

Г. А. Заборовский А. Е. Пупцев

ИНФОРМАТИКА

10

класс



Г. А. Заборовский А. Е. Пупцев

ИНФОРМАТИКА

**Учебное пособие для 10 класса
учреждений общего среднего образования
с русским языком обучения**

*Допущено Министерством образования
Республики Беларусь*

Минск
«Издательский центр БГУ»
2011

Правообладатель Издательский центр БГУ

УДК 004(075.3=161.1)

ББК 32.81я721

З-12

А в т о р ы:

Г. А. Заборовский — «От авторов», гл. 1, 2, 3, 5;

А. Е. Пупцев — гл. 1, 2, 4, 6

Р е ц е н з е н т ы:

кафедра экономической информатики учреждения образования
«Белорусский государственный аграрный технический университет»
(кандидат педагогических наук, доцент, зав. кафедрой *О. Л. Сапун*);
учитель информатики высшей категории
государственного учреждения образования «Гимназия № 2 г. Минска»
О. Н. Лапко

ISBN 978-985-476-907-3

© Заборовский Г. А., Пупцев А. Е., 2011

© Оформление. РУП «Издательский центр БГУ», 2011

Правообладатель Издательский центр БГУ

ОТ АВТОРОВ

Уважаемые школьники!

В десятом классе вы продолжите изучение информатики. Умение применять компьютер и современное программное обеспечение так же важно для человека, как умение читать и писать.

В первой главе пособия вы познакомитесь с методами хранения информации и инструментами архивации.

Во второй главе вы изучите возможности совместного использования цифровых устройств и компьютера.

В третьей главе будет продолжено изучение основ алгоритмизации и программирования. Вы изучите процедуры и функции, используемые при работе с символьными и строковыми величинами, познакомитесь с алгоритмами обработки текстовой информации.

Четвертая глава познакомит вас с информационными моделями и структурированием информации с использованием информационных моделей.

В пятой главе вы изучите инструменты и методы обработки информации в электронных таблицах, научитесь выполнять вычисления и представлять результаты в виде графиков и диаграмм. Полученные знания и умения вы сможете применять для решения практических задач из различных предметных областей.

Шестая глава продолжит знакомить вас с использованием компьютерных коммуникаций и Интернета. Вы рассмотрите современные сервисы: форумы, блоги, чаты, социальные сети.

Вопросы, отмеченные значком , помогут вам закрепить материал. Дополнительный материал для любознательных отмечен значком



После параграфов предлагаются упражнения, которые позволят вам закрепить свои знания и практические умения работы на компьютере.

Желаем успехов в изучении информатики и информационных технологий. Хорошие знания помогут вам выбрать профессию и принести пользу Родине.

Правообладатель Издательский центр БГУ

ХРАНЕНИЕ ИНФОРМАЦИИ

§ 1. АРХИВАЦИЯ ДАННЫХ

Одной из проблем, с которой приходится сталкиваться пользователю компьютера, является нехватка памяти для долговременного хранения информации. Каким бы большим ни был объем памяти жесткого диска, он может оказаться недостаточным для установки новой игры или хранения фильмов. Еще одна проблема — передача большого объема данных по компьютерной сети.

Уменьшить объем сохраняемых на диске или передаваемых по сети данных можно путем их архивации. *Архивацией* (сжатием, компрессией) называют преобразование данных в сжатую (архивную) форму.



Каким же образом осуществляется сжатие данных? Рассмотрим простейший алгоритм на примере сжатия текста.

Пусть строка содержит 60 повторяющихся букв (20 *a*, 20 *b*, 20 *c*):

aaaaaaaaaaaaaaaaaaaaaaaaabbbbbbbbbbbbbbbbbbbcccccccccccccccccccc

Для кодирования каждой буквы нужен 1 байт, поэтому строка без сжатия занимает $1 \text{ байт} \cdot 60 = 60 \text{ байт}$. Если для описания количества повторяющихся букв использовать 1 байт, то объем сжатой строки составит $2 + 2 + 2 = 6 \text{ байт}$. Получили уменьшение объема в 10 раз.

Если же в строке мало или нет повторяющихся групп символов, то может получиться не уменьшение, а увеличение объема.

Для сжатия данных и размещения их в архивах используют программы-*архиваторы*. Архивный файл (или просто архив) хранит в сжатом виде файлы и папки, которые при необходимости могут быть извлечены из архива в первоначальном виде.

Наиболее распространенными архиваторами для операционной системы Windows являются WinRAR и WinZip. Их архивные файлы имеют расширения *.rar* или *.zip* соответственно и обозначаются значками  (*rar*) и  (*zip*).

Данные в архиве хранятся в закодированном виде, поэтому непосредственно работать с ними нельзя. Для работы с файлом его нужно извлечь из архива. Извлекают файлы из архива с помощью того же архиватора, который использовался для создания архива. Но можно использовать и другие архиваторы, распознающие данный тип архива. Так, программа WinRAR может работать и с архивами .zip.

Интерфейсы программ-архиваторов достаточно просты, чтобы пользователь смог быстро освоить основные приемы работы.

Рассмотрим подробнее интерфейс архиватора WinRAR. После его запуска открывается окно, основные элементы которого представлены на рис. 1.1. С помощью меню **Файл** или адресной строки выбирается требуемый диск и папка. Содержащиеся в ней файлы и вложенные папки отображаются в рабочей области. Подлежащие архивации файлы и папки выделяются с помощью левой кнопки мыши или клавиши **Ins**. Основные операции выполняются с помощью меню или кнопок на панели инструментов.

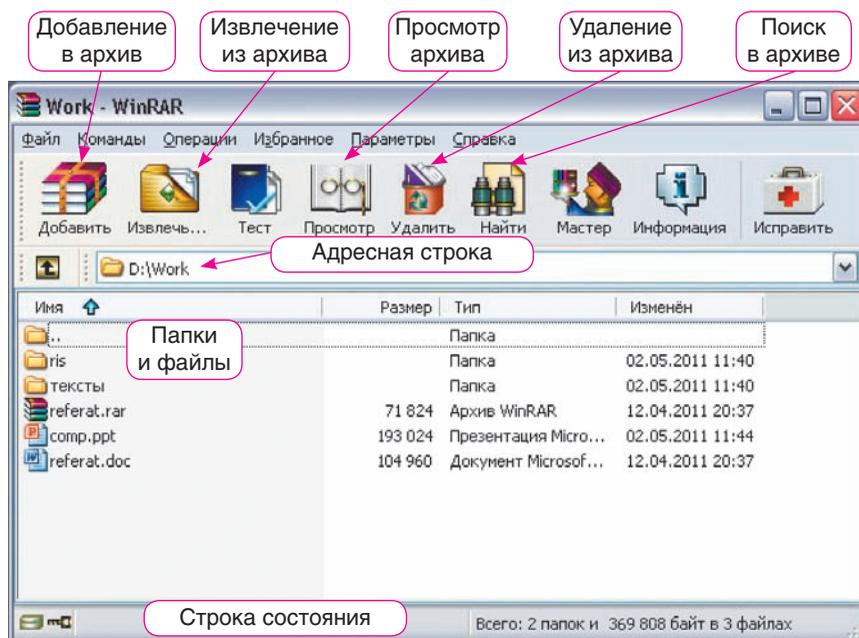


Рис. 1.1

Пример 1. Поместить файл в архив.

- Запустим программу-архиватор WinRAR.
- Выделим подлежащий архивации файл (например, referat.doc).
- С помощью меню **Команды** или кнопки  на панели инструментов выберем команду **Добавить** файл(ы) в архив.
- В появившемся диалоговом окне **Имя и параметры архива** оставим без изменения предлагаемое имя и остальные параметры (рис. 1.2). В этом случае будет создан архивный файл referat.rar (имеющий прежнее имя, но с расширением .rar), который будет сохранен в той же папке, где находится исходный.

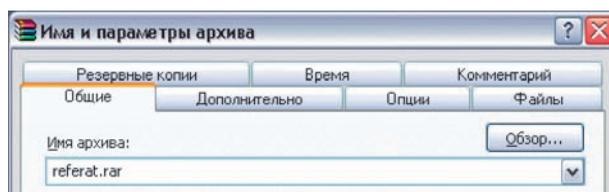


Рис. 1.2

В строке **Имя архива** можно ввести новое имя, а нажав кнопку **Обзор** — выбрать иную папку или создать новую.

Сравним объемы исходного файла и полученного архива. Их отношение называют *коэффициентом сжатия*. В приведенном примере он составляет $104\,960 / 71\,824 \approx 1,46$.

Программы-архиваторы используют различные алгоритмы для сжатия данных, поэтому размеры архивов, содержащих одни и те же файлы, но созданных с помощью различных архиваторов, могут отличаться. Размер архивного файла, как правило, меньше суммарного размера файлов, входящих в архив.

Заметим, что обычно хорошо сжимаются текстовые файлы (формат TXT) и несжатые графические в формате BMP. Практически не сжимаются архивы и графические файлы форматов GIF и JPG, поскольку они уже являются сжатыми.

Пример 2. Поместить в архив primer.rar все файлы в папках **рис** и **тексты**.

- Запустим архиватор.

Правообладатель Издательский центр БГУ

- Выделим папки **gis** и **тексты**.
- Выберем команду **Добавить** файл(ы) в архив.
- В появившемся диалоговом окне **Имя и параметры архива** введем имя **primer**. Все файлы в папках **gis** и **тексты** будут помещены в один архивный файл **primer.rar**.

Пример 3. Извлечь все файлы и папки из архива **primer.rar**.

- Запустим архиватор.
- Выделим архивный файл **primer.rar**.
- С помощью меню **Команды** или кнопки  на панели инструментов выберем команду **Извлечь** файл(ы) из архива.
- В появившемся диалоговом окне **Путь и параметры извлечения** оставим без изменения предлагаемый путь (рис. 1.3). В этом случае в папке, в которой находится архив, будет создана папка **primer** (имеющая по умолчанию то же имя, что и архив), в которую будут вложены папки **gis** и **тексты** с извлеченными файлами.

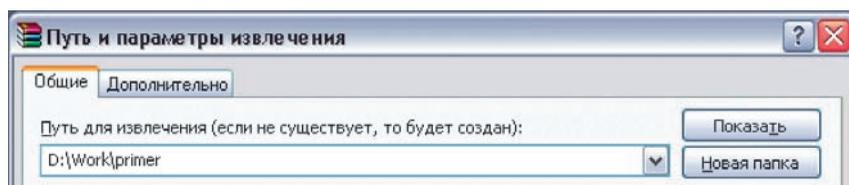


Рис. 1.3

При необходимости можно изменить строку **Путь для извлечения**, а нажав кнопку **Показать** — выбрать иную папку или диск.

В уже созданный архив можно добавлять, а также извлекать из него отдельные файлы и папки.

Пример 4. Из архива **primer.rar** извлечь файлы **text01.txt** и **text02.txt**.

- Запустим архиватор.
- Выделим архивный файл **primer.rar**.
- Просмотрим содержимое папок архива с помощью меню **Команды** → **Просмотреть файл**, либо нажатием кнопки **Просмотр**  на панели инструментов, либо двойным щелчком мышью.
- Зайдем в папку **тексты**. Выделим имена файлов **text01.txt** и **text02.txt** (рис. 1.4) и нажмем кнопку **Извлечь** .

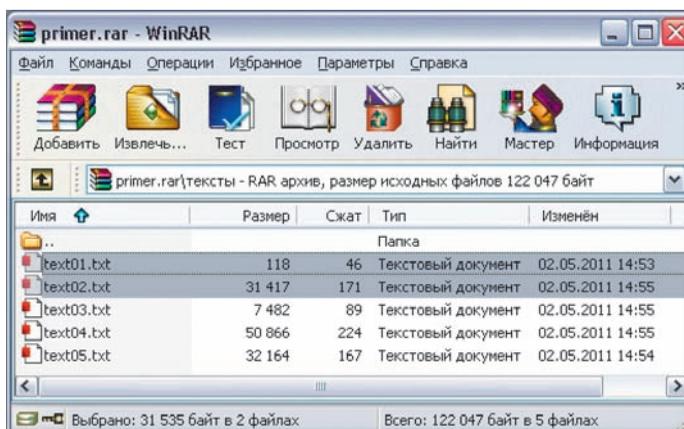


Рис. 1.4

- В появившемся диалоговом окне **Путь и параметры извлечения** изменим предлагаемый путь, введя имя папки **primer4**, и нажмем **ОК**. В результате будет создана папка **primer4**, в которую будет вложена папка **тексты** с двумя извлеченными файлами.

При создании архивов может возникнуть ситуация, когда размер архива больше размера носителя информации, на который планируется сохранить архив. Так, например, на обычный CD-диск можно поместить информацию до 700 Мбайт. Если размер архива составляет 1,3 Гбайт, то для записи потребуется 2 CD-диска и архив необходимо разбить на 2 тома. Многотомные архивы нередко используют при передаче по сети больших файлов частями. Не обойтись без многотомных архивов и при переносе большого объема данных на дискетах. Том — это фрагмент архива, состоящего из нескольких частей.

Архиватор WinRAR позволяет создавать *многотомные архивы* с указанием размера тома. По умолчанию тома получают имена вида имя_архива.partNN.rar, где NN — номер тома.

Действия при создании многотомного архива аналогичны приведенным в примерах 1 или 2. Нужно лишь задать размеры тома в окне **Имя и параметры архива** (рис. 1.5).

При разархивации многотомного архива все тома следует поместить в одну папку, а затем выполнить действия, аналогичные приведенным в примере 3.

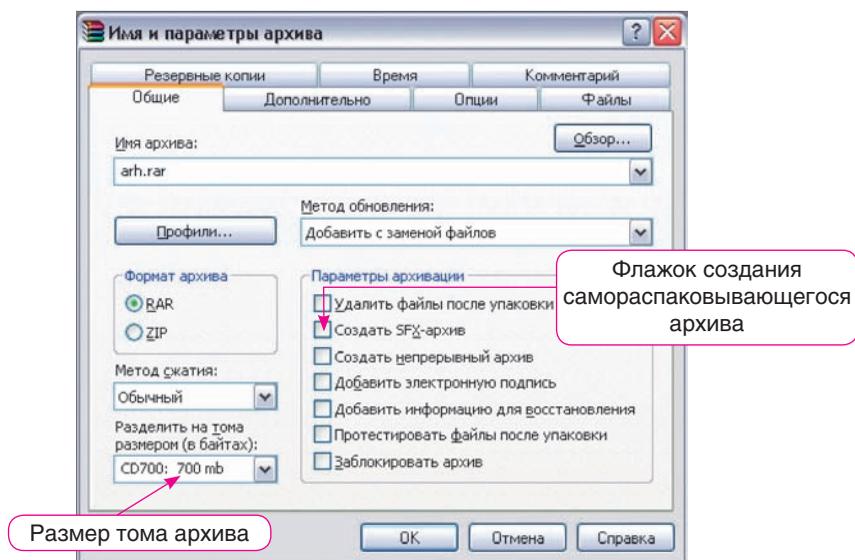


Рис. 1.5

В тех случаях, когда нужно передать кому-то архив, но нет уверенности, что у адресата есть соответствующий архиватор для его распаковки, используют *самораспаковывающийся*, или SFX-архив (от англ. Self-eXtracting). Такие архивы, как и любые исполняемые файлы, обычно имеют расширение .exe.

Действия при создании самораспаковывающегося архива аналогичны приведенным в примерах 1 и 2. Нужно лишь в окне **Имя и параметры архива** установить флажок **Создать SFX-архив** (рис. 1.5).

Для извлечения содержимого SFX-архива не требуется программ-архиваторов. Они разархивируются простым запуском архива как обычной программы. Вместе с тем для просмотра или извлечения содержимого SFX-архива можно использовать WinRAR, который позволяет работать с самораспаковывающимся архивом так же, как и с любым другим.



В настоящее время весьма актуальной проблемой является сжатие графической информации. Вы уже знаете, что растровые изображения состоят из пикселей. При сжатии информацию о каждом отдельном пикселе заменяют информацией о группах похожих

пикселей. Алгоритмы сжатия отыскивают в изображении повторы, после чего заменяют описания пикселей более крупными информационными элементами.

Различают алгоритмы сжатия без утраты информации (неразрушающие) и с контролируемой степенью утраты (разрушающие).

Неразрушающие алгоритмы позволяют впоследствии выполнить преобразование, обратное сжатию, и в точности восстановить всю исходную информацию. Такие алгоритмы используют все архиваторы, создающие файловые архивы, в том числе рассмотренный нами WinRAR.

Пусть изображение отрезка содержит 30 пикселей желтого, 50 — синего и 20 — зеленого цвета (рис. 1.6). В исходном изображении выявляются непрерывные последовательности одного цвета (цепочки одинаковых байт). Затем каждая такая последовательность описывается цветом и количеством повторений пикселя.



Рис. 1.6

Если цвет каждой из 100 точек описывать тремя байтами, то получим исходный объем $3 \text{ байта} \cdot 100 = 300 \text{ байт}$. Пусть количество повторений каждого цвета < 256 и его можно описать одним байтом. Получим $(3 \text{ байта} + 1 \text{ байт}) \cdot 3 = 12 \text{ байт}$, т. е. коэффициент сжатия $300 / 12 = 25$!

Приведенный алгоритм обеспечивает высокую степень сжатия рисованных штриховых изображений и низкую — при обработке фотографий. Если на фотографии имеется много мелких деталей, то после сжатия файл может стать не меньше, а даже больше. Описанный алгоритм *неразрушающий*, он допускает точное восстановление исходного изображения.

Более компактное представление изображения, иногда в несколько раз меньше исходного по размерам занимаемой памяти, дают разрушающие алгоритмы, например JPEG. Однако они делают невозможным точное восстановление исходного изображения.



1. Что понимается под сжатием данных?
2. С какой целью производят архивацию?
3. Каково назначение программ-архиваторов?



4. В каких случаях создают многотомные архивы? Самораспаковывающиеся архивы?
5. Чем отличаются разрушающие алгоритмы сжатия от неразрушающих?

Упражнения

1. Заархивируйте указанные учителем файлы. Сравните размеры файлов и архивов.

2. Извлеките указанные учителем файлы из архива.

3. Поместите все файлы указанной учителем папки в самораспаковывающийся архив.



4. Откройте в графическом редакторе Paint файл Ris.bmp. Сохраните его на диске в формате jpg в файле с именем Ris1.jpg, затем поместите оба файла с изображениями в архивы Rar и Zip.

Сравните размеры исходных файлов Ris.bmp и Ris1.jpg с размерами архивов Ris.zip, Ris.rar, Ris1.zip и Ris1.rar.

С помощью программы Калькулятор определите коэффициенты сжатия.

Какой архиватор лучше сжал исходный файл Ris.bmp? Файл Ris1.jpg?

ГЛАВА 2

ЦИФРОВЫЕ УСТРОЙСТВА ДЛЯ ОБРАБОТКИ ИНФОРМАЦИИ

§ 2. СОВМЕСТНОЕ ИСПОЛЬЗОВАНИЕ ЦИФРОВЫХ УСТРОЙСТВ И КОМПЬЮТЕРА

В настоящее время широко используются разнообразные цифровые устройства ввода, обработки и вывода информации, такие как цифровые фотоаппараты и видеокамеры, веб-камеры, сканеры, сенсорные экраны, электронные книги и фоторамки.

Цифровыми называют устройства, в которых информация представлена в цифровом виде и в дальнейшем может быть обработана на компьютере. Таким образом, совместное использование цифровых устройств и компьютера является неотъемлемым звеном в цепочке обработки информации. Цифровым устройствам чаще всего отводится роль получения информации в цифровом виде или ее преобразования в цифровую форму (оцифровки).

Наиболее популярными и доступными в настоящее время цифровыми устройствами стали *цифровые фотоаппараты* (рис. 2.1). Пришедшие на смену пленочным фотоаппаратам, цифровые более удобны в обращении, компактны и надежны. Представление информации в цифровом виде существенно упрощает обработку изображений с помощью компьютера.

В зависимости от типа и назначения цифрового фотоаппарата (любительский или профессиональный) интерфейс и особенности работы с ним могут отличаться.



Рис. 2.1

Чаще всего цифровые фотоаппараты подключаются к компьютеру через USB-порт. Для этой цели в комплект поставки входит специальный кабель. После подключения устройства к компьютеру операционная система определяет наличие сменного диска, каковым является устройство памяти фотоаппарата, и дальше работает с ним как с обычным носителем информации, обес-

печивая возможность удаления, копирования и перемещения файлов с изображениями. Простые бытовые фотоаппараты чаще всего сохраняют изображения в формате JPG или TIF в зависимости от качества снимка (разрешения изображения), которое выбирает пользователь. Современные цифровые фотоаппараты обеспечивают возможность видеозаписи, а также съемки и сохранения нескольких кадров в одном файле.

Совместное использование цифрового фотоаппарата и компьютера рассмотрим на примере создания презентации «Природа родного края». Эта работа включает в себя следующие этапы: фотографирование, выбор и копирование в компьютер подходящих фотографий, кадрирование и коррекция изображений и, наконец, вставка их на слайды презентации.

Пример 1. Используя цифровой фотоаппарат и компьютер, создать презентацию «Природа родного края».

- Пусть на карте памяти фотоаппарата сохранены фотографии требуемой тематики. Подключим фотоаппарат к компьютеру.
- Скопируем подходящие изображения с фотоаппарата в компьютер в папку **Rabota**. В эту же папку скопируем и фоновые изображения из папки **Ris**, например контурную карту Беларуси.
- Откроем программу PowerPoint и подготовим шаблон презентации — фоновые рисунки и заголовок.
- Будем последовательно открывать сохраненные изображения в графическом редакторе, например в Paint, вырезать нужные фрагменты и вставлять на слайды презентации.
- В результате получим презентацию собственных фотографий. Сохраним ее в папке **Rabota**. Возможный вид первого слайда показан на рис. 2.2.

В последнее время большое распространение получили цифровые видеокамеры, веб-камеры и документ-камеры.

Цифровая видеокамера (рис. 2.3) служит для получения, записи, хранения и передачи видеоизображения



Рис. 2.2



Рис. 2.3

и звука в цифровой форме. В простых бытовых видеокамерах видеoinформация сохраняется на DV-кассеты, DVD-диски или flash-память в виде файлов форматов AVI, MPEG, WMV. Эти файлы являются сжатыми, их объем практически не уменьшается при архивации. Их можно копировать и обрабатывать на компьютере: редак-

тировать, разрезать на фрагменты и вставлять в видеофильмы, презентации или flash-анимации.

