ байт = 2___ бит.

2.3. Алфавитный подход к определению количества информации

Определение количества информации на основе уменьшения неопределенности нашего знания рассматривает информацию с точки зрения содержания, ее понятности и новизны для человека. С этой точки зрения в опыте по бросанию монеты одинаковое количество информации содержится и в зрительном образе упавшей монеты, и в коротком сообщении «Орел», и в длинной фразе «Монета упала на поверхность земли той стороной вверх, на которой изображен орел».

Однако при хранении и передаче информации с помощью технических устройств целесообразно отвлечься от содержания информации и рассматривать ее как последовательность символов — знаков (букв, цифр, кодов цветов точек изображения и т. д.).



Набор символов знаковой системы (алфавит) можно рассматривать как различные возможные состояния (события).

Тогда, если считать, что появление символов в сообщении равновероятно, по формуле (2.1) можно рассчитать, какое количество информации несет каждый символ. Информационная емкость знаков зависит от их количества в алфавите (мощности алфавита): чем больше их количество, тем большее количество информации несет один знак.

Так, информационная емкость буквы в русском алфавите, если не использовать букву «ё», составляет:

$$32 = 2^I, \text{ т. е. } I = 5 \text{ битов.}$$

На основании алфавитного подхода можно подсчитать количество информации в сообщении I_c , для этого необходимо умножить количество информации, которое несет один символ I , на количество символов K в сообщении:

$$I_c = I \times K. \quad (2.2)$$

Задание «Определение количества информации в тексте». Система оптического распознавания символов позволяет преобразовывать отсканированные изображения страниц документа в текстовый формат со скоростью 4 страницы в минуту и использует алфавит мощностью 65 536 символов. Какое количество информации будет нести текстовый документ, каждая страница которого содержит 40 строк по 50 символов, после 10 минут работы приложения?

По формуле (2.1) определим информационную емкость символа алфавита:

$$65\,536 = 2^I \Rightarrow 2^{16} = 2^I \Rightarrow I = 16 \text{ битов.}$$

По формуле (2.2) определим количество информации на странице:

$$16 \text{ битов} \times 40 \times 50 = 32\,000 \text{ битов} = 4000 \text{ байтов.}$$

Определим количество информации, которое будет нести текстовый документ:

$$4000 \text{ байтов} \times 4 \times 10 = 160\,000 \text{ байтов} \approx 156 \text{ Кбайт.}$$

З а д а н и я

- 2.2. Пользователь компьютера может вводить в минуту 200 знаков. Мощность используемого алфавита равна 256. Какое количество информации может ввести пользователь в компьютер за 5 минут?

2.4. Формула Шеннона

Существует множество ситуаций, когда возможные события имеют различные вероятности реализации. Например, если монета несимметрична (одна сторона тяжелее другой), то при ее бросании вероятности выпадения «орла» и «решки» будут различаться.

Формулу для вычисления количества информации для событий с различными вероятностями предложил К. Шеннон в 1948 году. В этом случае количество информации определяется по формуле:

$$I = - \sum_{i=1}^N p_i \log_2 p_i, \quad (2.3)$$

- где I — количество информации,
 N — количество возможных событий,
 p_i — вероятности отдельных событий.

Для частного, но широко распространенного и рассмотренного выше случая, когда события равновероятны ($p_i = 1/N$), величину количества информации I можно рассчитать по формуле:

$$I = -\sum_{i=1}^N \frac{1}{N} \log_2 \frac{1}{N} = \log_2 N. \quad (2.4)$$

Задание «Бросание пирамидки». Определить количество информации, которую мы получаем в результате бросания несимметричной и симметричной пирамидок.

При бросании несимметричной четырехгранной пирамидки вероятности отдельных событий равны:

$$p_1 = 1/2, p_2 = 1/4, p_3 = 1/8, p_4 = 1/8.$$

Количество информации, которую мы получим после бросания несимметричной пирамидки, можно рассчитать по формуле (2.3):

$$\begin{aligned} I &= -(1/2 \cdot \log_2 1/2 + 1/4 \cdot \log_2 1/4 + 1/8 \cdot \log_2 1/8 + 1/8 \cdot \log_2 1/8) \text{ битов} = \\ &= (1/2 \cdot \log_2 2 + 1/4 \cdot \log_2 4 + 1/8 \cdot \log_2 8 + 1/8 \cdot \log_2 8) \text{ битов} = \\ &= (1/2 + 2/4 + 3/8 + 3/8) \text{ битов} = 14/8 \text{ битов} = 1,75 \text{ бита}. \end{aligned}$$

При бросании симметричной четырехгранной пирамидки вероятности отдельных событий равны между собой:

$$p_1 = p_2 = p_3 = p_4 = 1/4.$$

Количество информации, которую мы получим после бросания симметричной пирамидки, можно рассчитать по формуле (2.4):

$$I = \log_2 4 = 2 \text{ бита}.$$

Таким образом, при бросании симметричной пирамидки, когда события равновероятны, мы получим большее количество информации (2 бита), чем при бросании несимметричной пирамидки, когда события неравновероятны (1,75 бита).



Количество информации, которую мы получаем, достигает максимального значения, если события равновероятны.



В теории информации доказано, что максимальное количество информации несет сообщение, в котором вероятности появления всех знаков одинаковы.

Количество информации, которое несет знак, зависит от вероятности его получения. Если получатель заранее точно знает, какой знак придет, то полученное количество информации будет равно 0. Наоборот, чем менее вероятно получение знака, тем больше его информационная емкость.

В русской письменной речи частота использования букв в тексте различна, так в среднем на 1000 знаков осмысленного текста приходится 200 букв «а» и в сто раз меньшее количество буквы «ф» (всего 2). Таким образом, с точки зрения теории информации, информационная емкость знаков русского алфавита различна (у буквы «а» она наименьшая, а у буквы «ф» — наибольшая).

Выбор правильной стратегии в игре «Угадай число». На получении максимального количества информации строится выбор правильной стратегии в игре «Угадай число», в которой первый участник загадывает целое число (например, 3) из заданного интервала (например, от 1 до 16), а второй должен «угадать» задуманное число. Если рассмотреть эту игру с информационной точки зрения, то начальная неопределенность знания для второго участника составляет 16 возможных событий (вариантов загаданных чисел).

При правильной стратегии интервал чисел всегда должен делиться пополам, тогда количество возможных событий (чисел) в каждом из полученных интервалов будет одинаково и их отгадывание равновероятно. В этом случае на каждом шаге ответ первого игрока («Да» или «Нет») будет нести максимальное количество информации (1 бит).

Как видно из табл. 2.1, угадывание числа 3 произошло за четыре шага, на каждом из которых неопределенность знания второго участника уменьшалась в два раза за счет получения сообщения от первого участника, содержащего 1 бит информации. Таким образом, количество информации, необходимой для отгадывания одного из 16 чисел, составило 4 бита.

Таблица 2.1. Информационная модель игры «Угадай число»

Вопрос второго участника	Ответ первого участника	Неопределенность знания (количество возможных событий)	Полученное количество информации
		16	
Число больше 8?	Нет	8	1 бит
Число больше 4?	Нет	4	1 бит
Число больше 2?	Да	2	1 бит
Это число 3?	Да	1	1 бит

Практическое задание «Определение количества информации». В непрозрачном мешочке хранятся 10 белых, 20 красных, 30 синих и 40 зеленых шариков. Какое количество информации будет содержать зрительное сообщение о цвете вынутого шарика?

Так как количество шариков различных цветов неодинаково, то вероятности зрительных сообщений о цвете вынутого из мешочка шарика также различаются и равны количеству шариков данного цвета, деленному на общее количество шариков:

$$p_b = 0,1; \quad p_k = 0,2; \quad p_s = 0,3; \quad p_c = 0,4.$$

События неравновероятны, поэтому для определения количества информации, содержащейся в сообщении о цвете шарика, воспользуемся формулой (2.3):

$$I = -(0,1 \cdot \log_2 0,1 + 0,2 \cdot \log_2 0,2 + 0,3 \cdot \log_2 0,3 + 0,4 \cdot \log_2 0,4) \text{ битов.}$$

Для вычисления этого выражения, содержащего логарифмы, воспользуемся компьютерным калькулятором.



Определение количества информации

1. Запустить компьютерный калькулятор Wise Calculator.
2. Ввести выражение и получить результат (рис. 2.14).
Таким образом, $I \approx 1,85$ бита.

$-(0,1 \cdot \log(2,0,1) + 0,2 \cdot \log(2,0,2) + 0,3 \cdot \log(2,0,3) + 0,4 \cdot \log(2,0,4))$
Result: 1.846439344671015

Рис. 2.14. Вычисление количества информации

Контрольные вопросы

1. В каком случае количество информации, полученной о событии, достигает максимального значения?