Для записи на компьютер звука и видео с цифровой видеокамеры и дальнейшего использования записанного материала удобно использовать программу Windows Movie Maker. Она также позволяет использовать в создаваемых фильмах существующие аудио- и видеозаписи и неподвижные изображения, добавлять заголовки, видеопереходы и эффекты.

Пример 2. Создать видеофильм «Живая природа».

- Пусть наш видеофильм будет содержать две фотографии и видеофрагмент.
- Подключим видеокамеру к компьютеру.
- Запустим программу Movie Maker (рис. 2.4). Импортируем видеофрагмент с видеокамеры, а также фотографии.
- Перетащим на шкалу времени изображение, которое будет использоваться в качестве заставки, например aist.jpg. Добавим название фильма.
- Перетащим на шкалу времени видеофрагмент, например Bear.wmv. Зададим длительность 6 с.
- Добавим еще одно изображение, например doroga.jpg.
- Установим переходы между кадрами, например **Растворение**.
- Просмотрим созданный видеофильм. Сохраним его в папке **Rabota**.

Веб-камера (рис. 2.5) служит для получения видеоизображения и звука в цифровой форме и передачи их по компьютерной сети. Веб-камеры, как правило, не имеют встроенных устройств хранения ин-

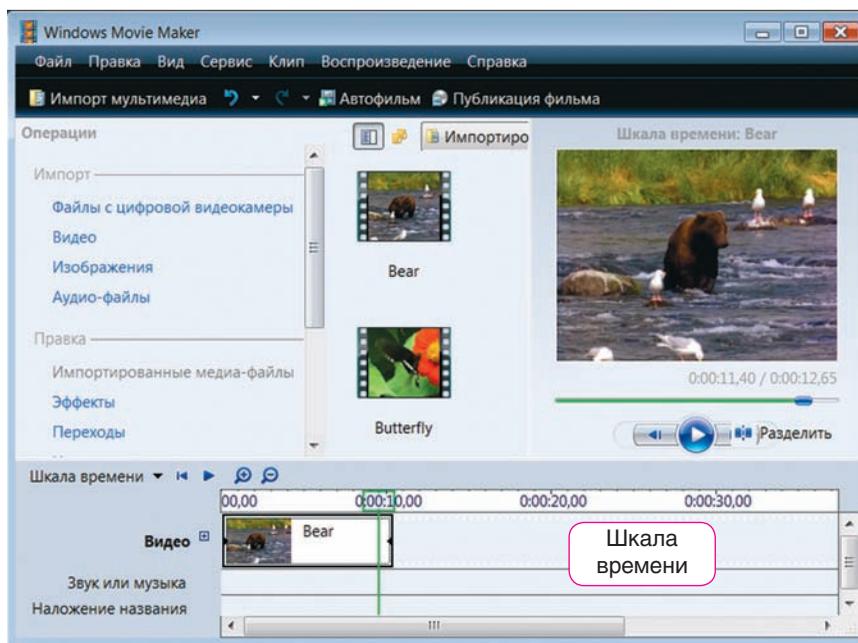


Рис. 2.4

формации. В то же время они не требуют специальных адаптеров для подключения к компьютерным сетям. Подключение веб-камеры к компьютеру может выполняться через порт USB (USB-разъем имеется в большинстве современных веб-камер), а также через локальную сеть и даже через стандартный телефонный модем.

Веб-камеры используются для видеосъемки и видеонаблюдения, организации видеоконференций и передачи видеоизображения по сети. Видеоизображение с веб-камеры можно сохранять и просматривать на любом сетевом компьютере, на котором установлен стандартный веб-браузер, например Internet Explorer. С помощью специальных программ можно дистанционно управлять веб-камерами в сети.

Документ-камера (рис. 2.6) предназначена для получения и передачи изображений: бумажных документов, рекламных материалов,



Рис. 2.5



Рис. 2.6

рентгеновских снимков и даже трехмерных объектов.

Изображение, полученное с помощью документ-камеры, может быть сохранено в компьютере, показано на экране телевизора, спроецировано на экран с помощью мультимедийного проектора, передано по сети Интернет. Документ-камеры широко применяются в науке и образовании.

В последнее время широко распространились новые цифровые устройства ввода-вывода информации.

Во многих устройствах используется **сенсорный экран** — монитор, который чувствителен к прикосновениям и позволяет работать с компьютером при помощи касаний. Иногда для прикосновения к экрану применяется специальный указатель — стилус, который не повреждает экран (рис. 2.7, а). Сенсорные экраны применяются в банкоматах, платежных терминалах и информационных киосках (рис. 2.7, б), в оборудовании для автоматизации торговых операций, в карманных компьютерах, планшетах, коммуникаторах и электронных книгах.

Термин **электронная книга** применяется как для обозначения произведений, представленных в электронном виде, так и для устройств, ис-



Рис. 2.7, а



Рис. 2.7, б

пользуемых для их отображения (рис. 2.8). Основным отличием электронной книги от персонального компьютера является ограниченная функциональность при существенно большем времени автономной работы.

Современные электронные мультимедиа-книги могут содержать не только тексты, но и изображения, видеофрагменты, а также звуковое и музыкальное сопровождение. Чаще всего электронные книги представляют собой документы в форматах: PDF, DJVU, DOC, HTML, JPEG. Их несложно создавать самостоятельно и затем переписывать на устройство чтения.



Рис. 2.8



1. Какие цифровые устройства вам известны?
2. Каково их назначение?

Упражнение

Разработайте проект видеофильма или мультимедиа-презентации, предполагающий использование цифрового фотоаппарата, видео- или веб-камеры и компьютера по одной из тем: Мой класс, Моя семья, Мои друзья, Мой сад, Дары леса, Растения, Птицы, Грибы, Ягоды, Цветы.

§ 3. РАБОТА СО СКАНЕРОМ

Одним из устройств бесклавиатурного ввода информации является сканер — устройство, предназначенное для создания и передачи в память компьютера цифровой копии изображения.

Принцип действия сканера следующий: отраженный от предмета или изображения свет подается через оптическую систему на светочувствительные элементы. Полученная в виде электрических сигналов аналоговая информация преобразовывается в цифровую форму, передается в компьютер и может быть сохранена в виде графического файла. Если сканируется текст, то с помощью специальной программы



Рис. 2.9



Рис. 2.10

распознавания его можно сразу преобразовать в текстовую форму. Если сканируется чертеж, иллюстрация или фотография, то их сохраняют в виде изображения. На практике чаще всего используются два типа сканеров: *ручные* и *планшетные* (настольные).

Ручной сканер (рис. 2.9) наиболее часто используется для считывания кода товара.

В планшетных сканерах (рис. 2.10) сканируемый материал помещается под крышку и сканируется построчно: над или под изображением, расположенным на прозрачном стекле, движутся светочувствительные элементы.

После ввода в компьютер цифровой копии изображения осуществляется его дальнейшая обработка.

Для работы со сканерами используются различные программы сканирования и распознавания. Одной из самых распространенных программ распознавания текстов является программа ABBYY FineReader. Она проста в обращении, основные ее операции вынесены на панель инструментов (рис. 2.11).

Инструмент **Сканировать** обеспечивает ввод информации в виде необработанного изображения, а **Сканировать и распознать** — сканирование и последующее распознавание текста. Распознавание осуществляется путем сравнения полученных изображений символов с символами имеющихся шрифтов. Инструмент **Проверить** позволяет обнаружить неправильно распознанные символы и слова и предлагает исправить возможные ошибки. **Мастер сохранения результатов** обеспечивает передачу результатов сканирования и распознавания в различные программы, например в приложения MS Office (рис. 2.12).

Пример. Отсканировать страницу, содержащую текст и изображение. Выполнить распознавание текста. Скопировать распознанный текст и изображение в документ Word.

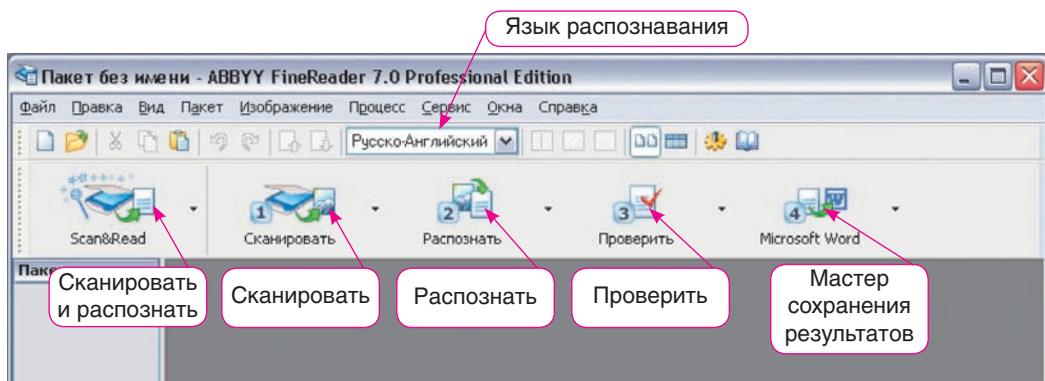


Рис. 2.11

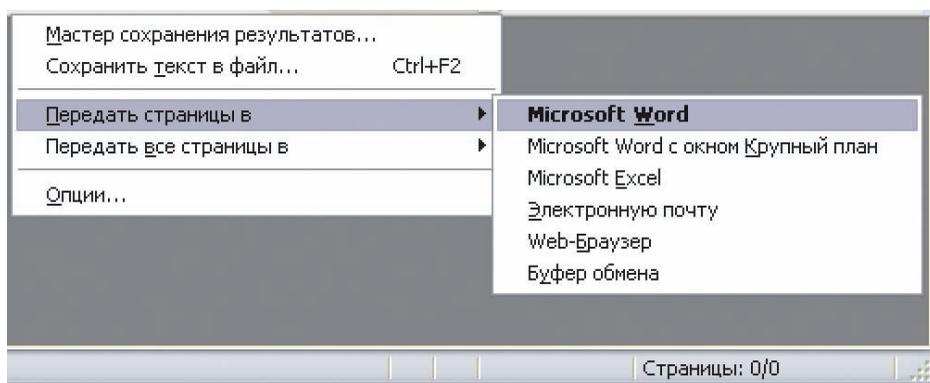


Рис. 2.12

- Включим сканер. Поместим сканируемый документ под крышку сканера.
- Запустим программу FineReader.
- Выберем инструмент **Сканировать**. Откроется диалоговое окно сканера (рис. 2.13). Его вид для разных моделей сканеров может несколько отличаться. В этом окне устанавливаются параметры сканирования, например: режим сканирования (цветной), разрешение (для наших целей достаточно 300 точек на дюйм). При необходимости можно указать масштаб и размер изображения, а также сканируемую область.

Правообладатель Издательский центр БГУ

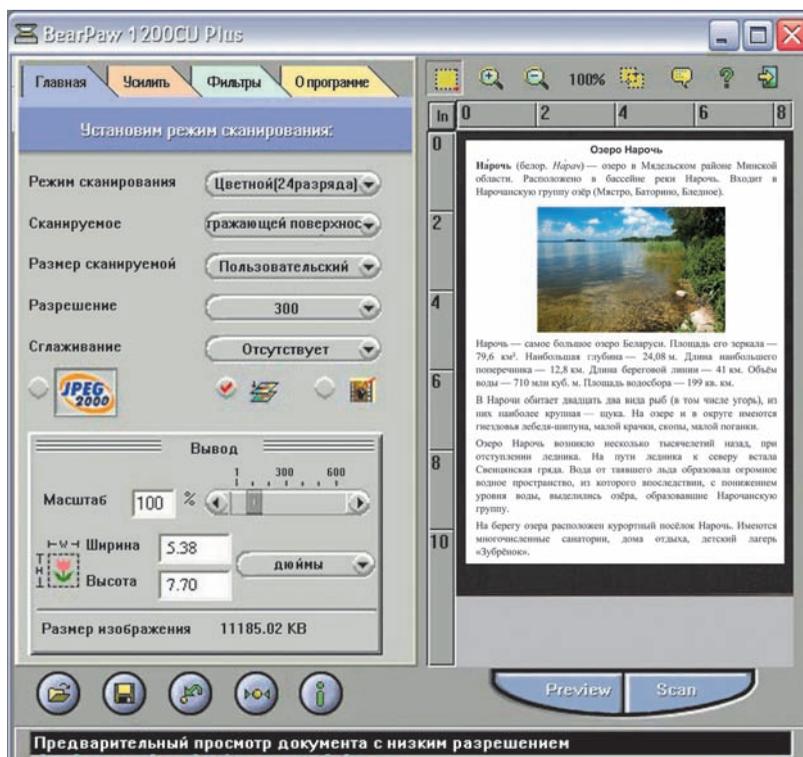


Рис. 2.13

- Нажмем кнопку **Preview** для предварительного просмотра сканируемого материала. С помощью мыши выделим сканируемую область (по умолчанию она соответствует всей сканируемой странице).
- Нажатием кнопки **Scan** запустим процесс сканирования. По его завершении изображение отсканированной страницы можно сохранить в одном из графических форматов (например, TIF или JPG). Это позволит разделить этапы сканирования и распознавания. Программа FineReader поддерживает работу с многими графическими форматами.
- Выделим подлежащий распознаванию фрагмент изображения и вызовем инструмент **Распознать**. Распознаваемый текст отображается в окне встроенного текстового редактора. Возможные ошибки выделяются цветом и могут быть исправлены.

Правообладатель Издательский центр БГУ

- Для дальнейшей обработки и форматирования текста передадим его в текстовый редактор Word (см. рис. 2.12).
- Выделим требуемый фрагмент с рисунком и с помощью буфера обмена скопируем его в графический редактор, например в Paint. Выполним обрезку и коррекцию изображения. Скопируем его в документ Word.
- Отформатируем документ Word и сохраним его в файле primer.doc. Подобная последовательность действий используется почти во всех программах сканирования и распознавания. Иногда некоторые шаги пропускаются или объединяются.



1. Каково основное назначение сканера?
2. Каковы основные этапы сканирования и распознавания информации?

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

§ 4. СИМВОЛЬНЫЕ И СТРОКОВЫЕ ВЕЛИЧИНЫ. ОПЕРАЦИИ НАД СИМВОЛЬНЫМИ И СТРОКОВЫМИ ВЕЛИЧИНАМИ

Компьютеры могут обрабатывать данные различных типов. Вам уже приходилось иметь дело с числовыми типами данных: целым и вещественным. Не менее широко компьютеры используются для обработки текста. Для этого имеются соответствующие процедуры и функции, которые работают с символьными и строковыми величинами. Эти величины отличаются друг от друга тем, что значением символьной переменной является один символ, а строковой — строка символов.

В этой главе вам предстоит познакомиться с обработкой данных символьного и строкового типов.

4.1. СИМВОЛЬНЫЙ ТИП ДАННЫХ

Для работы с символами в языке Pascal предусмотрен специальный тип данных, который называется *символьным* и обозначается **char** (от англ. character — символ). Значениями переменных этого типа являются отдельные символы: буквы, цифры, знаки.

Символьные переменные описываются так:

```
var имя_переменной: char;
```

Пример описания переменных: `var a, b, ch: char;`

Каждый символ имеет свой уникальный двоичный код от 00000000 до 11111111 (от 0 до 255 в десятичном представлении). В оперативной памяти один символ занимает 1 байт. Символы с кодами от 0 до 31 (управляющие) воспринимаются в тексте как пробел, а в операторах вывода — как символ или соответствующая команда, например: символ с кодом 07 — подача звукового сигнала; 08 — возврат на один символ и стирание следующего за ним (клавиша **BackSpace**); 09 — табуляция (клавиша **Tab**); 13 — ввод (клавиша **Enter**); 27 — отмена команды (клавиша **Esc**). В качестве символов могут использоваться буквы национальных алфавитов и некоторые специальные символы.

Все символы упорядочены в соответствии с их кодами. Код символа в виде целого числа возвращает функция `ord(x)`, если аргументом является символ алфавита. Коды всех символов размещают в таблице (приложение 1). Первая половина таблицы (коды 0...127) стала международным стандартом, который называется ASCII — American Standard Code for Information Interchange (читается «аски код»). В ней содержатся цифры, латинский алфавит и служебные символы. Во второй половине таблицы размещены символы национальных алфавитов. Размещение символов кириллицы имеет несколько вариантов. Для работы в Windows часто используют кодировку Windows-1251 (CP-1251) (приложение 2). В этой кодировке заглавные буквы кириллицы имеют коды от 192 (А) до 223 (Я), а строчные — коды от 224 (а) до 255 (я). Заглавная буква Ё имеет код 168, а строчная ё — код 184.

Над символьными переменными можно выполнять следующие действия: присваивание, ввод и вывод, сцепление (конкатенацию), сравнение символов.

Значения символьных констант и переменных заключают в апострофы (одинарные кавычки), например: 'А', 'В', 'С', '4', '7', ' ' (пробел). Если нужно использовать сам апостроф, то его повторяют дважды: В := '''. Символьные значения можно задавать и их числовыми кодами. В этом случае перед кодом записывают знак #.

Примеры присваивания символьных значений (они предварительно должны быть описаны `var a,aa,b,bb,c,d: char;`):

<code>a:='A';</code>	{прописная латинская буква А вводится непосредственно}
<code>aa:=#65;</code>	{прописная латинская буква А задается значением ее кода}
<code>b:='!';</code>	{символ восклицательный знак вводится непосредственно}
<code>bb:=#33;</code>	{символ восклицательный знак задается значением его кода}
<code>c:=#13;</code>	{задание кода клавиши Enter }
<code>d:=#27;</code>	{задание кода клавиши Esc }

Вывод символьных переменных выполняется с помощью стандартных процедур `write` и `writeln`. Примеры вывода:

Правообладатель Издательский центр БГУ

```

write(a) или write(aa); {выводится буква А}
write(b) или write(bb); {выводится символ !}
write(c);                {команда перевода курсора
                          на новую строку}
write(d);                {выводится специальный символ,
                          соответствующий коду #27}

```

Ввод символьных переменных выполняется с помощью стандартных процедур `read` и `readln`. Обратим внимание на следующую особенность ввода нескольких символов одной процедурой. Нажатие клавиши **Пробел** приводит к присвоению символьной переменной значения ' ' (пробел, код 32), а клавиши **Enter** — к присвоению кода этой клавиши (13). Поэтому значения символьных переменных набирают на клавиатуре одно за другим без пробелов.

Например, процедура `readln(a,b,c)` ожидает ввода значений трех символьных переменных `a`, `b`, `c`, т. е. набора на клавиатуре трех символов. Если набрать буквы `У`, `Р`, `А` и нажать **Enter**, то процедура `write(a,b,c);` выведет на экран слово **УРА**.

Операция соединения нескольких символов называется сцеплением или конкатенацией и обозначается символом `+` (плюс). Например, процедура `write (a+b+c+'!!!');` выведет на экран **УРА!!!**

Пример 1. Составить программу, в которой выполняется присваивание, ввод и вывод символов.

Прежде всего объявим тип `char` (символьный) переменных `a`, `b`, `c`, `d`. Значения переменных `a`, `b`, `c` будем вводить с клавиатуры. Пусть значение переменной `d` присваивается программой, например: `d:='!';`

Программа может выглядеть так:

```

program Primer41_1;
var a, b, c, d: char;
begin
  d:='!';                {присваивание значения}
  write('Введите три символа: ');
  read(a,b,c);          {ввод трех символов}
  writeln(a+b+c+d);    {сцепление и вывод}
end.

```

Результат выполнения программы может выглядеть так:

```
Введите три символа: УРА
УРА!
```

Pascal содержит несколько стандартных функций для работы с символами (приложение 3). Рассмотрим на примерах использование следующих функций:

- ord(s) — определяет код символа s,
- chr(n) — возвращает символ с кодом n.

Пример 2. Составить программу, которая определяет код введенного с клавиатуры символа.

Код введенного символа s определим с помощью функции ord(s).

Программа может выглядеть так:

```
program Primer41_2;
var s: char;
begin
    write('Введите символ: '); readln(s);
                                     {ввод символа}
    writeln('Код этого символа = ', ord(s));
                                     {вывод кода}
end.
```

Результат выполнения программы может выглядеть так:

```
Введите символ: R
Код этого символа = 82
Введите символ: Я
Код этого символа = 223
```

Пример 3. Составить программу, которая определяет символ по его коду, введенному с клавиатуры.

Символ с кодом n возвращается функцией chr(n).

Программа может выглядеть так:

```
program Primer41_3;
var n: integer;
begin
    write('Введите код символа от 33 до 255: ');
    readln(n);
                                     {ввод кода}
    writeln('Символ = ', chr(n));
                                     {вывод символа}
end.
```

Результат выполнения программы может выглядеть так:

```
Введите код символа от 33 до 255: 33
Символ = !
Введите код символа от 33 до 255: 200
Символ = И
Введите код символа от 33 до 255: 255
Символ = я
```



Пример 4. Составить программу, которая определяет, какой символ введен с клавиатуры: цифра, большая или малая латинская буква.

Введенный символ a преобразуем в код функцией $\text{ord}(a)$.

Цифры имеют коды от 48 до 57, заглавные латинские буквы — от 65 (A) до 90 (Z), строчные — от 97 (a) до 122 (z) (см. приложение 1).

Программа может выглядеть так:

```
program Primer41_4;
var a: char;
begin
  write('Введите символ: '); read(a);
  if (ord(a)>=48) and (ord(a)<=57)
    then writeln(a, ' цифра');
  if (ord(a)>=65) and (ord(a)<=90)
    then writeln(a, ' заглавная буква');
  if (ord(a)>=97) and (ord(a)<=122)
    then writeln(a, ' малая буква');
end.
```

Результат выполнения программы может выглядеть так:

```
Введите символ: q
q малая буква
Введите символ: G
G заглавная буква
Введите символ: 5
5 цифра
```

Благодаря тому, что в компьютере символы представляются в виде числовых кодов, к ним можно применять операции сравнения. Допустимы шесть операций сравнения: $=$, $<=$, $>=$, $<$, $>$, $<>$.

Из двух символов большим считается тот, код которого больше.

Символы упорядочены в соответствии с их кодами следующим образом: цифры — по возрастанию: $'0' < '1' < \dots < '9'$, буквы — в алфавитном порядке: $'A' < 'B' < \dots < 'Z'$, $'a' < 'b' < \dots < 'z'$, $'A' < 'B' < \dots < 'Я'$, $a' < 'б' < \dots < 'я'$.

Исключение составляют буквы Ё (код 168) и ё (код 184).

Пример 5. Составить программу, которая выполняет сравнение введенного с клавиатуры символа с заданным.

Прежде всего объявим символьный тип переменных *x* и *y*.

Переменной *y* присвоим некоторое значение, например *y* := 'h'. Запишем процедуру ввода значения переменной *x*, затем запишем условие сравнения переменных: *if x*<*y*. Если это условие выполняется, то выводим значения, соединенные знаком меньше: *x*, '<', *y*, иначе выводим *x*, '>=', *y*.

Программа может выглядеть так:

```
program Primer41_5;
var x, y: char;      {переменные символьного типа}
begin
  y := 'h';          {присваивание значения символа}
  write('Введите символ: '); read(x); {ввод символа}
  if x < y then writeln(x, '<', y)
  else writeln(x, '>=', y);      {сравнение символов}
end.
```

Результат выполнения программы может выглядеть так:

```
Введите символ: b
b<h
Введите символ: q
q>=h
```



1. По таблице кодов (см. приложение 1) определите, какому символу соответствует код: а) 27; б) 32; в) 33; г) 80; д) 120.
2. Какие действия можно выполнять над символьными переменными?
3. Какая функция возвращает код символа?
4. Какая функция возвращает символ с заданным кодом?
5. Какие возможны операции сравнения символов?
6. Какой из двух символов считается большим?
7. Какие утверждения верны?
 - а) 'д' > 'ж'; б) 'и' > 'л'; в) 'д' < 'п'; г) 'А' > 'а'; д) 'я' > 'Я'.

Упражнение

Составьте программу, которая:

- а) определяет коды двух введенных с клавиатуры символов;

б) выполняет операцию сцепления трех введенных с клавиатуры символов;

в) выполняет сравнение двух введенных с клавиатуры символов и выводит результат сравнения на экран. Например: 'а' < 'б', 'д' = 'д', 'я' > 'ы';



г) выполняет сравнение двух введенных с клавиатуры символов и выводит их на экран в порядке возрастания их кодов.

4.2. ПРОЦЕДУРЫ И ФУНКЦИИ ОБРАБОТКИ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ

Продолжим обзор стандартных процедур и функций, предназначенных для работы с символами. Рассмотрим функции `pred(s)`, которая возвращает предыдущий символ, и `succ(s)`, которая возвращает следующий символ. Напомним, что символы упорядочены в соответствии с их кодами.

Пример 1. Составить программу, которая для введенного с клавиатуры символа выводит предыдущий и следующий символы.

Программа может выглядеть так:

```
program Primer42_1;
var ch: char;
begin
    write('Введите символ '); read(ch);
    writeln('пред = ', pred(ch), ' след = ', succ(ch));
end.
```

Результат выполнения программы может выглядеть так:

Введите символ R
пред = Q след = S
Введите символ Д
пред = Г след = Е

Пример 2. Составить программу, которая выводит таблицу символов и их кодов от 32 до 255 (по 8 в строке).

Определение и вывод символов и кодов выполним в цикле `for`. Параметром цикла будет служить код символа `k`, изменяющийся от 32 до 255. С помощью процедуры `write(k:4, chr(k):2, ' ');` осу-

ствим форматный вывод кода и символа в строку. Переход на новую строку будет происходить при условии, что остаток целочисленного деления кода k на 8 равен 7, т. е. через каждые 8 символов.

Программа может выглядеть так:

```

program Primer42_2;
var k: integer;
begin
  for k:=32 to 255 do begin
    write(k:4, chr(k):2,' ');           {форматный вывод
                                        кода и символа}
    if k mod 8 = 7 then writeln;       {переход на новую
                                        строку}
  end;
end.

```

Результат выполнения программы может выглядеть так:

32	33 !	34 "	35 #	36 \$	37 %	38 &	39 '
40 (41)	42 *	43 +	44 ,	45 -	46 .	47 /
48 0	49 1	50 2	51 3	52 4	53 5	54 6	55 7
56 8	57 9	58 :	59 ;	60 <	61 =	62 >	63 ?
64 @	65 A	66 B	67 C	68 D	69 E	70 F	71 G
72 H	73 I	74 J	75 K	76 L	77 M	78 N	79 O
80 P	81 Q	82 R	83 S	84 T	85 U	86 V	87 W
88 X	89 Y	90 Z	91 [92 \	93]	94 ^	95 _
96 `	97 a	98 b	99 c	100 d	101 e	102 f	103 g
104 h	105 i	106 j	107 k	108 l	109 m	110 n	111 o
112 p	113 q	114 r	115 s	116 t	117 u	118 v	119 w
120 x	121 y	122 z	123 {	124	125 }	126 ~	127
128 Ъ	129 Ѓ	130 ,	131 Ѕ	132 „	133 ...	134 †	135 ‡
136 €	137 №	138 Ь	139 <	140 Ъ	141 Ѓ	142 Є	143 Ц
144 Ѕ	145 ´	146 ´	147 ¨	148 ¨	149 •	150 –	151 —
152 ~	153 ™	154 ъ	155 >	156 ъ	157 ѓ	158 ф	159 ц
160	161 Ÿ	162 Ÿ	163 Ј	164 ѡ	165 Ѓ	166 †	167 §
168 Ѓ	169 ©	170 €	171 «	172 ¬	173	174 ®	175 ĩ
176 ·	177 ±	178 I	179 i	180 r	181 μ	182 ¶	183 •
184 ë	185 №	186 e	187 »	188 j	189 S	190 s	191 i
192 A	193 B	194 B	195 Г	196 Д	197 E	198 Ж	199 З
200 И	201 Й	202 K	203 Л	204 М	205 Н	206 О	207 П
208 P	209 C	210 T	211 Y	212 Ф	213 X	214 Ц	215 Ч
216 Ш	217 Щ	218 Ъ	219 Ы	220 Ь	221 Э	222 Ю	223 Я
224 a	225 б	226 в	227 г	228 д	229 е	230 ж	231 з
232 и	233 й	234 к	235 л	236 м	237 н	238 о	239 п
240 p	241 c	242 t	243 y	244 ф	245 x	246 ц	247 ч
248 ш	249 щ	250 ъ	251 ы	252 ь	253 э	254 ю	255 я



Пример 3. Составить программу, которая выводит коды нажатых клавиш до тех пор, пока не нажата клавиша **Esc**.

Прежде всего подключим модуль CRT, который содержит функции и процедуры работы с клавиатурой. С помощью функции `readKey` в переменную `c` типа `char` будем считывать символ нажатой клавиши. Преобразуем его в код с помощью стандартной функции `ord(c)`. Считывание символов и вывод кодов будем продолжать до тех пор, пока не будет нажата клавиша **Esc** (код 27).

Программа может выглядеть так:

```
program Primer42_3;
uses CRT;                                {подключение модуля CRT}
var c: char;
begin
  while c<>#27 do begin {27 - код клавиши Esc}
    c:=readKey;          {считывание символа}
    write(ord(c),' ');  {преобразование и вывод кода}
  end;
end.
```

Результат выполнения программы может выглядеть так:
нажимаем клавиши: A Z a z A Я а я пробел Enter Esc



Заметим, что при нажатии клавиш управления курсором будут выводиться два числа (первое из которых — нуль). Например, при нажатии клавиши → (стрелка вправо) выводятся числа 0 и 75. Можно модифицировать вывод так, чтобы эти два числа выводились слитно:

```
if c<>#0 then writeln(ord(c), ' ')
else writeln('0', ord(readKey), ' ')
```



1. Какой символ вернет функция `pred('D')`?
2. Какой символ вернет функция `succ('Ж')`?

Упражнения

1. С клавиатуры вводят символ. Составьте программу, которая выводит два следующих символа.

2. С клавиатуры вводят код символа. Составьте программу, которая выводит предыдущий символ.

3. Составьте программу, которая выводит символы с кодами от 60 до 70.

4.3. СТРОКОВЫЙ ТИП ДАННЫХ

Символьный тип данных позволяет работать с отдельными символами. Для обработки более крупных фрагментов текста — строк используют тип данных, который так и называется: `string` — строка. Значениями переменных этого типа являются строки, содержащие не более 255 любых символов.

Переменные этого типа должны быть описаны одним из способов:
`var имя: string;` или `var имя: string[n];`

В первом случае длина строки по умолчанию не должна превышать 255 символов, во втором максимальное число символов `n` задается, причем $1 \leq n \leq 255$.

Основные действия, которые можно выполнять над строковыми переменными: ввод и вывод, присваивание, сцепление, сравнение.

Ввод строковых переменных выполняется с помощью стандартных процедур `read(st)` и `readln(st)`. Обратим внимание на отличия ввода числовых и строковых переменных.

Напомним, что при вводе значений нескольких числовых переменных (`var a,b,c: integer`) одной процедурой `read(a,b,c)` в строке ввода можно набрать несколько чисел или сразу все, отделяя их друг от друга пробелами, и нажать **Enter**. При этом каждый набор отображается в окне вывода в одной строке. Если в строке ввода набрать больше трех чисел, то будут присвоены только первые три значения. Если набрать меньше трех чисел, то программа будет ожидать продолжения ввода.

Ввод значения строковой переменной (`var a,b,c: string`) всегда завершают нажатием клавиши **Enter** (пробел — это обычный символ, и его для этой цели использовать нельзя).

Вывод строковых переменных выполняется с помощью стандартных процедур `write(a,b,c)` и `writeln(a,b,c)` и не отличается от вывода данных других типов. Процедура `write` выводит значения в одну строку. При этом выводимые значения необходимо отделять пробела-

ми или иными символами (например, запятой или точкой с запятой), иначе все они будут напечатаны слитно. Процедура `writeln` после вывода значений переводит курсор на новую строку.

Операция соединения (сцепления, конкатенации) строк обозначается знаком `+`. Например, результатом операции: `'пар'+ 'о'+ 'ход'` будет слово `'пароход'`.

Пусть `s`, `s1` и `s2` — переменные строкового типа. В результате выполнения операторов `s1:= 'пар'; s2:= 'воз'; s:=s1+'о'+s2;` в переменную `s` запишется слово `'паровоз'`. Если в операции сцепления поменять переменные `s1` и `s2` местами: `s:=s2+'о'+s1;`, то в переменной `s` окажется слово `'возопар'`.

Пример 1. Составить программу, которая выполняет присваивание, ввод, соединение и вывод строк.

Прежде всего объявим тип `string` (строковый) переменных `a`, `b` и `c`.

Переменной `a` присвоим некоторое значение, например `a:= 'Информатика'`. С клавиатуры будем вводить слова «компьютер» и «программы». Запишем процедуры вывода с различными разделителями между словами и операцией конкатенации.

Программа может выглядеть так:

```
program Primer43_1;
var a, b, c: string;
begin
  a:= 'Информатика';
  read(b); read(c);           {ввод}
  writeln(a, ', ', b, ', ', c); {вывод}
  writeln(a + ': ' + b + ' и ' + c); {соединение и вывод}
end.
```

Результат выполнения программы может выглядеть так:

```
компьютер
программы
Информатика, компьютер, программы
Информатика: компьютер и программы
```

Среди всевозможных значений строк есть пустая строка. Она вводится двумя апострофами (одинарными кавычками) без пробела между

ними. А чтобы записать апостроф в составе строки, его надо повторить дважды. Например, процедура `write('аб'ява')` выведет на экран слово **аб'ява**.

Для определения длины строковой переменной, т. е. количества символов в ней, используют функцию `length(s)`. Например, `s := 'ИНФОРМАТИКА'`; тогда переменной `n := length(s)` будет присвоено значение 11.

Заметим, что тип данных `string` обладает свойствами и простых, и составных типов. Например, при вводе и выводе строк используют имя строки (свойство простой переменной).

С другой стороны, строка — это последовательность символов. Ее можно рассматривать как совокупность из n символов, т. е. как *массив символов* (составной тип данных). К любому символу в строке можно обратиться так же, как к элементу массива из n символов. Для этого после имени строки надо указать в квадратных скобках индекс — порядковый номер символа в строке. Если описана переменная `s: string`, то `s[1]` — это первый символ строки, `s[2]` — второй и т. д. Например: `s := 'ИНФОРМАТИКА'`; тогда `s[3] = 'Ф'`; `s[4] = 'О'`.

Физическое представление строки приведено на рис. 3.1.

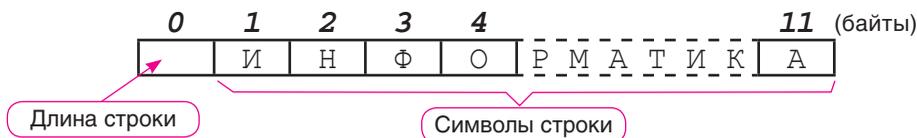


Рис. 3.1



Заметим, что у элемента массива `s[0]` особая роль — хранить длину строки. Значением `s[0]` является символ, код которого равен длине строки (количеству символов в строке).

Пример 2. Составить программу, которая выводит символы введенного с клавиатуры слова и их коды.

Будем рассматривать введенное слово `s` как массив символов. Значение каждого символа `s[i]` определим по его индексу i , а код определим с помощью функции `ord(s[i])`.

Программа может выглядеть так:

```

program Primer43_2;
var s: string; i: integer;
begin
  write('Введите слово: '); read(s);           {ВВОД}
  for i:=1 to length(s) do
    writeln(i, 'символ ', s[i], ' код=', ord(s[i]));
                                                    {ВЫВОД}
end.

```

Результат выполнения программы может выглядеть так (выводится индекс, символ и его код):

Итак, элементы строки имеют индексы от 0 до n . В процессе обработки данных для строки выделяется память, равная длине присвоенной ей строковой переменной, увеличенной на единицу. Таким образом, длина строки может быть от 0 до n (но не более 255), а в оперативной памяти для нее отводится от 1 до $(n + 1)$ байт.

```

Введите слово: школа
1 символ ш код=248
2 символ к код=234
3 символ о код=238
4 символ л код=235
5 символ а код=224

```

Пример 3. Составить программу, которая определяет, какое из двух введенных с клавиатуры слов разной длины более длинное, и выводит это слово.

Алгоритм основан на сравнении длин строковых переменных `length(a)` и `length(b)`.

Программа может выглядеть так:

```

program Primer43_3;
var a, b: string;
begin
  writeln('Введите два слова');
  read(a,b);                                     {ВВОД СЛОВ}
  if length(a)>length(b)
    then writeln(a) else writeln(b);
end.

```

Результат выполнения программы может выглядеть так:

Сравнение строк выполняется сравнением кодов соответствующих символов до первого несоп-

```

Введите два слова
школа
гимназия
гимназия

```

дения. Строки считаются равными, если в них совпадают все символы. Из двух строк «меньшей» считается та, в которой код первого несовпадающего символа меньше, чем код соответствующего символа другой строки (согласно этому правилу строчные буквы «больше» прописных). Так, строка 'стол' меньше строки 'стул' (первые два символа совпадают, третий символ 'о' < 'у'), 'teacher' > 'pupil', 'паровоз' < 'пароход'.

Сравнивать можно и строки разной длины. Так, строка 'пар' меньше строки 'парад', 'стол' < 'столик', '12' < '2', 'ABC' < 'ADBA'.

Сравним две строки: 'мотоцикл' и 'мотоблок':

```
s1:='мотоцикл'; s2:='мотоблок';
if s1>s2 then write(s1, ' больше, чем ', s2);
```

Этот фрагмент программы выведет на экран надпись: «мотоцикл больше, чем мотоблок».



Пример 4. С клавиатуры вводят два слова. Составить программу, которая проверяет, не начинается ли второе слово с той же буквы, которой оканчивается первое слово.

Алгоритм основан на определении индекса k последнего символа (длины) слова a и проверке выполнения условия $a[k]=b[1]$.

Программа может выглядеть так:

```
program Primer43_4;
var a, b: string; k: integer;
begin
    writeln('Введите два слова'); read(a,b);
                                     {ввод слов}
    k:=length(a);
        {длина (номер последнего символа) слова a}
    if a[k]=b[1] then writeln('Да')
        else writeln('Нет');
end.
```

Результат выполнения программы может выглядеть так:

```
Введите два слова
дом
окно
Нет
Введите два слова
дом
мост
Да
```



1. Какие действия можно выполнять над строковыми переменными?

2. Какую функцию используют для определения длины строки?
3. В чем заключается операция сравнения строк?
4. Какая из двух строк считается большей?
5. Какие утверждения верны?
 - а) 'ДОМ' > 'ДОРОГА';
 - б) 'человек' > 'ЧЕЛОВЕК';
 - в) '123' < '40';
 - г) 'шалаш' > 'дворец'.

Упражнения

1. С клавиатуры вводят названия трех геометрических фигур. Составить программу, которая выводит на экран эти названия, например: «геометрические фигуры: квадрат, ромб, круг».
2. С клавиатуры вводят слово. Составить программу, которая:
 - а) определяет количество символов в нем;
 - б) проверяет, не совпадают ли первая и последняя буквы слова;
 - в) выводит на экран первый, второй и последний символы;
 - г) определяет, четно или нечетно количество символов в слове.

§ 5. ПРОЦЕДУРЫ И ФУНКЦИИ ОБРАБОТКИ СТРОКОВЫХ ПЕРЕМЕННЫХ

Pascal имеет стандартные процедуры и функции для работы со строками (приложение 4). Напомним, что отличие процедуры от функции заключается в том, что функция возвращает результат, который может быть использован в каком-либо другом операторе (присвоить, вычислить, преобразовать и т. п.), а процедура просто выполняет некоторое действие. Так, `length(s)` — это функция, а `write(s)` и `read(s)` — процедуры.

Рассмотрим примеры применения стандартных процедур и функций обработки строковых переменных.

Определение позиции вхождения подстроки

Функция `pos(s1,s)` определяет, начиная с какого символа подстрока `s1` содержится в строке `s`. Если подстрока `s1` в `s` не обнаружена,

то функция возвращает нуль. Так, при выполнении последовательно-сти операторов

```
s1:='трос'; a:=pos(s1,'матрос'); b:=pos(s1,'капитан');
```

в переменной a окажется число 3, а в переменной b — число 0.

Пример 1. С клавиатуры вводят адрес электронной почты. Составить программу, которая определяет, сколько символов содержит имя почтового ящика (логин).

Адрес электронной почты будем вводить в строковую переменную s. Количество символов k в имени почтового ящика определим по позиции в строке s символа @ («собака»): `k:=pos('@',s)-1;`

Программа может выглядеть так:

```
program Primer5_1;
var s: string; k: integer;
begin
  writeln('Введите адрес электронной почты: ');
  read(s);                               {ввод адреса}
  k:=pos('@',s)-1;
  writeln('логин содержит ', k, ' символов');
end.
```

Результат выполнения программы может выглядеть так:

```
Введите адрес электронной почты:
zubrenok@les.by
логин содержит 8 символов
```

Копирование подстроки

Функция `copy(s,p,n)` копирует n символов из строки s, начиная с позиции p, т. е. возвращает часть строки s (подстроку) длиной n символов.

Пример 2. Переменная s содержит строку 'касіў Ясь канюшыну'. Составить программу, которая выводит на экран всю строку и заданную ее часть, например имя 'Ясь'.

Присвоим значение строковой переменной `s:='касіў Ясь канюшыну'`. Функция `s1:=copy(s,7,3);` скопирует в строковую переменную s1 имя «Ясь» (начинается с 7-й позиции, считая пробел, и содержит три буквы).

Программа может выглядеть так:

```

program Primer5_2;
var s, s1: string;
begin
    s:='касіў Ясь канюшыну';    {присвоение значения
                                строковой переменной}
    s1:=copy(s,7,3);            {копирование подстроки}
    writeln(s);                 {вывод строки}
    writeln(s1);                {вывод части строки}
end.

```

Результат выполнения программы может выглядеть так:

```

касіў Ясь канюшыну
Ясь

```

Удаление и вставка символов

Процедура `delete(s,p,n)` удаляет n символов из строки s , начиная с позиции p . Процедура `insert(s1,s,p)` вставляет подстроку $s1$ в строку s , начиная с позиции p .

Пример 3. Составить программу, которая из строки 'Журавли на юг летят' удаляет слова 'на юг' и вставляет слова 'на Полесье'.

Присвоим значения строковым переменным: `s:='Журавли на юг летят'` и `s1:=' на Полесье'`.

Процедура `delete(s,8,6)` удалит из строки s 6 символов, начиная с 8-го. Восьмой и одиннадцатый символы являются пробелами — их тоже удалим, так как пробел — равноправный символ, как и любой другой.

Журавли на юг летят

1 8 11 13

В переменной s останется строка 'Журавли летят'.

В результате выполнения процедуры `insert(s1,s,8)` в переменной s окажется строка 'Журавли на Полесье летят'.

Программа может выглядеть так:

```

program Primer5_3;
var s, s1: string;
begin
    s:='Журавли на юг летят';    {присвоение значений}
    s1:=' на Полесье';           {строковым переменным}

```

```
writeln(s);           { вывод строки }
delete(s,8,6);       { удаление символов }
writeln(s);           { вывод строки }
insert(s1,s,8);      { вставка подстроки s1 в строку s }
writeln(s);           { вывод строки }
end.
```

Результат выполнения программы может выглядеть так:

```
Журавли на юг летят
Журавли летят
Журавли на Полесье летят
```

Преобразование регистра



Для преобразования строчных букв в прописные и наоборот в Pascal предназначены следующие функции: `upcase`, `lowcase` (работают с отдельными символами) и `uppercase`, `lowercase` (работают с целыми строками).

Пример 4. Составить программу, которая изменяет регистр введенной с клавиатуры строки.

Будем вводить строку, содержащую строчные и прописные буквы, в переменную `s` типа `string`.

Используем два способа изменения регистра. Функции `uppercase` и `lowercase` преобразовывают сразу всю строку. Они более просты, однако неприменимы, когда нужно изменить регистр отдельных букв (например, первых).

При использовании функций `upcase` и `lowcase` преобразование и вывод выполняются посимвольно в цикле с параметром `i`, который изменяется от 1 до `length(s)` (последний символ строки). При этом можно задать регистр отдельных символов: выводить четные символы прописными, а нечетные — строчными.

Программа может выглядеть так:

```
program Primer5_4;
var s: string; i: integer;
begin
  write('Введите строку: '); read(s); { ввод строки }
  writeln(uppercase(s));              { вывод строки
                                     в верхнем регистре }
```

```
writeln(lowercase(s));      {вывод строки в нижнем
                             регистре}
for i:=1 to length(s) do   {посимвольное
                             преобразование и вывод}
  if (i mod 2 = 1) then write(uppercase(s[i]))
                             {нечетные в верхнем регистре}
  else write(lowercase(s[i]));
end.
```

Результат выполнения программы может выглядеть так:

```
Введите строку: город Минск
ГОРОД МИНСК
город минск
ГоРоД МиНсК
```

Преобразование типов данных

Число v преобразовывается в строку s процедурой $str(v,s)$.