Задания

- 2.3. Какое количество вопросов достаточно задать вашему собеседнику, чтобы наверняка определить:

- день недели, в котором он родился;
- месяц, в котором он родился;
- число, в которое он родился?

Компьютерный практикум

Windows-CD 

2.3. Выполнить практическое задание «Определение количества информации» для следующих данных:



- 25 белых, 25 красных, 25 синих и 25 зеленых шариков;
- 30 белых, 30 красных, 30 синих и 10 зеленых шариков.

2.5. Кодирование текстовой, графической и звуковой информации

Двоичное кодирование текстовой информации в компьютере

Для представления текстовой информации (прописные и строчные буквы русского и латинского алфавитов, цифры, знаки и математические символы) достаточно 256 различных знаков. По формуле (2.1) можно вычислить, какое количество информации необходимо, чтобы закодировать каждый знак:

$$N = 2^I \Rightarrow 256 = 2^I \Rightarrow 2^8 = 2^I \Rightarrow I = 8 \text{ битов.}$$

Для обработки текстовой информации на компьютере необходимо представить ее в двоичной знаковой системе. Каждому знаку необходимо поставить в соответствие уникальный 8-битовый двоичный код, значения которого находятся в интервале от 00000000 до 11111111 (в десятичном коде от 0 до 255).

Присвоение знаку конкретного двоичного кода — это вопрос соглашения, которое фиксируется в кодовой таблице. К сожалению, в настоящее время существуют пять различных кодовых таблиц для русских букв (*Windows*, *MS-DOS*, *КОИ-8*, *Mac*, *ISO*), поэтому тексты, созданные в одной кодировке, не будут правильно отображаться в другой.

В последние годы широкое распространение получил новый международный стандарт кодирования текстовых символов *Unicode*, который отводит на каждый символ 2 байта (16 битов). По формуле (2.1) можно определить количество символов, которые можно закодировать:

$$N = 2^I = 2^{16} = 65\,536.$$

Такого количества символов оказалось достаточно, чтобы закодировать не только русский и латинский алфавиты, цифры, знаки и математические символы, но и греческий, арабский, иврит и другие алфавиты.

Двоичное кодирование графической информации в компьютере

Графические изображения из аналоговой (непрерывной) формы в цифровую (дискретную) преобразуются путем **пространственной дискретизации**. Изображение разбивается на отдельные маленькие фрагменты (точки, или пиксели), причем каждый элемент может иметь свой цвет (красный, зеленый, синий и т. д.).



Пиксель — минимальный участок изображения, которому независимым образом можно задать цвет.

В результате пространственной дискретизации графическая информация представляется в виде растрового изображения, которое формируется из определенного количества строк, которые, в свою очередь, содержат определенное количество точек.

Важнейшей характеристикой качества растрового изображения является разрешающая способность.



Разрешающая способность растрового изображения определяется количеством точек по горизонтали и вертикали на единицу длины изображения.

При одних и тех же размерах экрана чем меньше размер точки, тем больше разрешающая способность (больше количество строк раstra и точек в строке), и, соответственно, выше качество изображения. Величина разрешающей способности обычно выражается в dpi (dot per inch — точек на дюйм), т. е. в количестве точек в полоске изображения длиной один дюйм (1 дюйм = 2,54 см).

В процессе дискретизации могут использоваться различные палитры цветов, т. е. наборы цветов, которые могут принимать точки изображения. Каждый цвет можно рассматривать как возможное состояние точки. Количество цветов N в палитре и количество информации I , необходимое для кодирования цвета каждой точки, связаны между собой и могут быть вычислены по формуле (2.1).



Количество информации, которое используется при кодировании цвета точек изображения, называется глубиной цвета.

Наиболее распространенными значениями глубины цвета при кодировании цветных изображений являются 4, 8, 16 или 24 бита на точку. Можно определить количество цветов в 24-битовой палитре:

$$N = 2^I = 2^{24} = 16\,777\,216.$$

Двоичное кодирование звуковой информации в компьютере

Звук представляет собой распространяющуюся в воздухе, воде или другой среде волну с непрерывно меняющейся интенсивностью и частотой.

Человек воспринимает звуковые волны (колебания воздуха) с помощью слуха в форме звуков различной громкости и тона, чем больше интенсивность звуковой волны, тем громче звук, чем больше частота волны, тем выше тон звука.

Для того чтобы компьютер мог обрабатывать реальный (записанный) звук, непрерывный звуковой сигнал должен быть преобразован в цифровую дискретную форму с помощью временной дискретизации. Непрерывная звуковая волна разбивается на отдельные маленькие временные участки, причем для каждого такого участка устанавливается определенная величина интенсивности звука.

Для записи аналогового звука и его преобразования в цифровую форму используется микрофон, подключенный к звуковой плате. Качество полученного цифрового звука зависит от количества измерений уровня громкости звука в единицу времени, т. е. частоты дискретизации. Чем большее количество измерений производится за 1 секунду (чем больше частота дискретизации), тем точнее «лесенка» цифрового звукового сигнала повторяет кривую аналогового сигнала.



Частота дискретизации звука — это количество измерений громкости звука за одну секунду.

Частота дискретизации звука может лежать в диапазоне от 8000 до 48 000 измерений громкости звука за одну секунду.

Каждому уровню дискретизации присваивается определенное значение уровня громкости звука. Уровни громкости

звука можно рассматривать как набор возможных состояний N , для кодирования которых необходимо определенное количество информации I , которое называется **глубиной кодирования звука**.



Глубина кодирования звука — это количество информации, которое необходимо для кодирования дискретных уровней громкости цифрового звука.

Если известна глубина кодирования, то количество уровней громкости цифрового звука можно рассчитать по формуле (2.1). Пусть глубина кодирования звука составляет 16 битов, тогда количество уровней громкости звука равно

$$N = 2^I = 2^{16} = 65\,536.$$

Чем больше частота дискретизации и глубина кодирования звука, тем более качественным будет звучание оцифрованного звука. Самое низкое качество оцифрованного звука, соответствующее качеству телефонной связи, будет при частоте дискретизации 8000 раз в секунду, глубине кодирования 8 битов и записи одной звуковой дорожки (режим моно). Высокое качество оцифрованного звука, соответствующее качеству аудио-CD, обеспечивается при частоте дискретизации 48 000 раз в секунду, глубине кодирования 16 битов и записи двух звуковых дорожек (режим стерео).

Контрольные вопросы

1. Чем различаются между собой кодировки русского алфавита?
2. Объясните, как разрешение и глубина цвета влияют на качество изображения.
3. Объясните, как частота дискретизации и глубина кодирования влияют на качество цифрового звука.



- 2.4. Представьте в форме десятичного кода слово «бит» в кодировках *Windows*, *MS-DOS*, *КОИ-8*, *Mac*, *ISO*. Воспользуйтесь *Windows-CD* для получения кодировочных таблиц.
- 2.5. Сканируется цветное изображение стандартного размера А4 (21×29,7 см). Разрешающая способность сканера 1200 dpi, глубина цвета 24 бита. Какой информационный объем будет иметь полученный графический файл?

- 2.6. Определить длительность звукового файла, который уместится на гибкой дискете 3,5" (для хранения данных на дискете выделяется 2847 секторов объемом 512 байтов каждый):
- при низком качестве звука: моно, 8 битов, 8 кГц;
 - при высоком качестве звука: стерео, 16 битов, 48 кГц.

2.6. Хранение информации

Информация, закодированная с помощью естественных и формальных языков, а также информация в форме зрительных и звуковых образов хранится в памяти человека. Однако для долговременного хранения информации, ее накопления и передачи из поколения в поколение используются **носители информации**.

Материальная природа носителей информации может быть различной: молекулы ДНК, которые хранят генетическую информацию; бумага, на которой хранятся тексты и изображения; магнитная лента, на которой хранится звуковая информация; фото- и киноплёнки, на которых хранится графическая информация; микросхемы памяти, магнитные и оптические диски, на которых хранятся программы и данные в компьютере, и т. д.

По оценкам специалистов, объем информации, фиксируемой на различных носителях, превышает один эксабайт в год (10^{18} байт/год). Примерно 80% всей этой информации хранится в цифровой форме на магнитных и оптических носителях и только 20% — на аналоговых носителях (бумага, магнитные ленты, фото- и киноплёнки). Если всю записанную в 2005 году информацию распределить на всех жителей планеты, то на каждого человека придется по 300 Мбайт.