Процедура $val(s,v,code)$ преобразовывает строковое представление числа s в число v . Если преобразование возможно, то в переменную $code$ возвращается 0, если невозможно, то в $code$ возвращается ненулевое значение.

Пример 5. Составить программу, которая преобразовывает введенные с клавиатуры числа в строки и демонстрирует сложение чисел и сцепление строк.

Введенные с клавиатуры два числа (например, 12 и 15) поместим в целочисленные переменные a и b . Выведем их сумму (в нашем примере $12 + 15 = 27$). Затем процедурами $str(a,sa)$ и $str(b,sb)$ преобразуем их в строки sa и sb . Выведем результат сцепления этих строк: '12'+ '15' = '1215'.

Программа может выглядеть так:

```
program Primer5_5;
var a,b: integer; sa,sb: string;
begin
  write('Введите два числа: '); readln(a,b);
                               {ввод чисел}
  writeln('Сумма чисел: ', a, ' + ', b, ' = ', a+b);
                               {вывод суммы чисел}
  str(a,sa); str(b,sb);       {преобразование в строки}
```

```
writeln('Сцепление строк: ', sa, ' + ',
sb, ' = ',sa+sb);           {вывод строки}
end.
```

Результат работы программы может выглядеть так:

```
Введите два числа: 12 15
Сумма чисел: 12 + 15 = 27
Сцепление строк: 12 + 15 = 1215
```

-  1. С помощью какой функции определяют позицию символа в строке?
- 2. Какую функцию применяют для копирования части строки?
- 3. Какую процедуру применяют для удаления символов из строки?
- 4. Какую процедуру применяют для вставки символов в строку?

Упражнения

1. С клавиатуры одной строкой вводят имя и фамилию. Составьте программу, которая:

- а) определяет, есть ли во введенной строке заданный символ;
- б) копирует в новую строку имя и выводит его;
- в) выводит сначала фамилию, а затем имя.

2. Составьте программу, которая в строке «Касіў Ясь канюшыну» заменяет имя «Ясь» на «Янка».

3. Составьте программу, которая в строке «реки горы и болота»:

- а) удаляет слово «горы»;
- б) заменяет слово «болота» на «озера».

-  4. С клавиатуры вводят строку «город минск» в нижнем регистре. Составьте программу, которая преобразовывает название города к верхнему регистру и выводит: «город МИНСК».

§ 6. ВЫПОЛНЕНИЕ ПРАКТИЧЕСКИХ ЗАДАНИЙ С ИСПОЛЬЗОВАНИЕМ СИМВОЛЬНЫХ И СТРОКОВЫХ ВЕЛИЧИН

Разберем примеры решения некоторых задач с использованием символьных и строковых величин.

Пример 1. Составить программу, которая выводит буквы введенного с клавиатуры слова в обратном порядке с задержкой 200 мс.

Пусть, например, введено слово «привет». При выполнении программы на экране должна появиться сначала последняя буква «т», по-

том предпоследняя буква «е» и т. д. Таким образом, мы должны сначала определить, сколько букв в слове — `length(s)`, а затем выбирать из заданной строки по одной букве, начиная с последней, и выводить их на экран в цикле `for i:=length(s) downto 1`. Задержка в 200 миллисекунд устанавливается процедурой `sleep(200)`;

Программа может выглядеть так:

```
program Primer6_1;
var s: string; i: integer;
begin
  writeln('Введите слово'); readln(s);      {ввод слова}
  for i:=length(s) downto 1 do begin
    write(s[i]);                            {вывод символов}
    sleep(200);                             {задержка 200 мс}
  end;
end.
```

Результат работы программы может выглядеть так:

```
Введите слово
привет
тевирп
```



Используя рассмотренный алгоритм вывода букв слова в обратном порядке, можно составить программу для проверки, является ли введенное слово перевертышем, т. е. читается ли оно одинаково слева направо и справа налево, как, например, слово «заказ».

Пример 2. С клавиатуры вводят строку и букву. Составить программу, которая подсчитывает, сколько раз указанная буква встречается в строке.

Алгоритм основан на последовательном сравнении каждого символа строки `st[i]` с введенной буквой `b` в цикле `for` с параметром `i`, который изменяется от 1 (первый символ) до `length(st)` (последний символ). При выполнении условия `st[i]=b` счетчик совпадений увеличивается на 1: `k:=k+1`;

Программа может выглядеть так:

```
program Primer6_2;
var st: string; b: char; i, k: integer;
begin
```

```

write('Введите строку: '); read(st); {ввод строки}
write('Введите букву: '); read(b);   {ввод буквы}
k:=0;                                {обнуление счетчика совпадений}
for i:=1 to length(st) do
  if st[i]=b then k:=k+1;           {подсчет совпадений}
write('Ответ: ',k);
end.

```

Результат работы программы может выглядеть так:

```

Введите строку: математика и матанализ
Введите букву: а
Ответ: 6

```



Пример 3. Составить программу, которая определяет количество гласных во введенном с клавиатуры слове.

Алгоритм основан на последовательном сравнении всех символов $s[i]$ введенного слова s с хранимой в переменной g строкой всех гласных (прописных и строчных). В цикле `for` с параметром i перебираются все буквы введенного слова от первой до последней: $i:=1$ to $length(s)$. Если очередная буква слова есть в строке гласных g , т. е. если выполняется условие $pos(s[i],g) <> 0$, счетчик гласных увеличивается на 1: $k:=k+1$;

Программа может выглядеть так:

```

program Primer6_3;
var s, g: string; i, k: integer;
begin
  g:= 'аеиоуыэюяАЕИОУЫЭЮЯ';           {строка гласных}
  write('Введите слово: '); read(s); {ввод слова}
  k:=0;                                {обнуление счетчика гласных}
  for i:=1 to length(s) do             {перебор букв слова}
    if pos(s[i],g) <> 0                 {сравнение со строкой
                                        гласных}
      then k:=k+1;
  writeln('гласных ', k);
end.

```

Результат выполнения программы может выглядеть так:

```

Введите слово: информатика
гласных 5
Введите слово: компьютер
гласных 3

```



Пример 4. Составить программу, которая подсчитывает количество слов во введенной с клавиатуры строке.

Между словами, в начале и конце строки может оказаться несколько пробелов. Поэтому окончанием слова будем считать символ, отличный от пробела, если за ним следует пробел. А чтобы посчитать последнее слово и в случае, когда после него пробел не введен, добавим пробел в конец строки: `st:=st+' '`;

Предлагаемый алгоритм основан на последовательном переборе всех пар рядом стоящих символов строки `st[i]` и `st[i+1]` в цикле `for` с параметром `i`, который изменяется от 1 (первый символ) до `length(st)` (последний символ). При выполнении условия `(st[i]<>' ') and (st[i+1]=' ')` значение счетчика слов увеличиваем на 1.

Программа может выглядеть так:

```
program Primer6_4;
var st: string; i,n: integer;
begin
  writeln('Введите строку: '); read(st); {ввод строки}
  st:=st+' '; {добавление пробела}
  n:=0; {обнуление счетчика слов}
  for i:=1 to length(st) do
    if (st[i]<>' ') and (st[i+1]=' ') then n:=n+1;
    {подсчет слов}
  writeln('В строке ',n,' слов');
end.
```

Результат работы программы может выглядеть так:

```
Введите строку:
мама папа брат и я
В строке 5 слов
```

Пример 5. Составить программу, которая выводит на экран заданную часть введенной с клавиатуры строки.

Строку вводим в переменную `s` типа `string`. Ввод позиции начала и длины копируемой подстроки в переменные `p` и `n` типа `integer` будем выполнять в одной процедуре `read(p,n)` через пробел.

Программа может выглядеть так:

```
program Primer6_5;
```

Правообладатель Издательский центр БГУ

```
var s: string; p,n: integer;
begin
  writeln('Введите строку: '); read(s); {ввод строки}
  write('Введите позицию начала и длину подстроки: ');
  read(p,n);
  writeln(copy(s,p,n));           {вывод части строки}
end.
```

Результат выполнения программы может выглядеть так:

```
Введите строку: квадрат и прямоугольник
Введите позицию начала и длину подстроки: 16 4
угол
```

Пример 6. С клавиатуры вводят слово. Составить программу, которая производит замену указанного слога на новый.

Введем слово в переменную *s* и два слога: заменяемый в переменную *s1* и заменяющий в переменную *z*. Определим позицию $p:=pos(s1,s)$ и длину $len:=length(s1)$ заменяемого слога. Удалим этот слог $delete(s,p,len)$ и вставим новый $insert(z,s,p)$.

Программа может выглядеть так:

```
program Primer6_6;
var s,s1,z: string; p,len: integer;
begin
  write('Введите слово: '); read(s);           {ввод слова}
  write('Заменить слог: '); read(s1);         {ввод
                                             заменяемого слога s1}
  write('на слог: '); read(z); {ввод заменяющего слога z}
  p:=pos(s1,s); len:=length(s1);             {позиция и длина
                                             слога s1}
  delete(s,p,len);                           {удаление части слова s}
  insert(z,s,p);                              {вставка слога z в слово s}
  writeln(s);                                 {вывод}
end.
```

Результат выполнения программы может выглядеть так:

```
Введите слово: пароходик
Заменить слог: ход
на слог: воз
паровозик
```

Пример 7. Составить программу, которая кодирует введенное с клавиатуры кириллицей слово, заменяя буквы их кодами.

Простейший алгоритм кодирования заключается в замене каждой буквы $s[i]$ ее числовым кодом $\text{ord}(s[i])$ в цикле с параметром i , который изменяется от 1 до $\text{length}(s)$ (последняя буква).

Программа может выглядеть так (пробелы между трехзначными кодами букв вставлены для наглядности):

```
program Primer6_7;
var s: string; i: integer;
begin
  write('Введите слово: '); read(s);      {ввод слова}
  for i:=1 to length(s) do write(ord(s[i]), ' ');
end.
```

Результат выполнения программы может выглядеть так:

```
Введите слово: привет
239 240 232 226 229 242
```



Пример 8. Составить программу, которая выводит латинскими буквами слово, введенное кириллицей.

Такой прием применяется при отправке электронных писем адресатам, на компьютерах которых может не оказаться кириллицы.

Алгоритм перекодирования заключается в замене символов одного алфавита на соответствующие символы другого. Зададим алфавиты в виде строк a и b . Пусть во избежание неоднозначности они содержат по 23 строчные буквы. Замену символов выполним в цикле с параметром i , который изменяется от 1 до $\text{length}(s)$ (последняя буква слова). Соответствие символов обеспечивается общим индексом (номером) символа в обоих алфавитах.

Программа может выглядеть так:

```
program Primer6_8;
var a,b,s: string; i,j: integer;
begin
  a:='абвгдежзийклмнопрстуфхц'; {исходный алфавит}
  b:='abvgdejzijklmnoprstufhc'; {новый алфавит}
  writeln('Введите слово: '); read(s);
  for i:=1 to length(s) do begin
    j:=pos(s[i],a); {индекс символа в старом алфавите}
```

```

        write(b[j]);    {вывод символа в новом алфавите}
    end;
end.

```

Результат выполнения программы может выглядеть так:

```

Введите слово:
привет
privet

```

В 9-м классе вы познакомились с массивами чисел. Строки также могут быть элементами массива.

Пример 9. С клавиатуры вводят число. Составить программу, которая выводит словами цифры этого числа.

Алгоритм заключается в выводе элементов массива $N[j]$, в котором хранятся названия цифр. Индекс элемента массива j определяется по позиции $j := \text{pos}(c[i], a)$, занимаемой символом $c[i]$, соответствующим очередной цифре, в строке $a := '1234567890'$.

Программа может выглядеть так:

```

program Primer6_9;
const n: array[1..10] of string=('один','два','три',
'четыре','пять','шесть','семь','восемь','девять','нуль');
var b,c: string; i,j: integer;
begin
    b:='1234567890';
    write('Введите число: '); read(c);
    for i:=1 to length(c) do begin
        j:=pos(c[i],b); write(n[j],' ');
    end;
end.

```

Результат выполнения программы может выглядеть так:

```

Введите число: 703164
семь нуль три один шесть четыре

```



Пример 10. В двух массивах строковых констант хранятся слова — существительные и глаголы. Составить программу «Перепутанные слова», которая выводит пары этих слов случайным образом.

По введенному с клавиатуры номеру (индексу) из массива a выбирается слово (существительное, обозначающее человека или животное). Алгоритм этой игры-шутки заключается в случайном выборе слова из массива b (глагола, обозначающего действия). Напомним, что

результатом выполнения функции `random(8)+1` будет случайное целое число от 1 до 8.

Программа может выглядеть так:

```
program Primer6_10;
const a: array[1..8] of string = ('ученик', 'артист',
'художник', 'рыба', 'птица', 'кошка', 'собака', 'корова');
const b: array[1..8] of string = ('читает', 'поет',
'рисует', 'плавает', 'летает', 'мяукает', 'лает', 'мычит');
var k: integer;
begin
  write('Введите число от 1 до 8: '); read(k);
  writeln(a[k]+' '+b[random(8)+1]);      {вычисление
                                          и вывод}
end.
```

Результат выполнения программы может выглядеть так:

```
Введите число от 1 до 8: 5
птица мяукает
```



Пример 11. В двух массивах строковых констант хранятся слова на русском и белорусском языках. Составить программу «Словарь», которая тестирует знание перевода слов.

Алгоритм основан на проверке соответствия введенного перевода слова и хранящегося в массиве `b` эталона.

Программа может выглядеть так:

```
program Primer6_11;
const a: array[1..8] of string = ('ученик', 'доска',
'рисунок', 'птица', 'обезьяна', 'петух', 'неделя', 'луна');
const b: array[1..8] of string = ('вучань', 'дошка',
'малюнак', 'птушка', 'малпа', 'певень', 'тыдзень', 'месяц');
var i,j: integer; s:string;
begin
  j:=random(8)+1;
  write('Увядзіце пераклад слова: ', a[j], ' - ');
  read(s);
  if s=b[j] then writeln('добра!')
  else writeln('памылка, трэба - ', b[j]);
end.
```

Результат выполнения программы может выглядеть так:

```

Увядзіце пераклад слова: доска – дошка
добра!
Увядзіце пераклад слова: птуца – птаха
памылка, трэба – птушка
    
```

Упражнення

1. С клавиатуры вводят слово и символ. Составьте программу, которая:
 - а) проверяет, есть ли в этом слове заданный символ;
 - б) определяет позицию заданного символа;
 - в) подсчитывает количество вхождений заданного символа.
2. С клавиатуры вводят строку и слово. Составьте программу, которая:
 - а) определяет позицию заданного слова;
 - б) удаляет заданное слово;
 - в) вставляет в строку заданное слово.
3. Составьте программу, которая:
 - а) из строки «Волга, Днепр, Неман, Енисей» удаляет названия рек, не протекающих по территории Беларуси;
 - б) из строки «Брест, Бобруйск, Витебск, Гомель, Мозырь, Могилев» удаляет названия городов, не являющихся областными центрами.
4. Составьте программу, которая:
 - а) к строке «край озер» добавляет слова «лесов и рек»;
 - б) в строке «Я учусь в школе» заменяет слова «в школе» на «в 10-м классе».
5. Составьте программу, которая переставляет на первое место:
 - а) в строке «моя родина Беларусь» слово «Беларусь»;
 - б) в строке «люблю цябе, мой родны кут» слова «мой родны кут».
-  6. Составьте программу, которая меняет местами первое и последнее слово:
 - а) в строке «Минск столица Беларуси»;
 - б) в строке «Беларусь ты моя синеокая».
-  7. С клавиатуры вводят строку. Составьте программу, которая:
 - а) заменяет все пробелы в строке символом подчеркивания;
 - б) подсчитывает количество букв в строке (исключая пробелы и цифры);
 - в) подсчитывает количество прописных букв.

ИНФОРМАЦИОННЫЕ МОДЕЛИ

§ 7. ПОНЯТИЕ И НАЗНАЧЕНИЕ ИНФОРМАЦИОННОЙ МОДЕЛИ

Под *моделью* (лат. *modulus* — аналог, образец) понимается некоторый материальный либо мысленно представляемый объект или явление, используемые вместо другого объекта (оригинала). Модель повторяет существенные для целей конкретного моделирования свойства оригинала, опуская несущественные свойства.

Модели могут быть разделены на два больших класса: материальные и информационные.

Материальная (предметная) модель воспроизводит геометрические, физические, химические, биологические свойства объектов в материальной форме. Примерами материальных моделей являются: глобус, макет застройки микрорайона, чучело животного, игрушки, изображающие людей или машины.

Информационная модель — это совокупность информации, описывающая существенные свойства и состояния объекта, процесса, явления. Информационные модели нельзя потрогать, они не имеют материального воплощения, потому что строятся только на информации. В то же время, рассматривая любую информационную модель, мы связываем ее с определенным носителем информации (бумагой, видеопленкой, магнитным диском и пр.).

Существует несколько классификаций информационных моделей.

Знаковые информационные модели описывают объект или явление на каком-либо естественном или формальном языке, а *образные* (графические) модели описывают зрительные образы (в виде рисунков, фотографий, схем).

Описательные информационные модели создаются на естественном языке (т. е. на любом языке общения между людьми: русском, белорусском, китайском, английском и т. п.) в устной или письменной форме. *Формальные* модели создаются на формальном языке (т. е. на научном, профессиональном или специализированном). Примеры формальных моделей: все виды формул, таблицы, графики, диаграммы, графы, карты, схемы и т. д.

Для более полного описания свойств или характеристик какого-либо объекта может использоваться несколько информационных моделей разных видов (рис. 4.1).

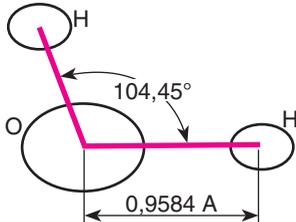
Вид модели	Модель
Словесное описание	Вода (оксид водорода) — прозрачная жидкость, не имеющая цвета (в малом объеме) и запаха.
Формула	H_2O
Изображение молекулы	
Схема молекулы	

Рис. 4.1. Описание воды как химического вещества с помощью различных моделей

Человек на протяжении всей жизни составляет словесные описания предметов, объектов, ситуаций, происшествий на естественном языке. При составлении словесной модели необходимо ясно и понятно формулировать предложения, использовать проверенные факты, нужные понятия и термины. Примерами описательных моделей является информация в учебниках, произведения художественной литературы, сводки происшествий, а также словесные алгоритмы — описания последовательности действий и процессов.

Для создания описательных моделей на компьютере используют текстовые редакторы, например MS Word, в котором набирают текст и задают его оформление: тип шрифта, размер, начертание, абзацный отступ и т. д.

Формальные информационные модели содержат математические и химические формулы, алгоритмы, представленные на языках програм-

мирования, и т. д. Например, формулы математики описывают соотношения между количественными характеристиками объекта моделирования.

Математической моделью называется совокупность математических соотношений, уравнений, неравенств, описывающих основные закономерности изучаемого объекта, процесса или явления.

Информационные модели, описывающие объекты, явления, процессы в определенный момент времени, без учета их изменений в пространстве и времени, называются *статическими*. Такими моделями являются, например, структура кристаллов, классификация растений или животных. *Динамические* модели учитывают изменения параметров процессов и явлений с течением времени.

Процесс создания и использования моделей для решения практических задач называют **моделированием**. Моделирование широко используется в науке и технике, экономике и производстве. Без моделирования невозможно создание машин и механизмов, строительство зданий и мостов, создание новых материалов, лекарств, торговых сетей.

Использование моделей позволяет упростить и удешевить исследование объектов и явлений реального мира. Основное назначение информационных моделей — описание свойств объектов, установление закономерностей, проектирование новых объектов, прогнозирование протекающих процессов и эффективное управление ими.



1. Что понимается под моделью?
2. На какие два класса разделяют модели?
3. Какие модели называют материальными? Информационными? Приведите примеры.
4. Какие модели называют знаковыми? Образными? Описательными? Формальными? Приведите примеры.
5. Какой процесс называют моделированием?

Упражнения

1. Определите, какие из перечисленных моделей являются информационными, а какие — материальными: а) прогноз погоды в сети Интернет; б) макет застройки агрогородка; в) расписание движения автобусов; г) карта метрополитена; д) рисунок родословного дерева;

е) список учеников класса; ж) компьютерная программа; з) модель самолета.

2. Какие из описаний представлены на формальном языке, а какие — на естественном языке?

<p>а)</p>	<p>в)</p> <table border="1"> <tr> <td></td> <td></td> <td></td> </tr> <tr> <td>$i(t) = \frac{u(t)}{R}$</td> <td>$i(t) = C \frac{du(t)}{dt}$</td> <td>$i(t) = \frac{1}{L} \int u(t) dt$</td> </tr> <tr> <td>$u(t) = Ri(t)$</td> <td>$u(t) = \frac{1}{C} \int i(t) dt$</td> <td>$u(t) = L \frac{di(t)}{dt}$</td> </tr> </table>				$i(t) = \frac{u(t)}{R}$	$i(t) = C \frac{du(t)}{dt}$	$i(t) = \frac{1}{L} \int u(t) dt$	$u(t) = Ri(t)$	$u(t) = \frac{1}{C} \int i(t) dt$	$u(t) = L \frac{di(t)}{dt}$
$i(t) = \frac{u(t)}{R}$	$i(t) = C \frac{du(t)}{dt}$	$i(t) = \frac{1}{L} \int u(t) dt$								
$u(t) = Ri(t)$	$u(t) = \frac{1}{C} \int i(t) dt$	$u(t) = L \frac{di(t)}{dt}$								
<p>б)</p> <p style="text-align: center;">Химические знаки</p> <p style="text-align: center;"><chem>H2SO4</chem></p> <p style="text-align: center;">Индексы</p>	<p>г)</p> <p>Теплохід перебуває в доці Запорізького судноремонтного заводу для проведення термінових ремонтних робіт.</p>									

3. Какого вида информационные модели присутствуют в представленном ниже учебном материале?

Задача	
<p>Через неподвижный блок перекинута нить, к концам которой подвешены грузы массой 3 и 1,9 кг. Найти ускорение грузов и силу натяжения нити. Считать, что трение в блоке отсутствует. Массой нити и блока пренебречь.</p>	
Решение	
	<p>На каждый груз действуют две силы: mg — сила тяжести и T — сила натяжения нити. Если массой нити и блока пренебречь, то силы натяжения, приложенные к грузу, можно считать равными друг другу.</p> <p>Запишем уравнения движения грузов:</p> $\begin{cases} m_1 \bar{a}_1 = m_1 \bar{g} + \bar{T}; \\ m_2 \bar{a}_2 = m_2 \bar{g} + \bar{T}. \end{cases}$

§ 8. СТРУКТУРИРОВАНИЕ ИНФОРМАЦИИ С ИСПОЛЬЗОВАНИЕМ ИНФОРМАЦИОННЫХ МОДЕЛЕЙ

Рассмотрим структурирование информации с использованием информационных моделей разного типа.

Наибольшее распространение получили табличные модели, чему в немалой степени способствовала простота их реализации в электронных таблицах.

В табличной модели названия или перечни однотипных объектов или свойств размещают в первой строке (или столбце) таблицы, а значения их свойств — в следующих строках (или столбцах) таблицы.

Столбцы с однотипными данными называют *полями*, а каждая строка представляет отдельную *запись*.

Различают таблицы типа «объект — свойство» — когда в одной строке содержится информация об одном объекте или одном событии — и таблицы типа «объект — объект», которые отражают взаимосвязи между разными объектами.

Пример 1. Разместить в таблицу информацию об осадках и среднесуточной температуре за 5 дней марта.

Однотипные данные будем размещать в столбцах. Поместим в первую строку названия столбцов (полей): Дата, Осадки, Температура. Таблица может выглядеть так:

Дата	Осадки	Температура
1.03	Снег	–5
2.03	Нет	–7
3.03	Нет	–2
4.03	Дождь	0
5.03	Дождь	+2

Можно разместить названия в первом столбце, а однотипные данные группировать по строкам. В этом случае таблица будет выглядеть так:

Дата	1.03	2.03	3.03	4.03	5.03
Осадки	Снег	Нет	Нет	Дождь	Дождь
Температура	–5	–7	–2	0	+2

Пример 2. Разместить в таблицу информацию об успеваемости трех учащихся по двум предметам.

Поместим в первую строку названия столбцов (полей): Ученик, Информатика, Физика.

Ученик	Информатика	Физика
Иванов	8	7
Петров	9	6
Сидоров	9	6

Для наглядного представления состава и структуры данных используют *графы*. Моделируемые объекты представляют как вершины (или узлы) графа, а связи между объектами — как дуги (или ребра) графа. Получают чертеж, который задает множество вершин и множество линий (связей), соединяющих некоторые пары вершин.

Связи между объектами называются *отношениями*. Симметричная связь обозначается отрезком, а несимметричная — стрелкой. В ненаправленном графе связи симметричны. В виде такого графа можно представить модель дорог, связывающих населенные пункты (рис. 4.2):



Рис. 4.2

В ориентированном графе отношения между объектами (связи) несимметричны, например: начальник → подчиненный, отец → сын, бабушка → внука (рис. 4.3).



Рис. 4.3

Система, объекты которой находятся друг с другом в отношении вложенности или подчиненности, называется *иерархической*. Иерархическая модель наглядно представляется графом, в котором вершины верхнего уровня связаны с вершинами нижнего уровня как «один ко многим». Примерами иерархических моделей являются графические представления разнообразных классификаций (животных, растений, типов информационных моделей (рис. 4.4)), размещение файлов и папок на жестком диске компьютера и т. п.



Рис. 4.4

Информационная модель, объекты которой (вершины графа различных уровней) связаны между собой по принципу «многие ко многим», называется *сетевой*. Примерами сетевых моделей могут служить различные сети (компьютеров, магазинов), глобальная сеть Интернет.

С помощью информационных моделей можно представлять знания.

Семантическая модель — модель знаний в форме графа, в основе которой лежит идея о том, что любые знания можно представить в виде совокупности объектов (понятий) и связей (отношений) между ними. В качестве примера представим в виде графа связи в предложении «Стаїць мая хата між хвой кучаравых» (рис. 4.5).

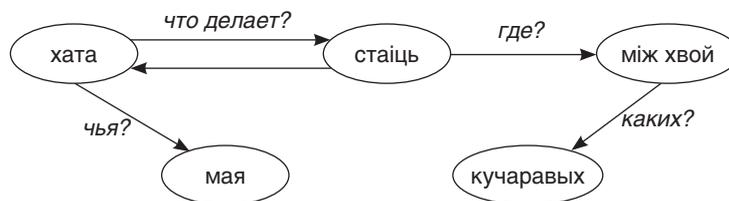


Рис. 4.5



1. Что понимают под полями и записями таблицы?
2. Какая информационная модель называется иерархической?

Упражнения

1. Структурируйте данные о погоде за прошедшую неделю и разместите их в форме таблицы, например:

Дата	Температура, °С	Давление, мм рт. ст.	Влажность, %
29.06.11	26	745	79
30.06.11	28	747	73
1.07.11	30	748	68
2.07.11	24	744	78
3.07.11	25	745	77

2. Представьте в виде иерархической модели предлагаемые учителем данные.



3. Представьте в виде графа:

а) состав вашей семьи; б) несколько населенных пунктов вашей местности; в) состав вашего класса.

§ 9. КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ

Появление компьютера обеспечило компьютерную реализацию информационных моделей, которая предполагает проведение вычислительного эксперимента и осуществление прогнозирования. Компьютерные модели незаменимы в тех случаях, когда реальные эксперименты невозможны или затруднены из-за финансовых или физических препятствий (например, в атомной и ядерной физике, астрофизике).

Логичность и формализованность компьютерных моделей позволяют выявить основные факторы, которые определяют свойства изучаемого объекта, в частности, исследовать отклик моделируемой системы на изменения ее параметров и начальных условий. Современные компьютеры позволяют строить весьма сложные модели, достаточно полно отражающие реальные объекты или процессы.

Построение компьютерной модели базируется на абстрагировании от конкретной природы явлений или изучаемого объекта. Выделяют следующие основные этапы компьютерного моделирования.

1. Постановка задачи: описание объекта и определение цели моделирования.

2. Разработка плана создания модели. Выделение свойств объекта, существенных для данной задачи, и отбрасывание второстепенных. Выбор формы представления модели (например, таблица) и необходимого инструментария (например, системы программирования).

3. Создание модели: формализация, т. е. переход к математической модели; создание алгоритма и написание программы.

4. Анализ модели на соответствие объекту-оригиналу.

Рассмотрим этапы построения компьютерной модели на примере.

Пример. Через сколько дней больной выздоровеет, т. е. концентрация болезнетворных бактерий в его крови уменьшится с начального значения (вводится с клавиатуры) до 12 единиц, если в результате применения лекарства концентрация бактерий ежедневно уменьшается на 20 % по сравнению с предыдущим днем?

1. Объектом моделирования является концентрация болезнетворных бактерий в крови больного. Цель — сделать прогноз, через сколько дней эта концентрация уменьшится до 12 единиц.

2. При данной постановке задачи будем учитывать только изменение концентрации и пренебрегать остальными свойствами объекта, например, влиянием на кровь температуры больного или рациона его питания.

В качестве формы представления модели выберем числовую форму, а в качестве инструментария реализации этой модели — систему программирования Pascal ABC.

3. Создание компьютерной модели начнем с построения математической модели изучаемого явления.

Поскольку каждый день концентрация бактерий уменьшается на p % по сравнению с концентрацией c предыдущего дня, т. е. на $c \cdot p / 100$, то ее можно выразить формулой: $c - c \cdot p / 100$.

Теперь составим алгоритм решения.

Будем хранить значение концентрации в любой день в переменной c , процент ежедневного уменьшения и безопасное значение в переменных p и cb , количество дней — в переменной t . Переменные c и cb имеют тип `real`, а процент p и количество дней t — тип `integer`.

Начальное значение концентрации будем вводить с клавиатуры (в переменную c). Вычисления будут повторяться в цикле `while`, пока

выполняется условие $c \geq cb$, т. е. пока не будет достигнута безопасная концентрация. В результате получим целое число дней.



Реализация этого алгоритма, т. е. программа на языке программирования Pascal, может выглядеть так:

```
program Model;  
var c, cb: real; p, t: integer;  
begin  
  write('Начальная концентрация: '); read(c); {ввод}  
  p:=20; cb:=12; t:=0;  
  while c>=cb do begin  
    c:=c-c*p/100; t:=t+1;  
  end;  
  writeln('Время: ', t, ' дней')  
end.
```

4. Протестируем модель. Будем вводить различные начальные значения концентрации бактерий. Результат работы программы может выглядеть так:

```
Начальная концентрация: 50  
Время: 7 дней  
Начальная концентрация: 80  
Время: 9 дней
```

Мы рассмотрели пример простейшей модели. Полученные в процессе выполнения программы результаты представляются достоверными.

При анализе более сложной модели необходимо выполнять проверку достоверности результатов. Так, для рассмотренного примера рекомендуется проверять, как изменяется концентрация бактерий, например, через каждый час. Полезно использовать графические формы представления результатов (графики зависимостей, диаграммы).



1. Что понимают под компьютерным моделированием?
2. В каких случаях компьютерные модели являются незаменимыми?
3. Каковы основные этапы компьютерного моделирования?

Упражнения

1. Предложите этапы построения компьютерной модели для решения одной из следующих задач.

а) В искусственный водоем запустили 100 000 рыб. Ежегодно из водоема забирают 12 % рыб. Естественный прирост составляет 20 %. Какое количество рыб будет в водоеме через 8 лет? Через сколько лет в водоеме будет 300 000 рыб, если начиная с пятого года из водоема стали ежегодно забирать еще 5000 рыб?

б) Пассажир опоздал на пароход и решил догнать пароход на такси по дороге, которая идет вдоль берега реки. Средняя скорость автомобиля 100 км/ч, а парохода — 60 км/ч. Догонит ли такси пароход в следующем порту, который находится на расстоянии 120 км от места отправления, если известно, что движение пассажир начал через 20 минут после отправления парохода? Будет ли пассажир ждать пароход в следующем порту? Если да, то какое время? Если нет, то на сколько он опоздает?



2. Составьте программу, реализующую алгоритм решения одной из этих задач.

ОБРАБОТКА ИНФОРМАЦИИ В ЭЛЕКТРОННЫХ ТАБЛИЦАХ

§ 10. ПОНЯТИЕ ЭЛЕКТРОННОЙ ТАБЛИЦЫ

Компьютер позволяет обрабатывать разнообразную информацию, которая может представляться в различных формах. Для повышения наглядности и удобства работы с большим количеством данных их размещают в таблицах.

С древнейших времен люди оформляли информацию в виде таблиц. Вавилонские и египетские жрецы составили множество таблиц, в которых упорядочена информация разного характера и значения: от астрономической (восходы Солнца, фазы Луны) до торговой и бытовой (разливы рек, урожаи, продажи зерна, расходы и доходы). В арабском мире появились математические таблицы, позволявшие делать вычисления (например, тригонометрические таблицы).

И в современной повседневной жизни таблицы широко используются для упорядочения информации: в виде таблицы оформляют спортивные результаты, оценки в школьном журнале, меню в школьной столовой. А с появлением компьютеров стали широко использоваться электронные таблицы.

Электронная таблица (ЭТ) — совокупность хранимых в памяти компьютера данных, которые отображаются в виде таблицы. Основная отличительная особенность ЭТ заключается в том, что хранимые в ней данные могут быть связаны формулами и пересчитываться автоматически. Таким образом, электронная таблица представляет собой табличную модель структурирования, обработки и отображения информации.

10.1. НАЗНАЧЕНИЕ ТАБЛИЧНОГО ПРОЦЕССОРА

Прикладные программы, предназначенные для работы с электронными таблицами, называют **табличными процессорами**. Они позволяют решать следующие основные типовые задачи:

- создание ЭТ;

- открытие (загрузку из файла), редактирование и сохранение ЭТ в файле;
- выполнение вычислений, анализ данных, автоматический пересчет значений при изменении данных;
- графическое представление информации;
- печать ЭТ.

Появление электронных таблиц совпадает с началом распространения персональных компьютеров. Первая программа для работы с электронными таблицами VisiCalc была создана в 1979 г. Современные табличные процессоры позволяют не только выполнять численные расчеты, но и обрабатывать данные других типов — например, сортировать или фильтровать данные текстового типа, упорядочивать даты и т. п.

Одним из самых популярных табличных процессоров является MS Excel, входящий в состав пакета Microsoft Office.

С помощью Excel можно решать множество задач в различных сферах деятельности человека — от простейших вычислений или планирования семейного бюджета до сложных технико-экономических расчетов и представления данных в виде графиков и диаграмм.

Запускается Excel так же, как и другие программы, — кнопкой **Пуск** на Панели задач → **Программы** или щелчком мышью по ярлыку Microsoft Excel  на рабочем столе.

Файлы MS Excel имеют расширение .xls и соответствующие значки: .

10.2. СТРУКТУРА ТАБЛИЦЫ: ЯЧЕЙКИ, СТОЛБЦЫ, СТРОКИ

Информация в электронной таблице может храниться в сотнях ячеек, поэтому необходимо позаботиться о ее правильной организации.

При запуске Excel открывается окно с *рабочим листом* электронной таблицы (рис. 5.1). Окно содержит заголовок, панели инструментов, строку формул, строку состояния. Совокупность листов составляет *рабочую книгу*. При первом запуске на экране по умолчанию появляется **Книга1**, в которой отображается **Лист1** (его фрагмент).

Excel позволяет назначать листам имена, добавлять к книге новые листы и удалять ненужные. С помощью ярлыков листов можно пере-

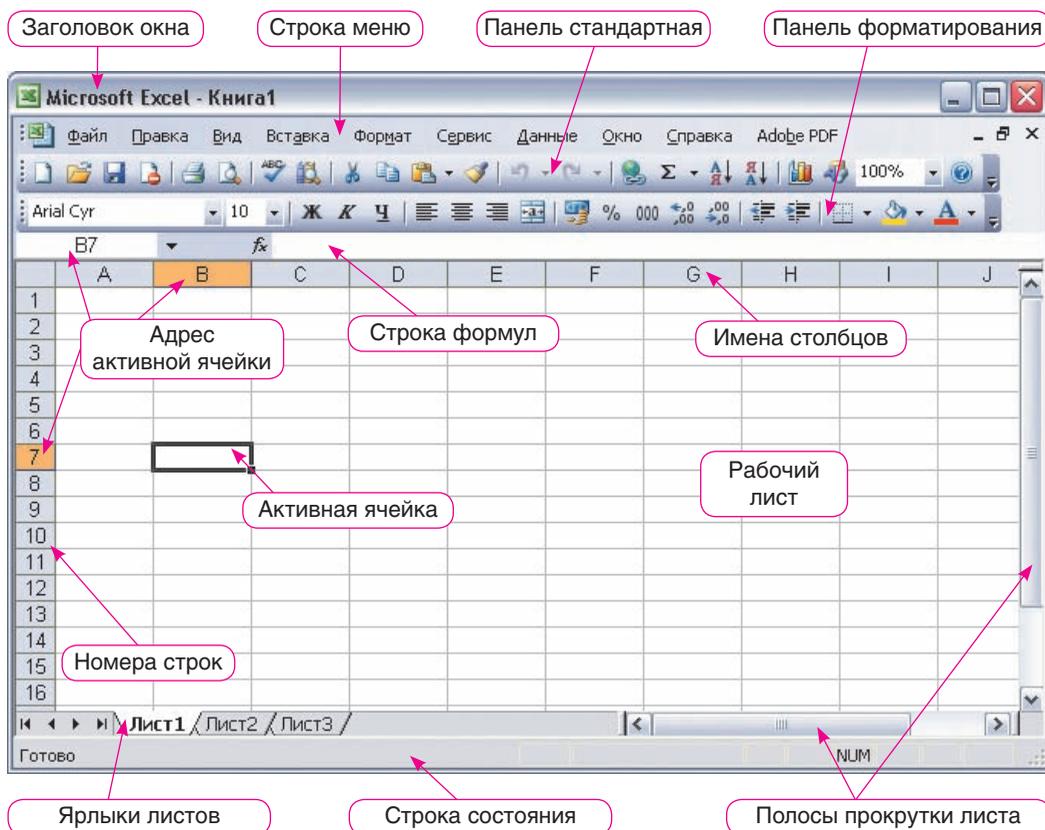


Рис. 5.1. Основные элементы интерфейса MS Excel

ходить от одного листа к другому. В окне имеются полосы прокрутки, с помощью которых можно перемещаться из одного места активного листа в другое.

Лист MS Excel 2003 содержит 65 536 строк и 256 столбцов. Строки пронумерованы числами (от 1 до 65 536), а столбцы обозначены буквами латинского алфавита. Одной буквы хватает для обозначения только первых 26 столбцов от А до Z. Поэтому столбцы с 27-го по 256-й обозначены двумя буквами: AA, AB, ..., AZ, BA, BB, ..., BZ, ..., IA, IB, ..., IV.

На пересечении строк со столбцами расположены *ячейки*, в которых размещаются данные. Каждая ячейка имеет уникальный *адрес*, который состоит из обозначения (имени) столбца и номера строки.

Например, ячейка с адресом В3 находится на пересечении столбца В и строки 3 (рис. 5.2, а).

	A	B	C
1			
2			
3			
4			
5			

Рис. 5.2, а

	A	B	C	D
1				
2				
3				
4				
5				
6				
7				

Рис. 5.2, б

Совокупность нескольких ячеек образует **диапазон**. Адрес диапазона задается адресами его верхней левой и нижней правой ячеек, например В2 и D5 (рис. 5.2, б), которые записывают через двоеточие В2:D5. Диапазон могут составлять и несмежные ячейки, которые разделяют точками с запятой, например: В2:D4; F3:G5; С7:F7 (рис. 5.3).

	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							

Рис. 5.3

Ввод и редактирование данных выполняют в **активной ячейке**. Чтобы сделать ячейку активной, ее необходимо выделить щелчком мыши (или перейти на нее с помощью клавиш перемещения курсора). Активная ячейка обрамляется рамкой, а ее адрес (имя) отображается в поле имен (В3 на рис. 5.2, а).

Когда указатель мыши находится в поле листа, он заменяется **указателем ячейки** в виде белого крестика \oplus .

Для просмотра части листа, которая в данный момент не видна в окне книги, нужно перетащить мышью бегунок на вертикальной или

горизонтальной полосе прокрутки рабочего листа или щелкнуть мышью по свободной зоне полосы прокрутки. Для перемещения между листами книги можно щелкать мышью по их ярлыкам или по кнопкам прокрутки листов, расположенным в левой нижней части листа.

10.3. ТИПЫ ДАННЫХ

В электронной таблице вся информация размещается в ячейках. В ячейку таблицы можно вводить различные данные: *числа*, *текст* или *формулу*. Возможности работы с данными определяются их *типом*.

Excel работает с данными следующих типов:

- числовые значения (например: 27; 5,7; $5\frac{1}{4}$; 1,3E–03);
- текстовые значения (например: Итого; Школа №; 10-й класс; Фамилия);
- дата и время суток (например: Январь 2010; 7.12.2011; 18:00 или 6:00 PM);
- примечания (например: это лидер по прыжкам в высоту);
- формулы (например: $=A3*B3/D2$, $=СУММ(B3:B7)$);
- изображения и объекты мультимедиа из файлов или коллекций клипов, фигуры и объекты WordArt;
- гиперссылки на ресурсы Интернета, а также на другие документы.

Данные вводятся в ячейки пользователем и могут быть изменены путем редактирования. Значения в ячейках, содержащих формулы, *изменяются автоматически*, как только изменяются входящие в формулу исходные данные. Для этого в формулах используют не сами исходные данные, а *ссылки* на адреса ячеек, в которых эти данные находятся.

По умолчанию в ячейках с формулами отображаются не сами формулы, а результаты вычислений по ним. При этом сама формула отображается в строке формул (рис. 5.4).

	А	В	С	Д
1			стоимость	
2		тетрадь	ручка	всего
3		1200	1500	2700
4				

Рис. 5.4



Можно перейти в режим отображения формул (вызвав меню **Сервис** → **Параметры** и установив на вкладке **Вид** в разделе **Параметры окна** флажок **Формулы**). В этом режиме в ячейке отображается не результат, а формула, причем используемые в ней адреса ячеек и рамки вокруг этих ячеек имеют разные цвета (рис. 5.5).

D3		fx =B3+C3		
	A	B	C	D
1			стоимость	
2		тетрадь	ручка	всего
3		1200	1500	=B3+C3
4				

Рис. 5.5

Заметим, что для перехода из одного режима в другой можно использовать и комбинацию клавиш **Ctrl+'** (в русской раскладке — буква ё).



1. Что понимают под электронной таблицей?
2. Что такое табличный процессор?
3. Какие структурные элементы содержит электронная таблица?
4. Как задается адрес ячейки, адрес диапазона ячеек?

Упражнения

1. На рисунках изображены фрагменты электронных таблиц.
 - Назовите адреса активных ячеек.
 - Определите тип данных, которые хранятся в указанных учителем ячейках.
 - Определите адреса ячеек с указанными учителем данными.

а)

	A	B	C	D	E	F	G
1							
2		Пн	1	8	15	22	29
3		Вт	2	9	16	23	30
4		Ср	3	10	17	24	
5		Чт	4	11	18	25	
6		Пт	5	12	19	26	
7		Сб	6	13	20	27	
8		Вс	7	14	21	28	

б)

	I	J	K	L	M
1		время	температура	влажность	скорость ветра
2		6:00	4,2	78	6,6
3		9:00	8,5	76	5,2
4		12:00	14,3	64	2,4
5		15:00	15,6	57	3,6
6		18:00	13,7	61	4,3
7		21:00	9,4	63	7,1
8					

в)

	A	B	C
1	Площадь треугольника		
2	основание	высота	площадь
3	16,2	9,4	76,14
4	26,6	8,2	109,06
5			

г)

	A	B	C	D
1	товар	цена	количество	стоимость
2	тетрадь	1300	5	6500
3	ручка	1600	2	3200
4	карандаш	470	3	1410

д)

	A	B	C	D	E	F
1	число 1	число 2	сумма	разность	произведение	частное
2	15	6	21	9	90	2,5
3	24	8	32	16	192	3
4						

2. Откройте рабочую книгу ur10.xls. Определите указанные учителем элементы.

§ 11. ВВОД И РЕДАКТИРОВАНИЕ ДАННЫХ. СОХРАНЕНИЕ РАБОЧЕЙ КНИГИ

Рассмотрим на примерах ввод и редактирование данных разных типов.