Информационная емкость носителей информации. Носители информации характеризуются информационной емкостью, т. е. количеством информации, которое они могут хранить. Наиболее информационно емкими являются молекулы ДНК, которые имеют очень малый размер и плотно упакованы. Это позволяет хранить огромное количество информации (до 10^{21} бит в 1 см^3), что дает возможность организму развиваться из одной-единственной клетки, содержащей всю необходимую генетическую информацию.

Современные микросхемы памяти позволяют хранить в 1 см^3 до 10^{10} битов информации, однако это в 100 миллиардов раз меньше, чем в ДНК. Можно сказать, что современные технологии пока существенно проигрывают биологической эволюции.

Однако если сравнивать информационную емкость традиционных носителей информации (книг) и современных компьютерных носителей, то прогресс очевиден. На каждом гибком магнитном диске может храниться книга объемом около 600 страниц, а на жестком магнитном диске или DVD — целая библиотека, включающая десятки тысяч книг.

Надежность и долговременность хранения информации. Большое значение имеет надежность и долговременность хранения информации. Большую устойчивость к возможным повреждениям имеют молекулы ДНК, так как существует механизм обнаружения повреждений ее структуры (мутаций) и самовосстановления.

Надежность (устойчивость к повреждениям) достаточно высока у аналоговых носителей, повреждение которых приводит к потере информации только на поврежденном участке. Поврежденная часть фотографии не лишает возможности видеть оставшуюся часть, повреждение участка магнитной ленты приводит лишь к временному пропаданию звука и т. д.

Цифровые носители гораздо более чувствительны к повреждениям, даже потеря одного бита данных на магнитном или оптическом дисках может привести к невозможности считать файл, т. е. к потере большого объема данных. Именно поэтому необходимо соблюдать правила эксплуатации и хранения цифровых носителей информации.

Наиболее долговременным носителем информации является молекула ДНК, которая в течение десятков тысяч лет (человек) и миллионов лет (некоторые живые организмы) сохраняет генетическую информацию данного вида.

Аналоговые носители способны сохранять информацию в течение тысяч лет (египетские папирусы и шумерские глиняные таблички), сотен лет (бумага) и десятков лет (магнитная лента, фото- и киноплёнка).

Цифровые носители появились сравнительно недавно, и поэтому об их долговременности можно судить только по оценкам специалистов. По экспертным оценкам, при правильном хранении оптические носители способны хранить информацию сотни лет, а магнитные — десятки лет.

Контрольные вопросы

1. Какие достоинства и недостатки имеют аналоговые и цифровые носители информации?

2.7. Кодирование числовой информации. Системы счисления

«Всё есть число», — говорили древнегреческие философы, ученики Пифагора, подчеркивая необычайно важную роль чисел в практической деятельности. Известно множество способов представления чисел. Числа записываются с использованием особых знаковых систем, которые называются **системами счисления**. Алфавит систем счисления состоит из символов, которые называются цифрами.



Система счисления — это знаковая система, в которой числа записываются по определенным правилам с помощью символов некоторого алфавита, называемых цифрами.

Все системы счисления делятся на две большие группы: **непозиционные** и **позиционные**.

2.7.1. Непозиционные системы счисления

В непозиционных системах счисления количественный эквивалент каждой цифры не зависит от ее положения (места, позиции) в записи числа.

Единичная система счисления. Как только люди начали считать, у них появилась потребность в записи чисел. Находки археологов на стоянках первобытных людей свидетельствуют о том, что первоначально количество предметов отображали равным количеством каких-либо значков (бирок): зарубок, черточек, точек.

Позже, для облегчения счета, эти значки стали группировать по три или по пять. Такая система записи чисел называется **единичной (унарной)**, так как любое число в ней образуется путем повторения одного знака, символизирующего единицу. Отголоски единичной системы счисления встречаются и сегодня. Так, чтобы узнать, на каком курсе учится курсант военного училища, нужно сосчитать, какое количество полосок нашито на его рукаве.

Древнеегипетская непозиционная система счисления. Примерно в третьем тысячелетии до нашей эры древние египтяне придумали свою числовую систему, в которой для

обозначения ключевых чисел 1, 10, 100 и т. д. использовались специальные значки — иероглифы (рис. 2.15).

Все остальные числа составлялись из этих ключевых при помощи операции сложения. Система счисления Древнего Египта является десятичной, но непозиционной.

Например, чтобы изобразить число 3252, рисовали три цветка лотоса (три тысячи), два свернутых пальмовых листа (две сотни), пять дуг (пять десятков) и два шеста (две единицы). Величина числа не зависела от того, в каком порядке располагались составляющие его знаки: их можно было записывать сверху вниз, справа налево или в произвольном порядке.

Римская система счисления. Примером непозиционной системы, которая сохранилась до наших дней, может служить система счисления, которая применялась более двух с половиной тысяч лет назад в Древнем Риме. В основе римской системы счисления лежали знаки I (один палец) для числа 1, V (раскрытая ладонь) для числа 5, X (две сложенные ладони) для 10, а для обозначения чисел 100, 500 и 1000 стали применять первые буквы соответствующих латинских слов (Centum — сто, Demimille — половина тысячи, Mille — тысяча).

Чтобы записать число, римляне разлагали его на сумму тысяч, полутысяч, сотен, полусотен, десятков, пятков, единиц. Например, десятичное число 28 представляется следующим образом:

$$\text{XXVIII} = 10 + 10 + 5 + 1 + 1 + 1$$

(два десятка, пяток, три единицы).

Для записи чисел римляне использовали не только сложение, но и вычитание. При этом применялось следующее правило: каждый меньший знак, поставленный справа от большего, прибавляется к его значению, а каждый меньший знак, поставленный слева от большего, вычитается из него.

Например, IX — обозначает 9, XI — обозначает 11.

Десятичное число 99 имеет следующее представление:

$$\text{XCIX} = -10 + 100 - 1 + 10.$$

Практическое задание «Римская система счисления». Перевести числа из римской системы счисления в десятичную и из десятичной в римскую с использованием компьютерного калькулятора NumLock Calculator.

I	-	единицы
X	-	десятки
C	-	сотни
K	-	тысячи

Рис. 2.15. Алфавит древнеегипетской системы счисления



Римская система счисления

1. Запустить компьютерный калькулятор NumLock Calculator.
2. Установить минималистский вид калькулятора командой [*Вид калькулятора-Универсальный*].
3. Установить формат исходных данных командой [*Формат чисел в выражении-Римский*] и формат результата командой [*Формат результата-Десятичный*].
4. В окне ввода данных ввести римское число, например XXVIII, и нажать клавишу со знаком «=». Появится результат, число 28, записанное в десятичной системе счисления (рис. 2.16).

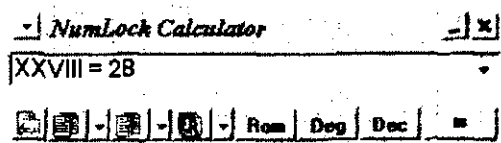


Рис. 2.16. Перевод из римской в десятичную систему счисления

5. Установить формат исходных данных командой [*Формат чисел в выражении-Десятичный*] и формат результата командой [*Формат результата-Римский*].
6. В окне ввода данных ввести десятичное число, например 99, и нажать клавишу со знаком «=». Появится результат, число XCIX, записанное в римской системе счисления (рис. 2.17).

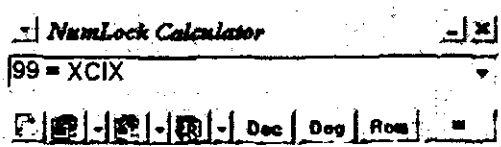


Рис. 2.17. Перевод из десятичной в римскую систему счисления

Римскими цифрами пользовались очень долго. Еще 200 лет назад в деловых бумагах числа должны были обозначаться римскими цифрами (считалось, что обычные арабские цифры легко подделать). Римская система счисления сегодня используется в основном для наименования знаменательных дат, томов, разделов и глав в книгах.

Алфавитные системы счисления. Более совершенными непозиционными системами счисления были алфавитные сис-

темы. К числу таких систем счисления относились греческая, славянская, финикийская и другие. В них числа от 1 до 9, десятки (от 10 до 90) и сотни (от 100 до 900) обозначались буквами алфавита.

В алфавитной системе счисления Древней Греции числа 1, 2, ..., 9 обозначались первыми девятью буквами греческого алфавита, например $\alpha = 1$, $\beta = 2$, $\gamma = 3$ и т. д. Для обозначения чисел 10, 20, ..., 90 применялись следующие 9 букв ($\iota = 10$, $\kappa = 20$, $\lambda = 30$, $\mu = 4$ и т. д.), а для обозначения чисел 100, 200, ..., 900 — последние 9 букв ($\rho = 100$, $\sigma = 200$, $\tau = 300$ и т. д.). Например, число 141 обозначалось $\rho\mu\alpha$.