Числовые данные

Вводимые числа могут быть целыми (например, 12), десятичными (45,95) или обыкновенными дробями (3 1/2), а также записываться в так называемом экспоненциальном представлении (1,3E+7).

Для ввода числа в ячейку ее выделяют с помощью мыши или клавиатуры. Набирают число, затем нажимают клавишу **Enter** (Ввод) или выделяют мышью другую ячейку. При вводе данные отображаются в активной ячейке и в строке формул. Табличный процессор распознает тип вводимых данных. По умолчанию числа выравниваются по правому краю ячейки.

Дробную часть числа от целой отделяют запятой или точкой, в зависимости от установок операционной системы. В русскоязычной версии Windows для отделения дробной части по умолчанию используется запятая.

Пример 1. Ввести 5 чисел: 7; 2,3; 12; 3 1/2; 1,3 · 10⁴ — в ячейки столбца В начиная со второй строки (ячейки В2) (рис. 5.6).

Правообладатель Издательский центр БГУ

B6		fx 13000		
	A	B	C	D
1				
2		7		
3		2,3		
4		12		
5		3 1/2		
6		1,30E+04		
7				

Рис. 5.6

На пересечении столбца В и второй строки выделим ячейку В2, подведя к ней указатель мыши и щелкнув левой кнопкой. Наберем на клавиатуре число 7 и нажмем клавишу **Enter**. Активной станет ячейка В3. Вместо нажатия **Enter** можно щелкнуть левой кнопкой мыши по следующей ячейке. Аналогично введем числа 2,3 и 12.

При этом табличный процессор распознает тип вводимых данных как числовой, на что указывает выравнивание по правому краю ячейки.

При вводе обыкновенной дроби в ячейке В5 после целой части 3 нажмем клавишу **Пробел** и наберем: 1/2. Если целая часть дроби отсутствует (например, 1/3), необходимо сначала набрать 0, затем пробел и далее 1/3. Выравнивание по правому краю ячейки указывает, что и в этом случае табличный процессор распознает тип вводимых данных как числовой.

При вводе числа в экспоненциальном представлении в ячейке В6 сначала наберем мантиссу 1,3, затем — латинскую букву Е (обозначающую основание степени 10) и после нее — порядок 4. После нажатия клавиши **Enter** в этой ячейке будет отображаться 1,30E+04, а в строке формул — 13000 (см. рис. 5.6).

Текстовые данные

Для ввода текстовых данных выделяют нужную ячейку, набирают текст и нажимают клавишу **Enter**. Excel автоматически распознает текстовые значения и по умолчанию выравнивает их по левому краю ячейки.

Если соседние ячейки не заполнены, то Excel позволяет отображать более длинный текст, перекрывающий расположенные справа ячейки. Если в них уже размещены данные, видимое текстовое значение обрезается.

Если требуется, чтобы Excel сохранял в виде текста числа, даты и формулы, вводить их начинают с апострофа. Например, если ввести в ячейку '55, там появится число 55, выровненное по левому краю без апострофа.

Пример 2. Создать таблицу «Численность населения городов» (рис. 5.7).

Выделим ячейку A1 и наберем текст: «Численность населения». Набранное отобразится в активной ячейке и в строке формул (рис. 5.8, а). Набрав всю строку, нажмем клавишу **Enter**. Активной станет ячейка A2 (рис. 5.8, б). Введенный текст выравнивается по левому краю. Это значит, что табличный процессор распознает тип вводимых данных как текст.

Аналогичным способом введем остальные слова в ячейки столбца А. Щелчком мыши выделим ячейку B2 и введем текст: «тыс.чел». В ячейки столбца В последовательно будем вводить численность населения городов. Табличный процессор распознает тип вводимых данных как числовой, на что указывает выравнивание по правому краю.

	D11		fx
	А	В	С
1	Численность населения		
2	Город	тыс.чел	
3	Брест	318	
4	Витебск	356	
5	Гомель	499	
6	Гродно	338	
7	Минск	1834	
8	Могилев	372	

Рис. 5.7

	A1		fx
	А	В	С
1	Численность		
2			

Рис. 5.8, а

	A2		fx
	А	В	С
1	Численность населения		
2			
3			

Рис. 5.8, б

Редактировать данные можно как в строке формул, так и в самой ячейке. Для замены содержимого ячейку выделяют и вводят новые данные. Двойной щелчок мыши по ячейке позволяет редактировать данные. Приемы редактирования не отличаются от применяемых в текстовом редакторе.

Примечания



Для привлечения внимания к наиболее важной информации или ввода пояснений можно снабдить ячейки таблицы примечаниями. Для этого нужно выбрать в меню **Вставка** пункт **Примечание**.

Правообладатель Издательский центр БГУ

Откроется всплывающее окно с именем пользователя и мигающим курсором, куда можно ввести текст примечания.

Чтобы изменить имя, которое появляется во всплывающем окне, выберем в меню **Сервис** команду **Параметры**, перейдем на вкладку **Общие** и введем новое имя в текстовом поле **Имя пользователя**.

Чтобы просмотреть все примечания в рабочей книге, достаточно выбрать в меню **Вид** команду **Примечания**. Несколько полезных кнопок для работы с примечаниями содержит также панель инструментов **Рецензирование**, которая вызывается из меню **Вид**.

Ячейки, снабженные примечаниями, имеют маленький красный треугольник в правом верхнем углу. Чтобы просмотреть примечание во время работы с листом, достаточно навести указатель мыши на ячейку с примечанием и задержаться до появления всплывающего окна с текстом. Чтобы удалить примечание, можно выделить ячейку, выбрать в меню **Правка** команду **Очистить**, а затем в подменю — команду **Примечания**, либо воспользоваться командами контекстного меню.



Пример 3. Создать примечания к ячейкам таблицы «Численность населения городов», указав год получения данных (рис. 5.9).

Выделим ячейку В3 (в которой содержатся данные о населении Бреста), вызовем всплывающее окно примечаний, в котором введем месяц и год получения данных: март 2010 (рис. 5.9, а). Откорректируем размеры окна перемещением маркеров его границ с помощью мыши.

	А	В	С	Д
1	Численность населения			
2	Город	тыс. чел		
3	Брест	318		
4	Витебск	356		
5	Гомель	499		
6	Гродно	338		
7	Минск	1834		
8	Могилев	372		
9				

Рис. 5.9, а

	А	В	С	Д
1	Численность населения			
2	Город	тыс. чел		
3	Брест	318		
4	Витебск	356		
5	Гомель	499		
6	Гродно	338		
7	Минск	1834		
8	Могилев	372		
9				

Рис. 5.9, б

Аналогичным способом создадим примечания к остальным ячейкам. Они будут помечены красными треугольниками. Наведем указатель мыши, например, на ячейку В5. Вид появившегося примечания показан на рис. 5.9, б.

Ввод формул

Табличный процессор Excel выполняет все вычисления по формулам. В состав формул могут входить числа, знаки математических операций, скобки, адреса ячеек и диапазонов, а также стандартные встроенные выражения, называемые *функциями*.

Ввод формулы в Excel начинают со знака равенства. Знак равенства является признаком начала формулы. Он указывает табличному процессору на необходимость интерпретации следующего за ним выражения в виде формулы. Ввод формулы завершают нажатием клавиши **Enter** либо щелчком мыши по кнопке  в строке формул или по любой свободной ячейке. После ввода формулы табличный процессор выполняет вычисления и по умолчанию отображает в ячейке результат вычислений.

Для обозначения арифметических операций используются следующие знаки: +, -, *, /; для возведения в степень — знак ^ («шапка»). Порядок выполнения действий не отличается от принятого в математике: сначала вычисляются значения функций и выполняются действия в скобках, затем — возведение в степень, умножение и деление, и наконец — сложение и вычитание.

Формулы содержат *ссылки* на ячейки или диапазоны, например: $= (A4+B8) * C6$, $= F7 * 3 + B5$, $= СУММ(A1:A5)$.

При создании формулы входящие в нее ссылки могут вводиться путем набора адресов на клавиатуре. Однако лучше их вводить выделением нужных ячеек с помощью мыши или клавиш перемещения курсора (стрелок). В этом случае можно не бояться спутать похожие по начертанию русские и латинские буквы.

Пример 4. В ячейки А2, А3 и А4 ввести числа 2, 6, 7, а в ячейки В2, В3 и В4 — числа 5, 3, 5. Выполнить вычисления: в ячейке С2 найти сумму по формуле $=A2+B2$; в ячейке С3 — произведение по формуле $=A3*B3/9$; в ячейке С4 — разность квадратов по формуле $=A4^2-B4^2$.

Введем в ячейки заданные числа.

Выделим ячейку С2. Наберем знак = и щелкнем мышью по ячейке А2. Ее адрес будет автоматически записан в ячейке С2 после знака =. Наберем знак + и щелкнем мышью по ячейке В2. Ее адрес будет записан в ячейке С2 после знака + (рис. 5.10, а). При этом рамки ячеек и их адреса в строке формул выделяются одинаковым цветом: А2 — синим, В2 — зеленым. Это облегчает проверку правильности формул и их редактирование.

После нажатия клавиши **Enter** будут произведены вычисления, и в ячейке С2 отобразится результат (число 7). Аналогичным способом введем заданные формулы в ячейки С3 и С4. Показатель степени (число 2) в Excel вводится с клавиатуры после нажатия символа ^ (рис. 5.10, б).

	А	В	С	Д
1			формула	
2	2	5	=A2+B2	
3	6	3		
4	7	5		

Рис. 5.10, а

	А	В	С	Д
1			формула	
2	2	5	7	
3	6	3		
4	7	5	=A4^2-B4^2	

Рис. 5.10, б

Ввод последовательностей



Нередко диапазоны ячеек требуется заполнять одинаковыми или изменяющимися по определенному закону данными, например, последовательностями натуральных или четных чисел, названиями месяцев или дней недели. Табличный процессор Excel имеет инструменты, которые упрощают задачу ввода таких данных, позволяя заполнить диапазон ячеек повторяющимся значением или последовательностями значений, называемыми *рядами*. Эта возможность экономит время при вводе текста, чисел или дат. Для ввода ряда значений в диапазон ячеек можно воспользоваться командой **Заполнить** из меню **Правка** или специальным приемом работы с мышью, который называется *автозаполнением*.

Автозаполнение выполняется с помощью перетаскивания маленького черного квадрата, называемого *маркером заполнения*. Он

находится в правом нижнем углу активной ячейки или выделенного диапазона. При наведении указателя мыши на маркер заполнения вид указателя меняется на знак +, означающий, что включается инструмент автозаполнения. Для создания последовательности текстовых значений, чисел или дат достаточно выделить ячейку или диапазон, перетащить указатель по заполняемым ячейкам и потом отпустить кнопку мыши.



Пример 5. Заполнить диапазон A1:A12 последовательностью нечетных чисел, столбец B — названиями месяцев, столбец C — названиями дней недели.

Введем в ячейку A1 первое число ряда (1), а в ячейку A2 — следующее (3). Выделим *две* эти ячейки. Установим указатель мыши на маркер заполнения и, удерживая левую кнопку, протащим его вниз до ячейки A12. Диапазон A1:A12 заполнится последовательностью нечетных чисел от 1 до 23 (рис. 5.11).

Если выделить только одну ячейку и протащить указатель мыши, то содержащееся в этой ячейке число будет скопировано во все ячейки диапазона. Для автозаполнения ячеек последовательностью чисел необходимо выделить две ячейки с числами из этой последовательности.

Введем в ячейку B1 название месяца. Выделим эту ячейку и протащим маркер заполнения вниз до B12. Диапазон B1:B12 автоматически заполнится названиями месяцев (рис. 5.12).

Аналогичным способом заполняются ячейки с названиями дней недели (включая и сокращенные названия), а также с датами (см. столбец D на рис. 5.12).

Еще раз подчеркнем, что при перетаскивании маркера заполнения отмеченные ячейки заполняются данными, характер изменения которых зависит от последовательности значений в выде-

	A
1	1
2	3
3	5
4	7
5	9
6	11
7	13
8	15
9	17
10	19
11	21
12	23

Рис. 5.11

	A	B	C	D
1	1	январь	Пн	
2	3	февраль	Вт	
3	5	март	Ср	5 мая
4	7	апрель	Чт	6 мая
5	9	май	Пт	7 мая
6	11	июнь	Сб	8 мая
7	13	июль	Вс	9 мая
8	15	август		10 мая
9	17	сентябрь		11 мая
10	19	октябрь		12 мая
11	21	ноябрь		
12	23	декабрь		

Рис. 5.12

ленном диапазоне ячеек. Если инструмент автозаполнения не распознает закономерности в значениях выделенных ячеек, то данные в заполняемых ячейках дублируются (повторяются).

Отметим, что простое копирование (дублирование) содержимого одной ячейки в диапазоны ячеек (в том числе и несмежные) можно выполнить, используя буфер обмена. Для этого выделяют исходную ячейку и командой **Правка** → **Копировать** (или комбинацией клавиш **Ctrl+C**) помещают ее содержимое в буфер обмена. После этого выделяют требуемые диапазоны (несмежные — с нажатой клавишей **Ctrl**) и выполняют команду **Вставить** (или нажимают комбинацию клавиш **Ctrl+V**).

Сохранение рабочей книги

Для сохранения рабочей книги следует выполнить команду **Сохранить** из меню **Файл** или щелкнуть по кнопке . При первом сохранении появляется диалоговое окно **Сохранение документа**, в котором выбирают папку и вводят имя файла. При повторном сохранении это окно не показывается, книга автоматически сохраняется в том же файле. Чтобы сохранить книгу под другим именем или в другой папке, следует в меню **Файл** выбрать команду **Сохранить как**, после чего в окне **Сохранение документа** набрать новое имя.

Для закрытия рабочей книги выполняют команду **Закрыть** из меню **Файл** или щелкают по кнопке  окна этой книги.

Для завершения работы с Excel нужно закрыть окно программы (щелкнуть по кнопке  окна программы или нажать комбинацию клавиш **Alt+F4**).



1. Данные каких типов могут храниться в ячейках MS Excel?

2. Как в ячейку электронной таблицы вводят формулу?



3. В чем заключается автозаполнение ячеек?

Упражнения

1. Создайте таблицу в соответствии с предложенным учителем образцом:

Правообладатель Издательский центр БГУ

а)	A	B
1	город	основан
2	Минск	1067
3	Брест	1019
4	Витебск	974
5	Гомель	1142
6	Гродно	1127
7	Могилев	1267

б)	A	B
1	наименование	цена
2	системный блок	430
3	монитор	210
4	клавиатура	11
5	мышь	5
6	коврик	1
7		

в)	A	B
1	Вещество	плотность
2	воздух	1,293
3	вода	1000
4	дерево	700
5	сталь	7800
6	свинец	11300
7		

г)	A	B	C	D
1	товар	тетрадь	ручка	карандаш
2	цена	1200	1500	470
3				

д)	A	B	C	D	E
1	Плотность водного раствора серной кислоты				
2	w, %	10	20	40	80
3	p, г/мл	1,07	1,14	1,30	1,73

е)	A	B
1	Самые большие озера	
2	Озеро	площадь, кв.км
3	Нарочь	79,6
4	Освейское	52,8
5	Червоное	40,8
6	Лукомское	37,7
7	Дривяты	36,1

ж)	A	B
1	Самые глубокие озера	
2	Озеро	глубина, м
3	Нарочь	24,8
4	Мядель	24,6
5	Снуды	16,5
6	Дривяты	12
7	Лукомское	11,5

2. Откройте рабочую книгу urp11.xls. Введите формулы для расчета неизвестных величин (вариант по указанию учителя):

	A	B	C	D	E	F	G	H
1	квадрат				прямоугольник			
2	сторона	периметр	площадь		сторона	сторона	периметр	площадь
3		3,6			7,4	5,2		
4								
5	треугольник				параллелограмм			
6	основание	высота	площадь		сторона	сторона	периметр	
7		5,6	4,2		8,2	3,5		
8								
9	окружность				круг			
10	радиус	длина			радиус	площадь		
11		4,7			6,8			

3. В ячейки A1 и B1 введите два числа (вариант по указанию учителя). Введите формулы для вычисления:

- а) в ячейке A3 — суммы, в ячейке B3 — разности этих чисел;
 б) в ячейке A3 — произведения, в ячейке B3 — частного этих чисел;

Правообладатель Издательский центр БГУ

в) в ячейке А3 — суммы квадратов, в ячейке В3 — разности квадратов этих чисел;

г) в ячейке А3 — суммы кубов, в ячейке В3 — разности кубов этих чисел.



4. Создайте таблицу, в которую занесите данные о трех своих одноклассниках: фамилия, имя, дата рождения, улица. В примечания поместите их любимые занятия.

	А	В	С	Д
1	Фамилия	Имя	Дата рождения	Улица
2	Александров	Павел	12.03.1995	Есенина
3	Иванов	Максим	08.05.1995	Монавтов
4	Петров	Андрей	компьютер, музыка	Имова
5				



5. Заполните диапазоны ячеек данными в соответствии с предложенными учителем образцами.

а)

	А	В	С
1			
2		t, с	x, м
3		1	1
4		2	4
5		3	7
6		4	10
7		5	13
8		6	16
9		7	19
10		8	22
11		9	25

б)

	А	В	С
1			
2		t, с	x, м
3		2	4
4		4	8
5		6	12
6		8	16
7		10	20
8		12	24
9		14	28
10		16	32
11		18	36

в)

	А	В	С
1			
2		x	y
3		1	2
4		2	4
5		3	8
6		4	16
7		5	32
8		6	64
9		7	128
10		8	256
11		9	512

г)

	А	В	С	Д	Е	Ф	Г
1							
2		Пн	1	8	15	22	29
3		Вт	2	9	16	23	30
4		Ср	3	10	17	24	31
5		Чт	4	11	18	25	
6		Пт	5	12	19	26	
7		Сб	6	13	20	27	
8		Вс	7	14	21	28	

§ 12. РАБОТА С ДИАПАЗОНАМИ ЯЧЕЕК

Вы уже умеете выделять отдельные ячейки и редактировать данные, которые в них содержатся. Рассмотрим теперь следующие операции:

- выделение диапазонов ячеек;
- очистка ячеек и диапазонов;
- копирование и перемещение содержимого ячеек и диапазонов;
- вставка и удаление ячеек, строк, столбцов.

Выделение диапазонов ячеек

Выделение ячейки делает ее активной, ее имя (адрес) появляется в поле имен (см. рис. 5.2, *a*).

Напомним, что для обозначения диапазонов ячеек используется особая форма записи: например, A1:E1 соответствует строке из пяти ячеек, а E5:E8 — столбцу из четырех ячеек.

Выделить диапазон ячеек можно с помощью мыши или клавиатуры.

Для выделения диапазона ячеек выделяют сначала одну из его крайних угловых ячеек. С этой целью на ней устанавливают указатель мыши и щелкают левой кнопкой. Удерживая кнопку нажатой, перетаскивают указатель по остальным ячейкам диапазона и отпускают кнопку.

Несмежные (т. е. несоприкасающиеся) диапазоны выделяют с нажатой клавишей **Ctrl**. Ее отпускают после завершения выделения всех диапазонов.

Для выделения диапазона ячеек с помощью клавиатуры необходимо перейти на крайнюю ячейку диапазона с помощью клавиш перемещения курсора. Затем, удерживая нажатой клавишу **Shift**, клавишами перемещения выделить оставшиеся ячейки и отпустить **Shift**.

Очистка ячеек

Для удаления содержимого ячеек и диапазонов просто выделяют их и нажимают клавишу **Del**. В меню **Правка** выбором пункта **Очистить** можно вызвать подменю с командами, позволяющими удалить только содержимое ячеек, форматы, примечания или все сразу (рис. 5.13).

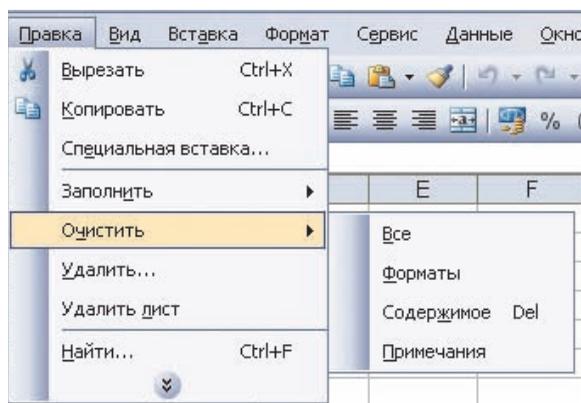


Рис. 5.13

Копирование и перемещение содержимого ячеек

Для копирования (перемещения) данных из одного места на листе в другое можно воспользоваться командами **Копировать (Вырезать)** → **Вставить** из меню **Правка**, кнопками () → на панели инструментов или комбинацией клавиш **Ctrl+C (Ctrl+X)** → **Ctrl+V**. При выполнении команд **Копировать (Вырезать)** выделенный диапазон ячеек обрамляется пунктирной линией, а его содержимое (включая примечания и форматирование) помещается в буфер обмена. При выполнении команды **Вставить** это содержимое из буфера обмена помещается на новое место, заменяя имеющееся там содержимое. Прекратить выполнение операций можно нажатием клавиши **Esc**.

Заметим, что команда **Вырезать** неприменима для перемещения несмежных диапазонов ячеек. Кроме того, в отличие от других приложений Windows, после вырезания содержимое буфера можно вставить только один раз. Для выполнения многократных вставок следует пользоваться командой **Копировать** (или комбинацией клавиш **Ctrl+C**).

Самый быстрый и наглядный способ перемещения диапазона ячеек — *перетаскивание (буксировка)* мышью из одного места листа в другое. Для выполнения этой операции сначала необходимо выделить требуемый диапазон ячеек (обычно с помощью мыши). Затем отпустить левую кнопку мыши и переместить указатель ячейки (белый крестик) на границу выделения так, чтобы он принял вид черного крестика со

стрелкой вниз. Далее надо нажать левую кнопку мыши и перетащить выделенные ячейки в нужное место. Для облегчения выбора места во время перемещения отображается выделенный диапазон, контур перетаскиваемого и его новый адрес.

Для выполнения копирования ячеек таким способом во время перетаскивания необходимо удерживать нажатой клавишу **Ctrl**.

Вставка и удаление строк, столбцов и ячеек

Новые строки и столбцы добавляются в лист командами **Строки** и **Столбцы** из меню **Вставка**.

Вставка новой строки сдвигает имеющиеся строки вниз. Вставка столбца сдвигает имеющиеся столбцы вправо. При добавлении строки выделяют строку, перед которой требуется вставить новую строку. Строка выделяется щелчком по ее номеру. При добавлении нового столбца выделяют столбец, слева от которого требуется вставить новый столбец. Столбец выделяют щелчком по его буквенному обозначению.