У славянских народов числовые значения букв установились в порядке славянского алфавита, который использовал сначала глаголицу, а затем кириллицу (рис. 2.18).

Буквы кириллицы	Цифровое значение кириллицы	Буквы глаголицы	Цифровое значение глаголицы	Кириллическое название
А	1	ⴐ	1	Азь
Б		ⴑ	2	Буки
В	2	ⴒ	3	Вѣди
Г	3	ⴓ	4	Глаголь
Д	4	ⴔ	5	Добро
Е	5	ⴕ	6	Есть
Ж		ⴖ	7	Живѣте
З	6	ⴗ	8	Зѣло
З	7	ⴘ	9	Земля
И	8	ⴙ	10	Иже
І	10	ⴚ	20	І

Рис. 2.18. Древнерусская алфавитная система счисления

В России славянская нумерация сохранилась до конца XVII века. При Петре I возобладала так называемая арабская нумерация, которой мы пользуемся и сейчас. Славянская нумерация сохранилась только в богослужебных книгах.

Непозиционные системы счисления имеют ряд существенных недостатков:

1. Существует постоянная потребность введения новых знаков для записи больших чисел.

2. Невозможно представлять дробные и отрицательные числа.
3. Сложно выполнять арифметические операции, так как не существует алгоритмов их выполнения.

Контрольные вопросы

1. Существуют ли алгоритмы проведения арифметических действий в непозиционных системах счисления?

Задания

- 2.7. Некоторые римские цифры легко изобразить, используя палочки или спички. Ниже написано несколько неверных равенств. Как можно получить из них верные равенства, если разрешается переложить с одного места на другое только одну спичку (палочку)?

$$VII - V = XI$$

$$VI - IX = III$$

$$IX - V = VI$$

$$VIII - III = X$$

Компьютерный практикум



Windows-CD 

- 2.4. С помощью компьютерного калькулятора (например, NumLock Calculator) выполнить практическое задание «Римская система счисления».



2.7.2. Позиционные системы счисления

В позиционных системах счисления количественный эквивалент (значение) цифры зависит от ее места (позиции) в записи числа. Позиция цифры в числе называется **разрядом**. Разряд числа возрастает справа налево, от младших разрядов к старшим.

Основанием позиционной системы счисления называется целое число, которое равно количеству цифр, используемых для изображения чисел в данной системе счисления. Основание показывает, во сколько раз изменяется количественное значение цифры при перемещении ее в младший или старший разряд.

Позиционные системы с произвольным основанием. Возможно использование множества позиционных систем счисления, основание которых равно или больше 2. В системах счисления с основанием q (q -ичная система счисления) числа в развернутой форме записываются в виде суммы ряда степеней основания q с коэффициентами, в качестве которых выступают цифры $0, 1, q-1$. Для записи дробей используются разряды с отрицательными значениями степеней основания.

В развернутой форме число в системе счисления с основанием q (q -ичная система счисления) записывается следующим образом:

$$A_q = a_{n-1} \cdot q^{n-1} + a_{n-2} \cdot q^{n-2} + \dots + a_0 \cdot q^0 + a_{-1} \cdot q^{-1} + \dots + a_{-m} \cdot q^{-m} \quad (2.5)$$

или

$$A_q = \pm \sum_{i=-m}^{n-1} a_i q_i. \quad (2.6)$$

Здесь: A_q — число в q -ичной системе счисления,
 q — основание системы счисления,
 a_i — цифры, принадлежащие алфавиту данной системы счисления,
 n — число целых разрядов числа,
 m — число дробных разрядов числа.

Коэффициенты a_i в этой записи являются цифрами числа, записанного в q -ичной системе счисления. Свернутой формой записи числа называется запись в виде:

$$A = a_{n-1} a_{n-2} \dots a_1 a_0 a_{-1} \dots a_{-m}. \quad (2.7)$$

Свернутой формой записи чисел мы и пользуемся в повседневной жизни, ее называют естественной или цифровой.

Десятичная система счисления

Основание: $q = 10$.

Алфавит: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Число в десятичной системе счисления записывается в виде суммы числового ряда степеней основания (в данном случае 10), в качестве коэффициентов которых выступают цифры данного числа. В развернутой форме запись числа A_{10} , которое содержит n целых разрядов числа и m дробных разрядов числа, производится следующим образом:

$$A_{10} = a_{n-1} \cdot 10^{n-1} + a_{n-2} \cdot 10^{n-2} + \dots + a_0 \cdot 10^0 + a_{-1} \cdot 10^{-1} + \dots + a_{-m} \cdot 10^{-m}. \quad (2.8)$$

Коэффициенты a_i в этой записи являются цифрами десятичного числа, которое в свернутой форме записывается следующим образом:

$$A_{10} = a_{n-1}a_{n-2}\dots a_1a_0,a_{-1}\dots a_{-m}. \quad (2.9)$$

Например, десятичное число $555,55_{10}$ в развернутой форме будет записываться следующим образом:

$$555,55_{10} = 5 \cdot 10^2 + 5 \cdot 10^1 + 5 \cdot 10^0 + 5 \cdot 10^{-1} + 5 \cdot 10^{-2}.$$

Умножение или деление десятичного числа на 10 (величину основания) приводит к перемещению запятой, отделяющей целую часть от дробной на один разряд вправо или влево. Например:

$$\begin{aligned} 555,55_{10} \cdot 10 &= 5555,5_{10}; \\ 555,55_{10} : 10 &= 55,555_{10}. \end{aligned}$$

Двоичная система счисления

Основание: $q = 2$.

Алфавит: 0, 1.

Число в двоичной системе счисления записывается в виде суммы числового ряда степеней основания (в данном случае 2), в качестве коэффициентов которых выступают цифры данного числа. В развернутой форме запись числа A_2 , которое содержит n целых разрядов числа и m дробных разрядов числа, производится следующим образом:

$$A_2 = a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \dots + a_0 \cdot 2^0 + a_{-1} \cdot 2^{-1} + \dots + a_{-m} \cdot 2^{-m}. \quad (2.10)$$

Коэффициенты a_i в этой записи являются цифрами двоичного числа (0 или 1), которое в свернутой форме записывается следующим образом:

$$A_2 = a_{n-1}a_{n-2}\dots a_1a_0,a_{-1}\dots a_{-m}. \quad (2.11)$$

Умножение или деление двоичного числа на 2 (величину основания) приводит к перемещению запятой, отделяющей целую часть от дробной на один разряд вправо или влево. Например:

$$\begin{aligned} 101,01_2 \cdot 2 &= 1010,1_2; \\ 101,01_2 : 2 &= 10,101_2. \end{aligned}$$

Восьмеричная система счисления

Основание: $q = 8$.

Алфавит: 0, 1, 2, 3, 4, 5, 6, 7.

Число в восьмеричной системе счисления записывается в виде суммы числового ряда степеней основания (в данном

случае 8), в качестве коэффициентов которых выступают цифры данного числа. В развернутой форме запись числа A_8 , которое содержит n целых разрядов числа и m дробных разрядов числа, производится следующим образом:

$$A_8 = a_{n-1} \cdot 8^{n-1} + a_{n-2} \cdot 8^{n-2} + \dots + a_0 \cdot 8^0 + a_{-1} \cdot 8^{-1} + \dots + a_{-m} \cdot 8^{-m}. \quad (2.12)$$

Коэффициенты a_i в этой записи являются цифрами восьмеричного числа, которое в свернутой форме записывается следующим образом:

$$A_8 = a_{n-1}a_{n-2}\dots a_1a_0,a_{-1}\dots a_{-m}. \quad (2.13)$$

Умножение или деление восьмеричного числа на 8 (величину основания) приводит к перемещению запятой, отделяющей целую часть от дробной на один разряд вправо или влево. Например:

$$567,12_8 \cdot 8 = 5671,2_8;$$

$$567,12_8 : 8 = 56,712_8.$$

Шестнадцатеричная система счисления

Основание: $q = 16$.

Алфавит: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

Здесь только десять цифр из шестнадцати имеют общепринятое обозначение 0, 1, ..., 9. Для записи остальных цифр (10, 11, 12, 13, 14 и 15) обычно используются первые шесть букв латинского алфавита.