Вставку отдельных ячеек в строки или столбцы листа выполняют командами **Вставка** → **Ячейки**. При этом в появившемся диалоговом окне указывают, куда следует сдвигать ячейки при вставке — вниз или вправо.

Удаление ячеек, строк или столбцов выполняют командами **Правка** → **Удаление**. В появившемся диалоговом окне выбирают, что требуется удалить и куда сдвигать ячейки при удалении (рис. 5.14).

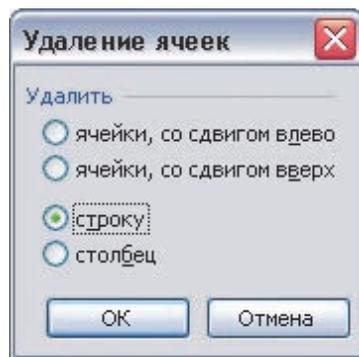


Рис. 5.14

Отмена и возврат изменений

В любой момент у пользователя остается возможность отменить последние сделанные изменения. С этой целью можно использовать команду **Отменить** из меню **Правка**, или кнопку  на панели инструментов, или комбинацию клавиш **Ctrl+Z**. Команда **Повторить** (или комбинация клавиш **Ctrl+Y**) позволяет вернуть отмененную команду.

Пример 1. Ввести данные в ячейки диапазона A2:В12 (рис. 5.15, а) и выполнить требуемые операции с диапазонами ячеек.

Введем данные в ячейки. Затем выделим диапазон B5:В11 (7 ячеек с B5 по B11). Для этого установим указатель мыши в виде крестика \oplus на ячейку B5, нажмем левую кнопку мыши и, не отпуская ее, протащим выделение (более темного цвета) до ячейки B11 включительно (рис. 5.15, б).

Подведем указатель мыши к одной из границ выделения (при этом белый крестик превратится в черный крестик со стрелками \oplus (см. рис. 5.15, б)) и, удерживая левую кнопку, перетащим выделенные данные в столбец D (рис. 5.15, в).

	A	B
1		
2	16	4,2
3	23	4,8
4	12	3,3
5	15	3,9
6	10	3,5
7	11	3,4
8	14	3,5
9	9	3,2
10	19	5
11	12	3,2
12	18	4,5

Рис. 5.15, а

	A	B
1		
2	16	4,2
3	23	4,8
4	12	3,3
5	15	3,9
6	10	3,5
7	11	3,4
8	14	3,5
9	9	3,2
10	19	5
11	12	3,2
12	18	4,5

Рис. 5.15, б

	A	B	C	D
1				
2	16	4,2		
3	23	4,8		
4	12	3,3		
5	15			3,9
6	10			3,5
7	11			3,4
8	14			3,5
9	9			3,2
10	19			5
11	12			3,2
12	18	4,5		

Рис. 5.15, в

Аналогичным способом перетащим ячейки на прежнее место.

Перетаскивание ячеек при нажатой клавише **Ctrl** приводит к копированию данных. При этом справа от светлой стрелки появляется знак «плюс»: \oplus . Выделим 10 ячеек A6:В10 и скопируем их содержимое в диапазон C4:D8 (рис. 5.16, а).

Напомним, что копировать и перемещать данные можно также с помощью команд меню **Правка**, или нажатия правой кнопки мыши и вызова контекстного меню, или с помощью комбинаций клавиш **Ctrl+X** (вырезать), **Ctrl+C** (копировать), **Ctrl+V** (вставить).

Выделим два столбца A и B и скопируем их содержимое в Лист2, столбцы D и F (лист переключим щелчком мыши по ярлыку в нижней части окна).

	A	B	C	D
1				
2	16	4,2		
3	23	4,8		
4	12	3,3	10	3,5
5	15	3,9	11	3,4
6	10	3,5	14	3,5
7	11	3,4	9	3,2
8	14	3,5	19	5
9	9	3,2		
10	19	5		
11	12	3,2		
12	18	4,5		

Рис. 5.16, а

	A	B	C	D	E
1					
2	16	4,2			
3	23	4,8			
4	12	3,3			
5	15	3,9			
6	10	3,5		12	3,3
7	11	3,4		15	3,9
8	14	3,5		10	3,5
9	9	3,2		19	5
10	19	5		12	3,2
11	12	3,2			
12	18	4,5			

Рис. 5.16, б

Выделим и очистим все ячейки **Листа2** (с помощью меню **Правка** → **Очистить все**). Вернемся на **Лист1**.



Выделение несмежных диапазонов ячеек осуществляют при нажатой клавише **Ctrl**. Выделим диапазон ячеек A4 : B6, нажмем клавишу **Ctrl** и, удерживая ее, выделим диапазон ячеек A10 : B11. Скопируем содержимое в буфер обмена и вставим в столбцы D и E (рис. 5.16, б).



1. Какие действия можно выполнять с выделенными диапазонами?
2. Каким образом можно копировать (перемещать) содержимое ячеек?
3. Каким образом можно удалить строку, столбец, ячейки электронной таблицы? Вставить строку, столбец, ячейки?

Упражнение

Откройте книгу upr12.xls (вариант по указанию учителя). В таблицах перепутаны некоторые строки или столбцы. С помощью операций копирования, перемещения, удаления и вставки исправьте таблицу.

а)

	A	B	C	D	E
1	Численность населения			Численность населения	
2	Город	тыс.чел		Город	тыс.чел
3	Брест	318		Брест	318
4	Витебск	356		356	Витебск
5	Гомель	499		499	Гомель
6	Гродно	338		Гродно	338
7	Минск	1834		1834	Минск
8	Могилев	372		372	Могилев

б)

	A	B	C	D	E
1	Пн	1		Пн	1
2	Вт	2		4 Чт	
3	Ср	3		5 Пт	
4	Чт	4		2	6
5	Пт	5		3	7
6	Сб	6		Вт	Сб
7	Вс	7		Ср	Вс

в)

	A	B	C	D	E	F	G
1	Пн	1	8		1 Пн		8
2	Вт	2	9		2 Вт		9
3	Ср	3	10		2 Вт		9
4	Чт	4	11		4 Чт		11
5	Пт	5	12		5 Пт		12
6	Сб	6	13		6 Сб		13
7	Вс	7	14		7 Вс		14
8					3 Ср		10
9					3 Ср		10
10					4 Чт		11
11					4 Чт		11
12					5 Пт		12

г)

	A	B	C	D	E
1	месяц	дней		месяц	дней
2	Январь	31		Январь	30
3	Февраль	28		Февраль	31
4	Март	31		31	Июль
5	Апрель	30		28	Август
6	Май	31		31	Сентябрь
7	Июнь	30		30	Октябрь
8	Июль	31		Март	31
9	Август	31		Апрель	31
10	Сентябрь	30		Май	30
11	Октябрь	31		Июнь	31
12	Ноябрь	30		Ноябрь	30
13	Декабрь	31		Декабрь	31

§ 13. ФОРМАТИРОВАНИЕ ЭЛЕКТРОННОЙ ТАБЛИЦЫ

К основным операциям форматирования относятся: форматирование данных (содержимого ячеек), форматирование символов (шрифта), форматирование вида ячеек (цвета и границ ячеек), изменение ширины столбцов и высоты строк.

Правообладатель Издательский центр БГУ

Форматирование данных

Формат отображения данных позволяет представлять данные в наиболее подходящем для пользователя виде. При вводе любых данных по умолчанию используется формат **Общий**. Тип данных при этом определяется автоматически.

Для форматирования содержащихся в ячейках данных эти ячейки выделяют и командой меню **Формат** → **Ячейки** или правой кнопкой мыши вызывают диалоговое окно **Формат ячеек** (рис. 5.17). Формат задается на вкладке **Число**. При этом сами данные не изменяются, в чем можно убедиться, посмотрев на строку формул.

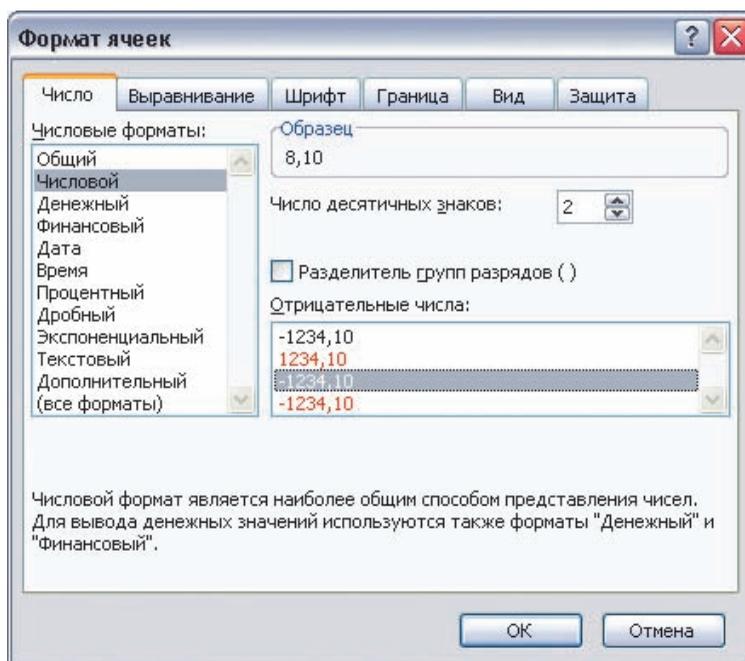


Рис. 5.17

Выбрав **Числовой** тип данных, можно установить количество десятичных знаков. Если введенное число имеет меньшее количество цифр после запятой, то табличный процессор добавляет нули, если большее — округляет число. Оперативно изменять количество десятичных знаков можно с помощью кнопок панели инструментов: увеличить 

или уменьшить  разрядность. Большие числа удобно отображать, разделив группы разрядов с помощью кнопки **Формат с разделителями** .

При выборе формата **Денежный** отображается число и название денежной единицы. В формате **Процентный** число в ячейке умножается на 100 и отображается со знаком «%». Эти форматы также можно задавать с помощью кнопок  и  соответственно. Формат **Дробный** отображает число в виде простой дроби, а формат **Экспоненциальный** — в экспоненциальном представлении. Форматы **Дата** и **Время** представляют введенные числа в виде дат или времени.

В диалоговом окне **Формат ячеек** можно выбрать вариант отображения каждого формата.

Пример 1. Открыть книгу prim13-1.xls. Ввести столбец чисел (рис. 5.18, а). Проследить изменение их отображения при изменении формата ячеек.

	А	В
1	Формат числа	
2	Общий	23,6
3	Числовой, 2 десятич. знака	23,60
4	Денежный	23,6р.
5	Дата	23 янв
6	Время	14:24
7	Процентный	2360,00%
8	Дробный	23 3/5
9	Экспоненциальный	2,36E+01
10	Текстовый	23,6

Рис. 5.18, а

	А	В
1	Формат числа	
2	Общий	23,6
3	Числовой, 2 десятич. знака	23,60
4	Денежный	23,60р.
5	Дата	23 янв
6	Время	14:24
7	Процентный	2360,00%
8	Дробный	23 3/5
9	Экспоненциальный	2,36E+01
10	Текстовый	23,6

Рис. 5.18, б

В ячейку В2 введем число, например 23,6. Скопируем его на диапазон ячеек В2:В10. Будем поочередно выделять эти ячейки и применять форматы, указанные в столбце А. Сравним вид чисел с показанным на рис. 5.18, б.

Пример 2. Открыть книгу prim13-2.xls. Установить форматы отображения данных, соответствующие содержанию ячеек приведенной таблицы (рис. 5.19).

В первой строке таблицы, а также в столбце В размещены текстовые данные, в строках 1—3 столбцов А, Е, F — числовые. При их вводе

	A	B	C	D	E	F	G
1		товар	цена	поступление	количество	продано	остаток
2	1	картофель	2 800р.	07.03.2011	400	340	15%
3	2	капуста	4 500р.	16.03.2011	240	180	25%
4	3	яблоки	3 900р.	17.02.2011	120	80	33%

Рис. 5.19

тип устанавливается автоматически. В столбце D установим формат **Дата**. Для чисел столбца C установим формат **Денежный** (число десятичных знаков равно нулю), а для чисел столбца G — **Процентный**.

Изменение внешнего вида ячеек

Операции изменения *шрифта, цвета, размера и начертания* символов в ячейках электронной таблицы аналогичны соответствующим операциям форматирования символов в текстовом редакторе MS Word. Необходимо выделить эти ячейки и установить требуемые параметры с помощью кнопок на панели инструментов (см. рис. 5.1). Эти параметры можно установить также на вкладке **Шрифт** диалогового окна **Формат ячеек**, которое вызывается из меню **Формат** → **Ячейки** или из контекстного меню, которое вызывается нажатием правой кнопки мыши (рис. 5.20).

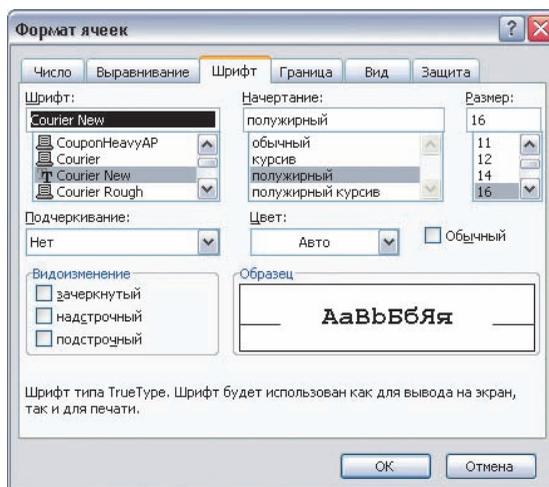


Рис. 5.20

Напомним, что при вводе в ячейки таблицы текст автоматически выравнивается по левой границе ячейки, а числа — по правой. Однако способ выравнивания можно изменить.

Для выравнивания по горизонтали проще всего воспользоваться кнопками на панели инструментов:  — по левому краю,  — по правому,  — по центру ячейки. Более сложные способы задают на вкладке **Выравнивание** диалогового окна **Формат ячеек** (рис. 5.21). Содержимое ячеек можно выровнять не только по горизонтали, но и по вертикали и даже изменить его ориентацию в ячейке, повернув на заданное количество градусов. Чтобы расположить текст в ячейке в несколько строк, следует установить флажок **переносить по словам**.

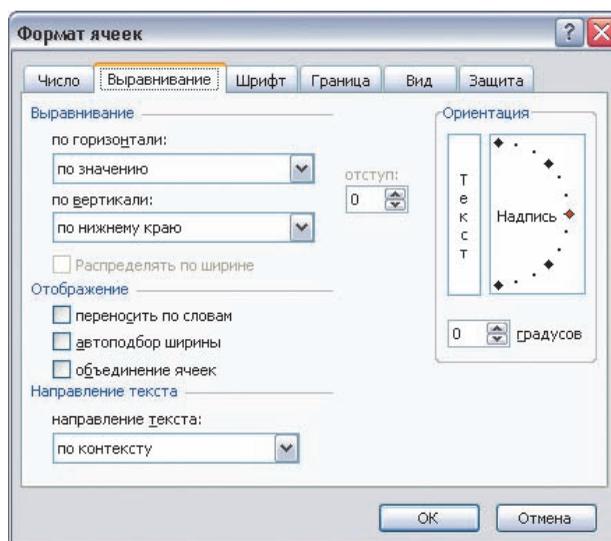


Рис. 5.21

Нередко требуется объединить несколько ячеек — например, отобразить заголовок в нескольких столбцах или строках. Для этого предусмотрена операция **Объединение ячеек**. Объединяемые ячейки выделяют и с помощью меню или кнопки на панели инструментов (рис. 5.22) выполняют объединение.

Повышению наглядности таблицы способствует заливка ячеек фоновым цветом, а также обрамление их границами.

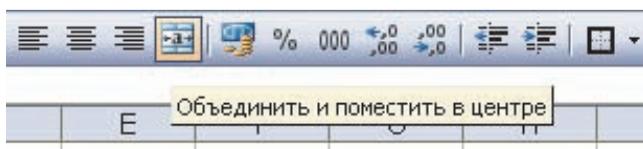


Рис. 5.22

Чтобы задать границы ячеек, проще всего использовать кнопку  на панели инструментов (рис. 5.23, а), а цвет заливки можно выбрать из палитры, которая появляется при нажатии кнопки  (рис. 5.23, б). Более сложные установки выполняются в диалоговом окне **Формат ячеек**. На вкладке **Граница** можно задать внешний вид линии, цвет и расположение границ (рис. 5.24), на вкладке **Вид** — выбрать цвет и узор фона.

Чтобы изменить ширину столбцов и высоту строк, достаточно перетащить с нажатой левой

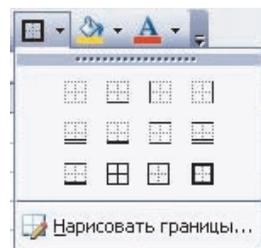


Рис. 5.23, а

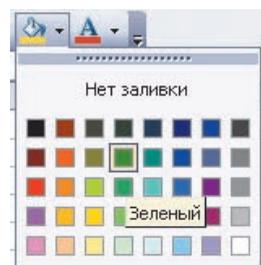


Рис. 5.23, б

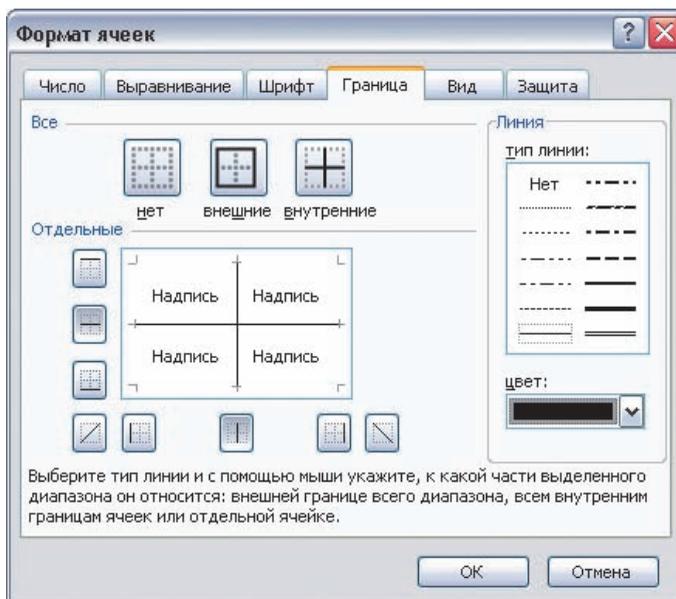


Рис. 5.24

кнопкой мыши границы их заголовков. Для точного задания используют меню **Формат** → **Строка** → **Высота** или **Формат** → **Столбец** → **Ширина**.

Пример 3. Создать таблицу «Экспертные оценки телевизоров» и оформить ее в соответствии с рис. 5.25.

	A	B	C	D	E	F
1	Экспертные оценки					
2		параметр	четкость текста	красочность	дизайн	удобство настройки
3	модель	Астра	7	6	8	9
4		Радуга	6	8	7	8

Рис. 5.25

Сначала введем данные. В ячейку B1 поместим заголовок «Экспертные оценки». Поскольку ячейка C1 свободна, он будет отображен полностью. В ячейки B2:F2 введем подзаголовки. В ячейках C2 и D2 они отображаются частично, поскольку соседние ячейки справа заняты. Аналогично заполним остальные ячейки. Вид исходной таблицы показан на рис. 5.26.

	A	B	C	D	E	F	G
1		Экспертные оценки					
2		параметр	четкость т	красочнос	дизайн	удобство настройки	
3	модель	Астра	7	6	8	9	
4		Радуга	6	8	7	8	

Рис. 5.26

Оформление таблицы начнем с форматирования текста. Установим полужирное начертание в ячейках A3, B1 и B3:B4. Выровняем содержимое ячеек C2:F4 по центру, а B2 — по правому краю.

Объединим ячейки A1:F1. С этой целью выделим их и нажмем кнопку **Объединить и поместить в центре**. Зальем объединенные ячейки бледно-зеленым цветом, нажав кнопку **Цвет заливки** и выбрав требуемый цвет. Аналогично объединим ячейки A3:A4 и зальем светло-голубым цветом.

Выделим ячейки В2:F2. С помощью меню **Формат** → **Ячейки** на вкладке **Выравнивание** установим флажок **переносить по словам**, **Выравнивание по вертикали — по центру**.

Выделим ячейки А3:A4. С помощью меню **Формат** → **Ячейки** на вкладке **Выравнивание** установим параметр **Ориентация 90** градусов.

Таблица приобрела вид, показанный на рис. 5.27.

	A	B	C	D	E	F	G
1	Экспертные оценки						
2		параметр	четкость текста	красоч ность	дизайн	удобство настройки	
3	mode	Астра	7	6	8	9	
4		Радуга	6	8	7	8	

Рис. 5.27

Откорректируем ширину столбца F. Для этого подведем указатель мыши к границе столбцов F и G в заголовке листа. Он примет вид крестика (см. рис. 5.27). Увеличим ширину столбца, протаскив границу вправо с нажатой левой кнопкой мыши. Аналогичным способом откорректируем ширину столбца D, протаскив границу столбцов D и E. Высоту строк 3 и 4 увеличим перемещением их границ вниз.

Осталось задать границы ячеек в соответствии с рис. 5.25. Для этого будем выделять необходимые диапазоны ячеек и с помощью кнопок на панели инструментов выбирать требуемый вид границ.



В MS Excel предусмотрена возможность автоматического форматирования таблиц. При этом выделенному диапазону ячеек придается заранее установленный стандартный вид. Подходящий вариант можно выбрать в диалоговом окне **Автоформат** (рис. 5.28), которое вызывается из меню **Формат**.



Рис. 5.28

Подготовка таблицы к печати

Перед выводом документа на печать рекомендуется предварительно просмотреть его. Если границы ячеек не установлены, то они на печать не выводятся (хотя отображаются на рабочем листе). Отсутствие границ у ячеек можно обнаружить, выполнив предварительный просмотр.

С помощью меню **Файл** → **Предварительный просмотр** или кнопки  на панели инструментов можно вызвать диалоговые окна и инструменты, которые позволяют оформить окончательный вид таблицы, а также произвести необходимую коррекцию в установленные по умолчанию параметры печати. Так, на вкладке **Страница** окна **Параметры страницы** можно выбрать книжную или альбомную ориентацию, установить масштаб, размер бумаги и другие параметры (рис. 5.29).