Число в шестнадцатеричной системе счисления записывается в виде суммы числового ряда степеней основания (в данном случае 16), в качестве коэффициентов которых выступают цифры данного числа. В развернутой форме запись числа A_{16} , которое содержит n целых разрядов числа и m дробных разрядов числа, производится следующим образом:

$$A_{16} = a_{n-1} \cdot 16^{n-1} + a_{n-2} \cdot 16^{n-2} + \dots + a_0 \cdot 16^0 + a_{-1} \cdot 16^{-1} + \dots + a_{-m} \cdot 16^{-m}. \quad (2.14)$$

Коэффициенты a_i в этой записи являются цифрами восьмеричного числа, которое в свернутой форме записывается следующим образом:

$$A_{16} = a_{n-1}a_{n-2}\dots a_1a_0,a_{-1}\dots a_{-m}. \quad (2.15)$$

Умножение или деление шестнадцатеричного числа на 16 (величину основания) приводит к перемещению запятой, отделяющей целую часть от дробной на один разряд вправо или влево. Например:

$$\begin{aligned} ABC,DE_{16} \cdot 16 &= ABCD,E_{16}; \\ ABC,DE_{16} : 16 &= AB,CDE_{16}. \end{aligned}$$

Контрольные вопросы

1. Чем отличаются позиционные системы счисления от непозиционных?
2. Какое количество цифр используется в q -ичной системе счисления?

Задачи

- 2.8. Во сколько раз увеличатся числа $10,1_{10}$, $10,1_2$, $64,5_8$, $39,F_{16}$ при переносе запятой на один знак вправо?
- 2.9. При переносе запятой на два знака вправо число $11,11_x$ увеличилось в 4 раза. Чему равен x ?

2.8. Перевод чисел из одной системы счисления в другую

2.8.1. Перевод целых чисел из десятичной системы в двоичную, восьмеричную и шестнадцатеричную

Перевод чисел из десятичной системы в двоичную, восьмеричную и шестнадцатеричную может осуществляться различными способами. Рассмотрим один из алгоритмов перевода на примере перевода чисел из десятичной системы в двоичную, при этом необходимо учитывать, что алгоритмы перевода целых чисел и правильных дробей будут различаться.

Вывод алгоритма перевода целых десятичных чисел в двоичную систему счисления

Пусть $A_{\text{цд}}$ — целое десятичное число, разложим его в ряд по основанию 2. Получим запись в развернутой форме двоичного целого числа, так как в этой записи будут отсутствовать отрицательные степени основания, т. е. числа 2:

$$A_{\text{цд}} = a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \dots + a_1 \cdot 2^1 + a_0 \cdot 2^0.$$

На первом шаге разделим число $A_{\text{цд}}$ на основание двоичной системы, т. е. на 2. Частное от деления будет равно:

$$a_{n-1} \cdot 2^{n-2} + a_{n-2} \cdot 2^{n-3} + \dots + a_1,$$

а остаток равен a_0 .

На втором шаге целое частное опять разделим на 2, остаток от деления будет теперь равен a_1 .

Если продолжать этот процесс деления, то после n -го шага получим последовательность остатков:

$$a_0, a_1, \dots, a_{n-1}.$$

Легко заметить, что их последовательность совпадает с обратной последовательностью цифр целого двоичного числа, записанного в свернутой форме:

$$A_{\text{цд}} = a_{n-1} \dots a_1 a_0.$$

Таким образом, достаточно записать остатки в обратной последовательности, чтобы получить искомое двоичное число.

Аналогичные рассуждения могут быть проведены и для перевода целых десятичных чисел в восьмеричную и шестнадцатеричную системы счисления.

Алгоритм перевода целых десятичных чисел в двоичную, восьмеричную и шестнадцатеричную системы счисления. Алгоритм перевода целого десятичного числа в целое двоичное, восьмеричное или шестнадцатеричное число будет следующим:

1. Последовательно выполнять деление исходного целого десятичного числа и получаемых целых частных на основание системы (на 2, 8 или 16) до тех пор, пока не получим частное, равное нулю.
2. Получить искомое двоичное, восьмеричное или шестнадцатеричное число, для чего записать полученные остатки в обратной последовательности.

Практическое задание «Перевод целого десятичного числа в целое двоичное, восьмеричное и шестнадцатеричное числа». Осуществить перевод десятичного числа (например, числа 19_{10}) в двоичную, восьмеричную и шестнадцатеричную системы счисления с помощью алгоритма перевода и с помощью компьютерного калькулятора NumLock Calculator.



**Перевод целого десятичного числа
в целые двоичное, восьмеричное
и шестнадцатеричное числа**

Применить алгоритм перевода целого десятичного числа в целое двоичное и записать результаты в таблицу.

Десятичное число/целое частное	Делитель (основание системы)	Остаток	Цифры двоичного числа
19	2	1	a_0 ▲
9	2	1	a_1
4	2	0	a_2
2	2	0	a_3
1	2	1	a_4
0			

В результате получаем двоичное число:

$$A_2 = a_4 a_3 a_2 a_1 a_0 = 10011_2.$$

Применить алгоритм перевода целого десятичного числа в целое восьмеричное и записать результаты в таблицу.

Десятичное число/целое частное	Делитель (основание системы)	Остаток	Цифры восьмеричного числа
19	8	3	a_0 ▲
2	8	2	a_1
0			

В результате получаем восьмеричное число:

$$A_8 = a_1 a_0 = 23_8.$$

Применить алгоритм перевода целого десятичного числа в целое шестнадцатеричное и записать результаты в таблицу.

Десятичное число/целое частное	Делитель (основание системы)	Остаток	Цифры шестнадцатеричного числа
19	16	3	a_0 ▲
1	16	1	a_1
0			

В результате получаем шестнадцатеричное число:

$$A_{16} = a_1 a_0 = 13_{16}.$$



**Перевод целого десятичного числа
в целые двоичное, восьмеричное
и шестнадцатеричное числа**

1. Запустить компьютерный калькулятор NumLock Calculator.
2. Установить минималистский вид калькулятора командой [*Вид калькулятора-Универсальный*].
3. Установить формат исходных данных командой [*Формат чисел в выражении-Десятичный*] и формат результата командой [*Формат результата-Двоичный*].
4. В окне ввода данных ввести десятичное число, например 19, и нажать клавишу со знаком «=». Появится результат — двоичное число 10011 (рис. 2.19).

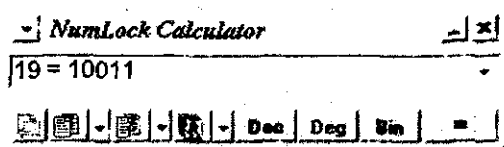


Рис. 2.19. Перевод из десятичной системы счисления в двоичную

5. Установить формат исходных данных командой [*Формат чисел в выражении-Десятичный*] и формат результата командой [*Формат результата-Восьмеричный*].
6. В окне ввода данных ввести десятичное число, например 19, и нажать клавишу со знаком «=». Появится результат — восьмеричное число 23 (рис. 2.20).

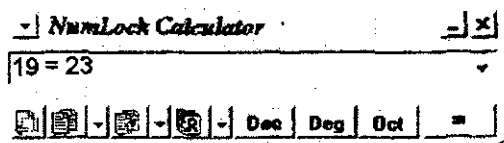


Рис. 2.20. Перевод из десятичной системы счисления в восьмеричную

7. Установить формат исходных данных командой [*Формат чисел в выражении-Десятичный*] и формат результата командой [*Формат результата-Шестнадцатеричный*].
8. В окне ввода данных ввести десятичное число, например 19, и нажать клавишу со знаком «=». Появится результат — шестнадцатеричное число 13 (рис. 2.21).

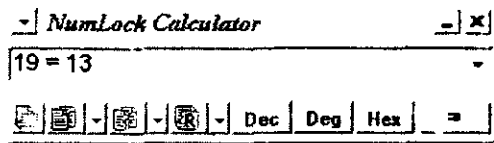


Рис. 2.21. Перевод из десятичной системы счисления в шестнадцатеричную

Контрольные вопросы

1. Сформулируйте алгоритм перевода целых десятичных чисел в двоичную, восьмеричную и шестнадцатеричную системы счисления.

Задания

- 2.10. Заполните таблицу, в каждой строке которой одно и то же целое число должно быть записано в различных системах счисления:

Двоичная	Восьмеричная	Десятичная	Шестнадцатеричная
101010			
	127		
		269	
			9B

Компьютерный практикум

Windows-CD 

- 2.5. Выполнить практическое задание «Перевод целого десятичного числа в целые двоичное, восьмеричное и шестнадцатеричное числа» (для нескольких чисел).



2.8.2. Перевод дробей из десятичной системы в двоичную, восьмеричную и шестнадцатеричную

Вывод алгоритма перевода десятичных дробей в двоичную систему счисления

Пусть $A_{дл}$ — десятичная дробь, разложим ее в ряд по основанию 2. Получим запись в развернутой форме двоичной

дроби, так как в этой записи будут отсутствовать положительные степени основания, т. е. числа 2:

$$A_{\text{дд}} = a_{-1} \cdot 2^{-1} + a_{-2} \cdot 2^{-2} + \dots$$

На первом шаге умножим число $A_{\text{дд}}$ на основание двоичной системы, т. е. на 2. Произведение будет равно:

$$a_{-1} + a_{-2} \cdot 2^{-1} + \dots$$

Целая часть равна a_{-1} , именно это число и является значением первого дробного разряда двоичного числа.

На втором шаге оставшуюся дробную часть опять умножим на 2, получим целую часть, равную a_{-2} , именно это число и является значением второго дробного разряда двоичного числа.

Описанный процесс необходимо продолжать до тех пор, пока в результате умножения мы не получим нулевую дробную часть или не будет достигнута требуемая точность вычислений.

Легко заметить, что последовательность полученных чисел совпадает с последовательностью цифр дробного двоичного числа, записанного в свернутой форме:

$$A_{\text{дд}} = a_{-1} a_{-2} \dots$$

Аналогичные рассуждения могут быть проведены и для перевода дробных десятичных чисел в восьмеричную и шестнадцатеричную системы счисления.

Алгоритм перевода дробных десятичных чисел в двоичную, восьмеричную и шестнадцатеричную системы счисления. Алгоритм перевода десятичной дроби в двоичную, восьмеричную или шестнадцатеричную дробь будет следующим:

1. Последовательно выполнять умножение исходной десятичной дроби и получаемых дробей на основание системы счисления (на 2, 8 или 16) до тех пор, пока не получим нулевую дробную часть или не будет достигнута требуемая точность вычислений.
2. Получить искомую двоичную, восьмеричную или шестнадцатеричную дробь, записав полученные целые части произведения в прямой последовательности.

Задание «Перевод десятичной дроби в двоичную, восьмеричную и шестнадцатеричную дроби». Осуществить перевод десятичной дроби (например, числа $0,65625_{10}$) в двоичную, восьмеричную и шестнадцатеричную системы счисления с помощью алгоритма перевода.



Перевод десятичной дроби в двоичную, восьмеричную и шестнадцатеричную дроби

Применить алгоритм перевода дробного десятичного числа в дробное двоичное и записать результаты в таблицу.

Десятичная дробь/ дробная часть произведения	Множитель (основание системы)	Целая часть произведения	Цифры двоичной дроби
0,65625	2	1	a_{-1}
0,3125	2	0	a_{-2}
0,625	2	1	a_{-3}
0,25	2	0	a_{-4}
0,5	2	1	a_{-5} ▼
0,0			

В результате получаем двоичную дробь:

$$A_2 = 0, a_{-1} a_{-2} a_{-3} a_{-4} a_{-5} = 0,10101_2.$$

Применить алгоритм перевода дробного десятичного числа в дробное восьмеричное и записать результаты в таблицу.

Десятичная дробь/ дробная часть произведения	Множитель (основание системы)	Целая часть произведения	Цифры восьмерич- ной дроби
0,65625	8	5	a_{-1}
0,25	8	2	a_{-2} ▼
0,0			

В результате получаем восьмеричную дробь:

$$A_8 = 0, a_{-1} a_{-2} = 0,52_8.$$

Применить алгоритм перевода дробного десятичного числа в дробное шестнадцатеричное и записать результаты в таблицу.

Десятичная дробь/ дробная часть произведения	Множитель (основание системы)	Целая часть произ- ведения	Цифры шест- надцатерич- ной дроби
0,65625	16	10 (A)	a_{-1}
0,5	16	8	a_{-2} ▼
0,0			

В результате получаем шестнадцатеричную дробь:

$$A_{16} = 0, a_{-1} a_{-2} = 0, A8_{16}.$$

Перевод чисел, содержащих и целую, и дробную части, производится в два этапа. Отдельно переводится по соответствующему алгоритму целая часть и отдельно — дробная. В итоговой записи полученного числа целая часть от дробной отделяется запятой.

Контрольные вопросы

1. Сформулируйте алгоритм перевода целых десятичных чисел в двоичную, восьмеричную и шестнадцатеричную системы счисления.

Задания

- 2.11. Выполнить задание «Перевод десятичной дроби в двоичную, восьмеричную и шестнадцатеричную дроби» (для нескольких дробей).
- 2.12. Заполнить таблицу, в каждой строке которой одно и то же дробное число должно быть записано в различных системах счисления:

Двоичная	Восьмеричная	Десятичная	Шестнадцатеричная
0,101			
	0,6		
		0,125	
			0,4

2.8.3. Перевод чисел из двоичной системы в восьмеричную и шестнадцатеричную и обратно

Перевод чисел между системами счисления, основания которых являются степенями числа 2 ($q = 2^n$), может производиться по более простым алгоритмам. Такие алгоритмы могут применяться для перевода чисел между двоичной ($q = 2^1$), восьмеричной ($q = 2^3$) и шестнадцатеричной ($q = 2^4$) системами счисления.

Перевод чисел из двоичной системы счисления в восьмеричную

Для записи двоичных чисел используются две цифры, т. е. в каждом разряде числа возможны два варианта записи.

Для определения количества информации, которое содержит один двоичный разряд, воспользуемся формулой (2.2). Решаем показательное уравнение:

$$2 = 2^I, \text{ так как } 2 = 2^1, \text{ то } I = 1 \text{ бит.}$$

Каждый разряд двоичного числа содержит 1 бит.

Для записи восьмеричных чисел используются восемь цифр, т. е. в каждом разряде числа возможны восемь вариантов записи. Решаем показательное уравнение:

$$8 = 2^I, \text{ так как } 8 = 2^3, \text{ то } I = 3 \text{ бита.}$$

Каждый разряд восьмеричного числа содержит 3 бита. Таким образом, для перевода двоичного числа в восьмеричное двоичное число нужно разбить на группы по три цифры. Для упрощения перевода можно заранее подготовить таблицу преобразования двоичных триад (групп по 3 цифры) в восьмеричные цифры (табл. 2.1).

Таблица 2.2. Двоичные триады

Двоичные триады	000	001	010	110	100	101	110	111
Шестнадцатеричные цифры	0	1	2	3	4	5	6	7

Перевод целых чисел. Для перевода целого двоичного числа в восьмеричное двоичное число нужно разбить на группы по три цифры, справа налево; если в последней левой группе окажется меньше чем три разряда, то необходимо ее дополнить слева нулями. Затем надо преобразовать каждую группу в восьмеричную цифру.

Переведем таким способом двоичное число 101001_2 в восьмеричное число. Получаем:

$$101\ 001_2 = 51_8.$$

Перевод дробей. Для перевода дробного двоичного числа в восьмеричное необходимо разбить двоичное число на триады слева направо; если в последней правой группе окажется меньше разрядов, надо дополнить ее справа нулями. Далее следует триады заменить на восьмеричные числа.

Например, преобразуем дробное двоичное число $A_2 = 0,110101_2$ в восьмеричную систему счисления. Получаем:

$$0,110101_2 = 0,35_8.$$

Например, преобразуем дробное двоичное число $A_2 = 0,110101_2$ в восьмеричную систему счисления. Получаем:

$$0,110101_2 = 0,35_8.$$

Перевод чисел из двоичной системы счисления в шестнадцатеричную

Для записи шестнадцатеричных чисел используются шестнадцать цифр, т. е. в каждом разряде числа возможны шестнадцать вариантов записи. Решаем показательное уравнение:

$$16 = 2^I, \text{ так как } 16 = 2^4, \text{ то } I = 4 \text{ бита.}$$

Каждый разряд шестнадцатеричного числа содержит 4 бита. Таким образом, для перевода двоичного числа в шестнадцатеричное двоичное число нужно разбить на группы по четыре цифры. Для упрощения перевода можно заранее подготовить таблицу преобразования двоичных тетрад (групп по 4 цифры) в шестнадцатеричные цифры (табл. 2.3).

Таблица 2.3. Двоичные тетрады

Двоичные тетрады	0000	0001	0010	0110	0100	0101	0110	0111
Шестнадцатеричные цифры	0	1	2	3	4	5	6	7
Двоичные тетрады	1000	1001	1010	1011	1100	1101	1110	1111
Шестнадцатеричные цифры	8	9	A	B	C	D	E	F

Перевод целых чисел. Для перевода целого двоичного числа в шестнадцатеричное его нужно разбить на группы по четыре цифры (тетрады), начиная справа; если в последней левой группе окажется меньше разрядов, надо дополнить ее слева нулями.

Переведем целое двоичное число $A_2 = 101001_2$ в шестнадцатеричное:

$$0010\ 1001_2 = 29_{16}.$$

Перевод дробей. Для перевода дробного двоичного числа в шестнадцатеричное необходимо разбить его на тетрады слева направо; если в последней правой группе окажется меньше чем четыре разряда, необходимо ее дополнить справа нулями.

Переведем дробное двоичное число $A_2 = 0,110101_2$ в шестнадцатеричную систему счисления:

$$0,1101\ 0100_2 = 0,D4_{16}.$$

Для того чтобы преобразовать любое двоичное число в восьмеричную или шестнадцатеричную системы счисления, не-

обходимо произвести преобразования по рассмотренным выше алгоритмам отдельно для его целой и дробной частей.

Перевод чисел из восьмеричной и шестнадцатеричной систем счисления в двоичную

Для перевода чисел из восьмеричной и шестнадцатеричной систем счисления в двоичную необходимо цифры числа преобразовать в группы двоичных чисел. Для перевода из восьмеричной системы в двоичную каждую цифру числа надо преобразовать в группу из трех двоичных разрядов (триаду), а при преобразовании шестнадцатеричного числа — в группу из четырех разрядов (тетраду).

Например, преобразуем дробное восьмеричное число $A_8 = 0,47_8$ в двоичную систему счисления:

$$0,47_8 = 0,100111_2.$$

Переведем целое шестнадцатеричное число $A_{16} = AB_{16}$ в двоичную систему счисления:

$$AB_{16} = 10101011_2.$$

Контрольные вопросы

1. Сформулируйте алгоритм перевода целых двоичных чисел в восьмеричную и шестнадцатеричную системы счисления.
2. Сформулируйте алгоритм перевода дробных двоичных чисел в восьмеричную и шестнадцатеричную системы счисления.

Задания

- 2.13. Заполнить таблицу, в каждой строке которой одно и то же произвольное число (число может содержать как целую, так и дробную часть) должно быть записано в различных системах счисления:

Двоичная	Восьмеричная	Шестнадцатеричная
111101,1		
	233,5	
		59,B

2.9. Арифметические операции в позиционных системах счисления

Арифметические операции во всех позиционных системах счисления выполняются по одним и тем же хорошо известным правилам.

Сложение. Рассмотрим сложение чисел в двоичной системе счисления. В его основе лежит таблица сложения одноразрядных двоичных чисел:

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= 10 \end{aligned}$$

Важно обратить внимание на то, что при сложении двух единиц происходит переполнение разряда и производится перенос в старший разряд. Переполнение разряда наступает тогда, когда величина числа в нем становится равной или большей основания системы счисления. Для двоичной системы счисления эта величина равна двум.

Сложение многоразрядных двоичных чисел происходит в соответствии с вышеприведенной таблицей сложения с учетом возможных переносов из младших разрядов в старшие. В качестве примера сложим в столбик двоичные числа 110_2 и 11_2 .

$$\begin{array}{r} 110_2 \\ + 11_2 \\ \hline 1001_2 \end{array}$$

Проверим правильность вычислений сложением в десятичной системе счисления. Переведем двоичные числа в десятичную систему счисления и затем их сложим.

$$110_2 = 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 6_{10};$$

$$11_2 = 1 \cdot 2^1 + 1 \cdot 2^0 = 3_{10};$$

$$6_{10} + 3_{10} = 9_{10}.$$

Теперь переведем результат двоичного сложения в десятичное число.

$$1001_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 9_{10}.$$

Сравним результаты, сложение выполнено правильно.

Вычитание. Рассмотрим вычитание двоичных чисел. В его основе лежит таблица вычитания одноразрядных двоичных чисел. При вычитании из меньшего числа (0) большего (1) производится заем из старшего разряда. В таблице заем обозначен 1 с чертой.

$$\begin{aligned} 0 - 0 &= \underline{0} \\ 0 - 1 &= \underline{11} \\ 1 - 0 &= \underline{1} \\ 1 - 1 &= \underline{0} \end{aligned}$$

Вычитание многоразрядных двоичных чисел происходит в соответствии с вышеприведенной таблицей вычитания с учетом возможных заемов в старших разрядах. В качестве примера произведем вычитание двоичных чисел 110_2 и 11_2 .

$$\begin{array}{r} 110_2 \\ - 11_2 \\ \hline 11_2 \end{array}$$

Умножение. В основе умножения лежит таблица умножения одноразрядных двоичных чисел:

$$\begin{aligned} 0 \cdot 0 &= 0 \\ 0 \cdot 1 &= 0 \\ 1 \cdot 0 &= 0 \\ 1 \cdot 1 &= 1 \end{aligned}$$

Умножение многоразрядных двоичных чисел происходит в соответствии с вышеприведенной таблицей умножения по обычной схеме, применяемой в десятичной системе счисления, с последовательным умножением множимого на очередную цифру множителя. В качестве примера произведем умножение двоичных чисел 110_2 и 11_2 .

$$\begin{array}{r} 110_2 \\ \times 11_2 \\ \hline 110 \\ 110 \\ \hline 10010_2 \end{array}$$

Деление. Операция деления выполняется по алгоритму, подобному алгоритму выполнения операции деления в десятичной системе счисления. В качестве примера произведем деление двоичного числа 110_2 на 11_2 .

$$\begin{array}{r} 110_2 : 11_2 = \\ \underline{11} \\ 0 \end{array}$$

Арифметические операции в восьмеричной и шестнадцатеричной системах счисления. Аналогично можно выполнять арифметические действия в восьмеричной и шестнадцатерич-

ной системах счисления. Необходимо только помнить, что перенос в следующий разряд при сложении и заем из старшего разряда при вычитании определяется величиной основания системы счисления:

$$\begin{array}{r} 37_8 \\ + 25_8 \\ \hline 64_8 \end{array} \qquad \begin{array}{r} 9C_{16} \\ - 78_{16} \\ \hline 24_{16} \end{array}$$

Для проведения арифметических операций над числами, выраженными в различных системах счисления, необходимо предварительно перевести их в одну и ту же систему.

Практическое задание «Арифметические операции в позиционных системах счисления». Выполнить сложение, вычитание, умножение и деление чисел, выраженных в различных системах счисления (например, 12_8 и 64_{16}), с помощью компьютерного калькулятора Wise Calculator.



Арифметические операции в позиционных системах счисления

1. Запустить Wise Calculator и ввести команду [*Tools-Multi-Base Calculator...*].

В режиме *Multi-Base Calculator* появляется многооконная панель, позволяющая ввести число в любой системе счисления и автоматически получить значения этого числа в других системах счисления. В режиме *Multi-Base Calculator* возможно проведение арифметических операций (сложение, вычитание, умножение и деление) в различных системах счисления. Для этого в окна операндов *A* и *B* необходимо ввести числа (можно в различных системах счисления) и выбрать в группе переключателей *Operation* арифметическую операцию.

2. На появившейся многооконной панели *Multi-Base Calculator* (рис. 2.22) ввести пару чисел в окна операндов *A* и *B*. Например, шестнадцатеричное число 64_{16} в окно *HEX* (операнд *A*) и восьмеричное число 12_8 в окно *OCT* (операнд *B*).
3. Выбрать в группе переключателей *Operation*, например операцию умножения $A*B$.
4. В окнах *Result* появится результат выполнения арифметической операции одновременно в четырех системах счисления: *DEC* (1000_{10}), *BIN* (1111101000_2), *OCT* (1750_8) и *HEX* ($3E8_{16}$).

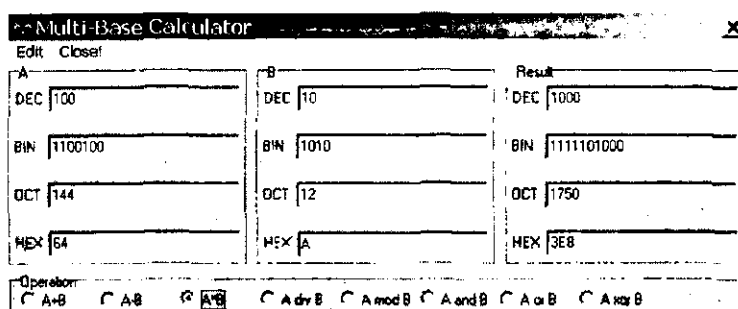



Рис. 2.22. Умножение чисел в различных системах счисления

5. Последовательно выбрать в группе переключателей *Operation* операции сложения $A+B$, вычитания $A-B$ и деления $A \div B$.

Компьютерный практикум



Windows-CD 

- 2.6. Выполнить с использованием компьютерного калькулятора Wise Calculator практическое задание «Арифметические операции в позиционных системах счисления».



2.10. Представление чисел в компьютере

2.10.1. Представление чисел в формате с фиксированной запятой

Целые числа в компьютере хранятся в памяти в формате с **фиксированной запятой**. В этом случае каждому разряду ячейки памяти соответствует всегда один и тот же разряд числа, а запятая находится справа после младшего разряда, т. е. вне разрядной сетки.

Достоинствами представления чисел в формате с фиксированной запятой являются простота и наглядность представления чисел, а также простота алгоритмов реализации арифметических операций (вычитание благодаря использованию дополнительного кода (см. ниже) для представления отрицательных чисел сводится к сложению).

Недостатком представления чисел в формате с фиксированной запятой является конечный диапазон представления

величин, недостаточный для решения математических, физических, экономических и других задач, в которых используются как очень малые, так и очень большие числа.

Целые неотрицательные числа. Для хранения целых неотрицательных чисел отводится одна ячейка памяти (8 битов). Например, число $A_2 = 10101010_2$ будет храниться в ячейке памяти следующим образом:

1	0	1	0	1	0	1	0
---	---	---	---	---	---	---	---

Минимальное число соответствует восьми нулям, хранящимся в восьми ячейках памяти, и равно нулю.

Максимальное значение целого неотрицательного числа достигается в случае, когда во всех ячейках хранятся единицы. Для n -разрядного представления оно будет равно

$$2^n - 1.$$

Задание «Диапазон хранения целых неотрицательных чисел». Определить диапазон чисел, которые могут храниться в оперативной памяти в формате *целого неотрицательного числа*.

Минимальное число равно 0.

Максимальное число соответствует восьми единицам, хранящимся в ячейках памяти, и равно:

$$A = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 1 \cdot 2^8 - 1 = 255_{10}.$$

Диапазон изменения целых неотрицательных чисел — от 0 до 255.

Целые числа со знаком. Для хранения *целых чисел со знаком* отводится две ячейки памяти (16 битов), а для хранения *больших целых чисел со знаком* отводится четыре ячейки памяти (32 бита). Старший (левый) разряд отводится под знак числа (если число положительное, то в знаковый разряд записывается 0, если число отрицательное — записывается 1).

Представление в компьютере положительных чисел с использованием формата «знак-величина» называется **прямым кодом числа**. Например, число $2002_{10} = 11111010010_2$ будет представлено в 16-разрядном представлении следующим образом:

0	0	0	0	0	1	1	1	1	1	0	1	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

При представлении целых чисел в n -разрядном представлении со знаком максимальное положительное число (с учетом выделения одного разряда на знак) равно

$$A = 2^{n-1} - 1.$$

Минимальное отрицательное число равно

$$A = -2^{n-1}.$$

Задание «Диапазон хранения целых чисел со знаком и больших целых чисел со знаком». Определить диапазон чисел, которые могут храниться в оперативной памяти в формате:

- *целых чисел со знаком;*
- *больших целых чисел со знаком.*

а) Максимальное положительное целое число в формате *целых чисел со знаком* равно:

$$A_{10} = 2^{15} - 1 = 32\,767_{10}.$$

Минимальное отрицательное целое число равно

$$A_{10} = -2^{15} = -32\,768_{10}.$$

б) Максимальное положительное целое число в формате *больших целых чисел со знаком* равно

$$A_{10} = 2^{31} - 1 = 2\,147\,483\,647_{10}.$$

Минимальное отрицательное целое число равно

$$A_{10} = -2^{31} = -2\,147\,483\,648_{10}.$$

Дополнительный код. Для представления отрицательных чисел используется дополнительный код. Дополнительный код позволяет заменить арифметическую операцию вычитания операцией сложения, что существенно упрощает работу процессора и увеличивает его быстродействие.



Дополнительный код отрицательного числа A , хранящегося в n ячейках, равен $2^n - |A|$.

Дополнительный код представляет собой дополнение модуля отрицательного числа A до 0, поэтому в n -разрядной компьютерной арифметике:

$$2^n - |A| + |A| = 0.$$

Это равенство тождественно справедливо, так как в компьютерной n -разрядной арифметике $2^n \equiv 0$. Действительно, двоичная запись такого числа состоит из одной единицы и n нулей, а в n -разрядную ячейку может поместиться только n младших разрядов, т. е. n нулей.

Задание «Дополнительный код числа». Записать дополнительный код отрицательного числа -2002 для 16-разрядного компьютерного представления.

Проведем вычисления в соответствии с определением дополнительного кода:

2^{16}	=	1000000000000000_2	65536_{10}
2002_{10}	=	0000011111010010_2	2002_{10}
$2^{16} - 2002_{10} $	=	1111100000101110_2	63534_{10}

Проведем проверку с использованием десятичной системы счисления. Дополнительный код $63\ 534_{10}$ в сумме с модулем отрицательного числа 2002_{10} равен $65\ 536_{10}$, т. е. дополнительный код дополняет модуль отрицательного числа до 2^{16} (до нуля 16-разрядной компьютерной арифметики).

Алгоритм получения дополнительного кода. Для получения дополнительного кода отрицательного числа можно использовать довольно простой алгоритм:

1. Модуль числа записать прямым кодом в n двоичных разрядах.
2. Получить **обратный код** числа, для этого значения всех битов инвертировать (все единицы заменить на нули и все нули заменить на единицы).
3. К полученному обратному коду прибавить единицу.

Задание «Вычисление дополнительного кода числа с использованием обратного кода». Вычислить дополнительный код отрицательного числа -2002 для 16-разрядного компьютерного представления с использованием алгоритма.

Прямой код	$ -2002_{10} $	0000011111010010_2
Обратный код	Инвертирование	1111100000101101_2
	Прибавление единицы	1111100000101101_2 + 0000000000000001_2
Дополнительный код		1111100000101110_2

Задание «Арифметическое действие». Выполнить арифметическое действие $3000_{10} - 5000_{10}$ в 16-разрядном компьютерном представлении.

Представим положительное число в прямом, а отрицательное число — в дополнительном коде:

Десятичное число	Прямой код	Обратный код	Дополнительный код
3000	0000101110111000		
-5000	0001001110001000	1110110001110111	+ 1110110001110111 0000000000000001 1110110001110000

Сложим прямой код положительного числа с дополнительным кодом отрицательного числа. Получим результат в дополнительном коде:

3000-5000			1111100000110000
-----------	--	--	------------------

Переведем полученный дополнительный код в десятичное число:

- 1) Инвертируем дополнительный код: 0000011111001111.
- 2) Прибавим к полученному коду 1 и получим модуль отрицательного числа:

$$\begin{array}{r}
 0000011111001111 \\
 + \quad 0000000000000001 \\
 \hline
 0000011111010000
 \end{array}$$

- 3) Переведем в десятичное число и припишем знак отрицательного числа: -2000.

Контрольные вопросы

1. Чем различаются обратный и дополнительный коды числа в компьютерном представлении отрицательных чисел?

Задачи

- 2.14. Заполнить таблицу, записав отрицательные десятичные числа в прямом, обратном и дополнительном кодах в 16-разрядном представлении:

Десятичные числа	Обратный код	Дополнительный код
-10		
-100		
-1000		
-10000		

2.15. Выполнить арифметическое действие $20_{10} - 60_{10}$ в 16-разрядном компьютерном представлении.

2.10.2. Представление чисел в формате с плавающей запятой

Вещественные числа (конечные и бесконечные десятичные дроби) хранятся и обрабатываются в компьютере в формате с плавающей запятой. В этом случае положение запятой в записи числа может изменяться.

Формат чисел с плавающей запятой базируется на экспоненциальной форме записи, в которой может быть представлено любое число. Так число A может быть представлено в виде:

$$A = m \times q^n, \quad (2.16)$$

где m — мантисса числа,

q — основание системы счисления,

n — порядок числа.

Для однозначности представления чисел с плавающей запятой используется нормализованная форма, при которой мантисса отвечает условию:

$$1/n \leq |m| < 1.$$

Это означает, что мантисса должна быть правильной дробью и иметь после запятой цифру, отличную от нуля.

Задание «Приведение числа с плавающей запятой к нормализованной форме». Преобразовать десятичное число 888,888, записанное в естественной форме, в экспоненциальную форму с нормализованной мантиссой.

$$888,888 = 0,888888 \times 10^3.$$

Нормализованная мантисса $m = 0,888888$, порядок $n = 3$.

Число в форме с плавающей запятой занимает в памяти компьютера четыре байта (*число обычной точности*) или восемь байтов (*число двойной точности*). При записи числа с плавающей запятой выделяются разряды для хранения знака мантиссы, знака порядка, порядка и мантиссы.

Диапазон изменения чисел определяется количеством разрядов, отведенных для хранения порядка числа, а точность (количество значащих цифр) определяется количеством разрядов, отведенных для хранения мантиссы.

Задание «Определение максимального числа и его точности». Определить максимальное число и его точность для формата чисел *обычной точности*, если для хранения порядка и