В окне **Печать** выбирают принтер, указывают число копий, номера выводимых на печать страниц или выделенный диапазон ячеек (рис. 5.30).

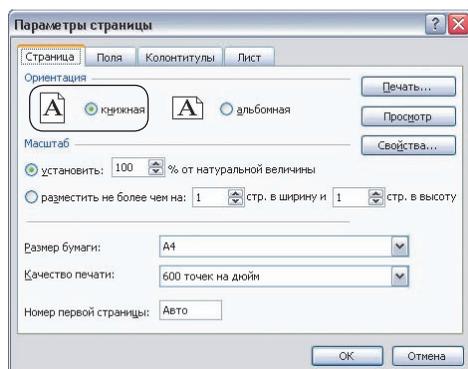


Рис. 5.29

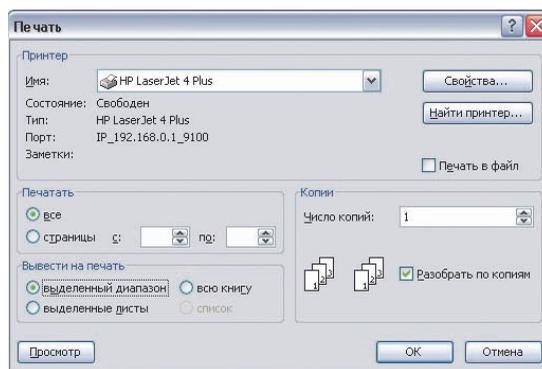


Рис. 5.30



1. Каковы основные операции форматирования электронной таблицы?
2. Каковы основные форматы данных в ячейках электронной таблицы?
3. Какими способами можно выравнивать содержимое ячеек?

Упражнения

1. Откройте рабочую книгу upr13-1.xls. Отформатируйте данные указанной учителем таблицы в соответствии с рисунком.

а)

	A	B	C	D
1	спрос/предложение			
2	1	январь	90/120	75%
3	2	февраль	88/100	88%
4	3	март	85/90	94%
5	4	апрель	70/80	88%
6	5	май	68/60	113%
7	6	июнь	65/70	93%

б)

	A	B	C	D
1	дата	время	температура	влажность
2	1 мар	10.00	2	45%
3	2 мар	9.00	-1	42%
4	3 мар	6.00	-5	37%
5	4 мар	9.00	4	53%
6	5 мар	16.00	8	46%
7	6 мар	12.00	6	49%

в)

	A	B	C	D	E	F		
1	Начало		Завершение		Отправлено, МБ	Получено, МБ	Длительность сессии	Стоимость, руб.
2	01.04.2011	22:03	01.04.2011	22:24	1,597	12,314	0:20:58	104,0р.
3	01.04.2011	21:51	01.04.2011	22:00	1,462	9,966	0:09:04	86,0р.
4	01.04.2011	16:50	01.04.2011	18:28	2,052	22,434	1:37:16	294,0р.
5	01.04.2011	14:33	01.04.2011	16:08	3,162	24,188	1:35:17	328,0р.
6	01.04.2011	12:08	01.04.2011	12:20	2,319	10,828	0:11:57	158,0р.

2. Откройте рабочую книгу upr13-2.xls. Оформите указанную учителем таблицу в соответствии с рисунком.

а)

	A	B	C
1			
2	цвет ячейки		
3	красный		
4	зеленый		
5	синий		

б)

	A	B	C	D
1				
2	шрифт			
3	обычный	Arial	Times	
4	жирный	Arial	Times	
5	курсив	Arial	Times	

в)

	A	B	C
1	Результаты		
2	t, с	x, м	v, м/с
3	2,4	6	2,50
4	5,2	12	2,31
5	12,5	23	1,84

г)

	A	B	C	D
1	класс	количество учеников	мальчики	
2	10 А	28	12	43%
3	10 Б	29	15	52%
4	10 В	26	11	42%

д)

	A	B	C	D	E	F
1						
2	текст	текст	текст	текст	текст	текст
3		текст	текст	текст	текст	текст
4		текст	текст	текст	текст	текст
5		текст	текст	текст	текст	текст

е)

	A	B	C	D
1				
2	Города Беларуси	город	основан	
3		Минск	1067	
4		Брест	1019	
5		Витебск	974	
6		Полоцк	862	
7		Гомель	1142	
8		Гродно	1127	
9		Могилев	1267	

3. Откройте рабочую книгу `ur13-3.xls`. Оформите указанную учителем таблицу в соответствии с рисунком.

а)

Климат Минска													
Показатель	Янв	Фев	Мар	Апр	Май	Июн	Июл	Авг	Сен	Окт	Ноя	Дек	Год
Норма осадков, мм	40	34	42	42	62	83	88	72	60	49	52	53	677

б)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Климат Минска													
2	Показатель	Янв	Фев	Мар	Апр	Май	Июн	Июл	Авг	Сен	Окт	Ноя	Дек	Год
3	Абсолютный максимум, °С	10	14	19	26	31	33	34	35	30	25	16	10	35
4	Средний максимум, °С	-2	0	5	12	18	21	22	22	16	10	3	0	10,6
5	Средняя температура, °С	-5	-4	-1	7	13	16	17	17	11	6	0	-3	6,2
6	Средний минимум, °С	-7	-6	-3	3	8	11	12	12	7	3	-2	-5	2,7
7	Абсолютный минимум, °С	-39	-35	-31	-18	-5	0	4	2	-5	-13	-20	-31	-39

§ 14. ИСПОЛЬЗОВАНИЕ ФОРМУЛ. ОТНОСИТЕЛЬНЫЕ И АБСОЛЮТНЫЕ ССЫЛКИ

Рассмотрим на примерах использование формул для выполнения расчетов в электронных таблицах.

Пример 1. Внести в таблицу название, цену и количество купленных принадлежностей (рис. 5.31). Рассчитать их стоимость.

	A	B	C	D
1	Товар	Цена	Количество	Стоимость
2	Тетрадь	1200	3	
3	Карандаш	650	2	
4	Ручка	1600	1	
5	Блокнот	3400	2	

Рис. 5.31

При вычислении стоимости каждого товара нужно его цену (содержимое ячейки в столбце B) умножить на количество (содержимое ячейки в столбце C). Для вычисления стоимости тетрадей в ячейку D2 введем формулу $=B2*C2$. Формулы для расчета стоимости принадлежностей имеют одинаковый вид: формула в ячейке D3 для подсчета стоимости карандашей $=B3*C3$, а формула в ячейке D5 для подсчета стоимости блокнотов $=B5*C5$. Ссылки в этих формулах различаются только номерами строк.

Правообладатель Издательский центр БГУ

Если для различных исходных данных необходимо выполнять повторяющиеся вычисления по одинаковым формулам, то такие формулы можно копировать. При этом в формулах будут изменяться ссылки на ячейки с исходными данными. Например, если формулу $=B2*C2$ из ячейки D2 скопировать в ячейку D3, то в ячейке D3 она примет вид: $=B3*C3$.

При копировании формул в электронных таблицах применяется *относительная адресация*, при которой ссылки на ячейки в формуле изменяются в соответствии с новым положением скопированной формулы. Таким образом, при копировании формулы из одной ячейки в другую номер строки (обозначение столбца) изменяется на столько, на сколько строк (столбцов) сместилась формула.

Формулу из одной ячейки можно скопировать сразу в диапазон ячеек. Выделим в нашем примере ячейку D2 и с помощью меню **Правка**, или кнопки **Копировать**  на панели инструментов, или комбинации клавиш **Ctrl+C** скопируем формулу в буфер обмена. Выделим диапазон ячеек D3 : D5 и с помощью меню **Правка** → **Вставить**, или кнопки **Вставить** , или комбинации клавиш **Ctrl+V** вставим формулу в ячейки диапазона (рис. 5.32).



Для копирования формул в диапазон ячеек можно использовать *автозаполнение*. Для этого выделим ячейку с формулой (в нашем примере D2) и протащим маркер заполнения до ячейки D5. Формула будет скопирована во все ячейки диапазона D3 : D5.

	A	B	C	D
1	Товар	Цена	Количество	Стоимость
2	Тетрадь	1200	3	3600
3	Карандаш	650	2	1300
4	Ручка	1600	1	1600
5	Блокнот	3400	2	6800

Рис. 5.32

Ссылки, которые изменяются при копировании формул, называют *относительными*. По умолчанию относительными являются все ссылки на ячейки в формулах. Механизм относительной адресации работает

и при изменении количества строк и столбцов в электронной таблице. Если в результате удаления (вставки) строк (столбцов) изменился адрес ячейки, на которую ссылается формула, то и ссылка тоже изменится так, чтобы выполнялись прежние вычисления.

Если необходимо, чтобы при копировании формул адреса ячеек не изменялись, используют **абсолютные ссылки**. Для создания абсолютной ссылки нужно поставить знак \$ в формуле перед именем столбца и номером строки, например: \$B\$7. В **смешанных ссылках** при копировании формул остается неизменным только номер строки (например, B\$7) или имя столбца (\$B7).

Изменять тип ссылки в MS Excel удобно нажатием клавиши **F4**. Сначала вводим в формулу адрес ячейки, например B7. По умолчанию он является относительной ссылкой. Затем устанавливаем на него указатель мыши. При нажатии клавиши **F4** ссылка меняется на абсолютную: \$B\$7. Последующие два нажатия **F4** дают смешанные ссылки: B\$7 и \$B7, затем снова появляется исходная относительная ссылка B7.

Пример 2. В ячейках E2 : E5 рассчитать стоимость купленных принадлежностей в российских рублях. Курс перевода поместить в ячейку B7.

Для расчета стоимости товара в российских рублях нужно его стоимость в белорусских рублях разделить на курс перевода из белорусских рублей в российские. В ячейку E2 введем формулу =D2/B7. Для того чтобы ссылка на ячейку B7 не изменялась, сделаем ее абсолютной, т. е. зафиксируем ее адрес (рис. 5.33). Полученную формулу =D2/\$B\$7 скопируем в диапазон ячеек E3 : E5, в которых будет изменяться только ссылка на ячейки столбца D: =D3/\$B\$7, =D4/\$B\$7, =D5/\$B\$7.

СУММ					
	A	B	C	D	E
1	Товар	Цена	Количество	Стоимость	руб РФ
2	Тетрадь	1200	3	3600	=D2/\$B\$7
3	Карандаш	650	2	1300	
4	Ручка	1600	1	1600	
5	Блокнот	3400	2	6800	
6					
7	курс	160			

Рис. 5.33



1. Какие ссылки можно использовать в формулах?
2. В каких случаях применяют относительные ссылки, абсолютные ссылки?
3. В каких из приведенных формул использованы относительные, абсолютные, смешанные ссылки?

а) = \$A\$2+B2;

г) = \$B\$6*C6;

ж) = B\$4/C4;

б) = B3*C\$3;

д) = A2+B2;

з) = B4*D8+4.

в) = \$B\$4/\$D\$4;

е) = \$B\$3*\$D\$3;

Упражнения

1. Откройте рабочую книгу ur14.xls. Выполните задание по указанию учителя.

1.1. В таблице а) приведена цена и количество купленных для школьной библиотеки книг. В столбце D рассчитайте их стоимость.

1.2. В таблице б) приведены тарифы и расход воды. В столбце D рассчитайте стоимость холодной и горячей воды.

1.3. В таблице в) приведены результаты измерений массы и объема 5 образцов. В столбце D рассчитайте плотность вещества.

1.4. В таблице г) приведены тарифы и объемы отправленной и полученной информации. В столбце D рассчитайте расходы на Интернет.

а)

	A	B	C	D
1		цена	к-во	стоимость
2		18,4	5	
3		7,5	12	
4		14,2	3	
5		12	8	
6		6,8	6	

б)

	A	B	C	D
1	Расход и стоимость воды			
2	вода	тариф	расход	стоимость
3	холодная	627	10,4	
4	горячая	43458	17,1	
5				

в)

	A	B	C	D
1		масса, г	объем, см ³	плотность, г/см ³
2	1	12	2,3	
3	2	25	4,7	
4	3	17	2,8	
5	4	9	1,3	
6	5	16	1,9	

г)

	A	B	C	D
1	Расходы на Интернет			
2	Тариф, руб/ Мбайт	Отправлено, Мбайт	Получено, Мбайт	Стоимость, руб.
3	3	17,2	971,3	
4	7,5	14,3	145,2	
5	12	9,6	198,5	

2. В таблице приведены: наименование, цена и количество оборудования. Поместите в столбцы E и F стоимость в условных единицах и рублях.

	A	B	C	D	E	F
1		Наименование	цена, уе	к-во, шт	стоимость	
2					уе	тыс руб
3		системный блок	430	10		
4		монитор	210	12		
5		клавиатура	11	10		
6		мышь	5	12		
7		ИТОГО				
8						
9		курс: тыс руб / уе				



3. Для таблицы из упражнения 1, г рассчитайте расходы на Интернет, если вам предоставили скидку в 5 %. Значение скидки внесите в ячейку А6.



4. Создайте таблицу и вычислите значения функций:

а) $y = x^2 - 3x + 7$ на промежутке $[-6; 6]$, аргумент изменяется с шагом 1;

б) $y = x^2 + 5x - 12$ на промежутке $[-3; 3]$, аргумент изменяется с шагом 0,2.

§ 15. ИСПОЛЬЗОВАНИЕ ФУНКЦИЙ

Для выполнения многих расчетов приходится применять функции. Табличный процессор Excel имеет большую библиотеку стандартных функций, позволяющих выполнять разнообразные вычисления.

Функция вызывается в формуле по имени, после которого в скобках следуют аргументы. Аргументами функции могут быть числа, тексты, ссылки на ячейки или диапазоны ячеек. Если аргументов несколько, то они перечисляются через точку с запятой. Приведем примеры записи некоторых функций в Excel:

Функция	Запись в Excel
Сумма чисел	СУММ(В2:В7)
Среднее значение	СРЗНАЧ(В3:С8; D3:D8)
Максимальное значение	МАКС(В3:D7; F4; F6)
Минимальное значение	МИН(В3:F9)
Сумма квадратов чисел	СУММКВ(D2:G2)
Квадратный корень \sqrt{x}	КОРЕНЬ(3)

Окончание

Функция	Запись в Excel
Синус $\sin x$	SIN(A5)
Косинус $\cos x$	COS(A6)

Пример 1. Вычислить значение $\sqrt{5}$.

Поместим значение аргумента 5 в ячейку A2. Значение функции будем вычислять в ячейке B2. Выделим ячейку B2, введем знак = и наберем имя функции: =КОРЕНЬ(A2). Правильному вводу помогает появившаяся подсказка (рис. 5.34, а). Аргумент функции (адрес ячейки A2) можно набрать с клавиатуры, но лучше ввести выделением этой ячейки мышью. В завершение ввода аргумента не забудем закрыть скобку и щелкнем мышью по любой свободной ячейке. В ячейке B2 будет отображен результат вычисления (рис. 5.34, б).

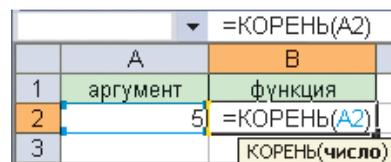


Рис. 5.34, а

Содержащие функции выражения можно вводить также и в строке формул (см. рис. 5.34, а). Эти выражения могут содержать несколько функций.

На практике очень часто приходится вычислять суммы и средние значения диапазонов ячеек таблицы, находить минимальные и максимальные значения. Для этого на панели инструментов имеется кнопка Σ — **Авто-сумма**. Рядом с ней находится значок Σ в выпадающего списка, в котором можно выбрать еще несколько часто встречающихся функций: **Среднее**, **Максимум**, **Минимум** (рис. 5.35), а также перейти к спискам других функций.

Следует быть внимательным при использовании этой кнопки, поскольку табличный процессор автоматически пытается выделить диапазон ячеек, для которых предполагается выполнить выбранное действие. При необхо-

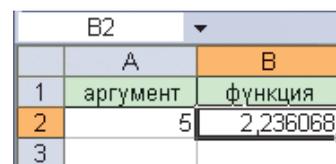


Рис. 5.34, б

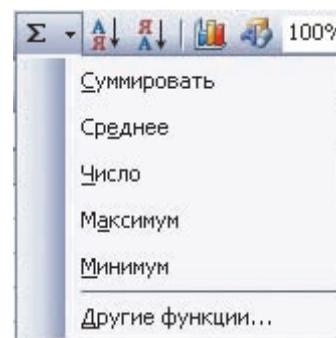


Рис. 5.35

B10		fx =СРЗНАЧ(В3:В9)	
	A	B	C
1		Температура	
2	Дата	день	ночь
3	1 марта	6	-2
4	2 марта	3	-4
5	3 марта	4	-1
6	4 марта	7	0
7	5 марта	8	3
8	6 марта	11	5
9	7 марта	9	2
10	средняя	6,86	
11	максимум		
12	минимум		

Рис. 5.36

дет отображен результат выполнения функции `СРЗНАЧ(В3:В9)`. Аналогичными действиями определим минимальные и максимальные значения температур.

Количество стандартных функций в MS Excel очень велико. Для облегчения поиска они сгруппированы по категориям. Список всех категорий (рис. 5.37) можно вызвать либо нажатием кнопки  в строке

димости это выделение нужно изменить и лишь затем нажать клавишу **Ввод** или кнопку .

Пример 2. Вычислить среднее, минимальное и максимальное значения ночных и дневных температур за первую неделю марта (рис. 5.36).

Выделим ячейку B10. Нажмем кнопку . Из выпадающего списка выберем функцию **Среднее**, выделим требуемый диапазон ячеек и нажмем **Ввод**. В ячейке B10 будет

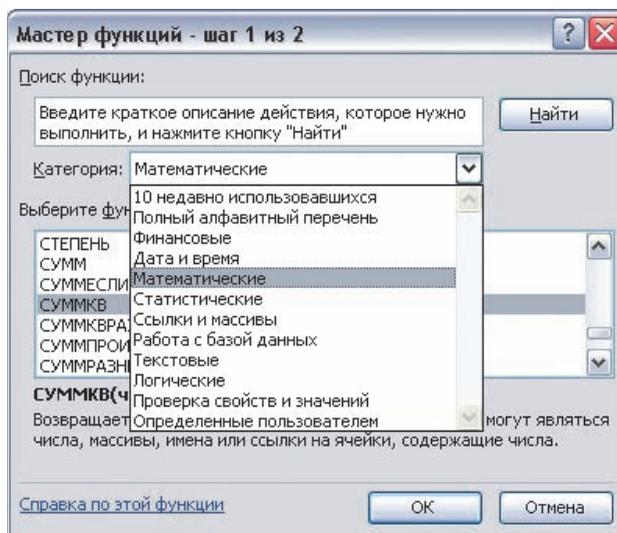


Рис. 5.37

формул, либо выбором пункта **Другие функции** в выпадающем списке кнопки Σ , либо с помощью меню **Вставка** → **Функция**.

Категория **Полный алфавитный перечень** содержит все функции, расположенные в алфавитном порядке. Наиболее часто используются функции из категорий **Дата и время**, **Математические**, **Статистические**. Для повторных вычислений удобна категория **10 недавно использовавшихся**, которая хранит список 10 последних использованных функций. После выбора категории появляется список соответствующих функций (рис. 5.38).

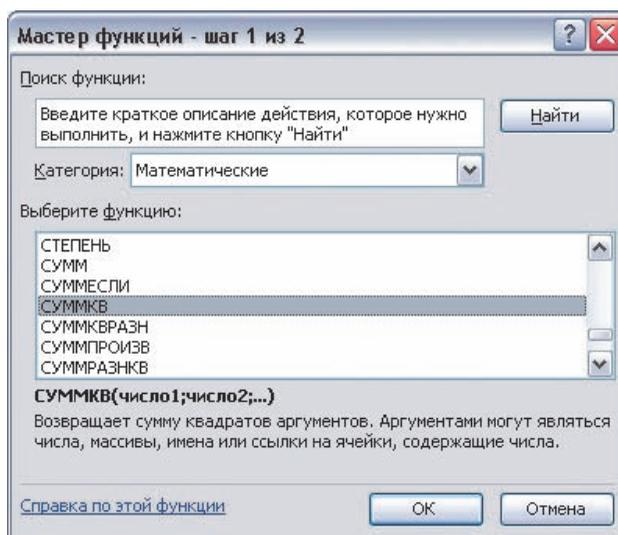


Рис. 5.38

Избежать ошибок при вводе имени функции и адресов ячеек позволяет использование **Мастера функций**. Работа с **Мастером функций** включает два шага. На первом шаге из списка выбирают нужную функцию (см. рис. 5.38). На втором шаге открывается новое окно, в котором вводят аргументы выбранной функции (рис. 5.39). Ссылки лучше не вводить вручную, а указывать мышью ячейку или диапазон.



Пример 3. Даны два катета a и b прямоугольного треугольника. Найти гипотенузу $c = \sqrt{a^2 + b^2}$.

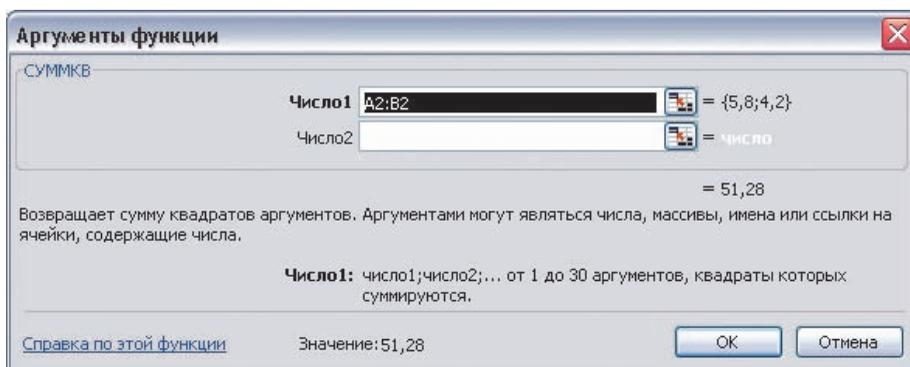


Рис. 5.39

Поместим значения катетов в ячейки A2 и B2, а гипотенузу будем вычислять в ячейке C2. В категории **Математические** есть стандартная функция для вычисления суммы квадратов СУММКВ(A2:B2).

Выделим ячейку C2. Вызовем функцию КОРЕНЬ() и вложим в нее функцию СУММКВ(A2:B2). Расчетная формула в Excel примет вид: =КОРЕНЬ(СУММКВ(A2:B2)) (рис. 5.40).

	A	B	C	D
1	катет a	катет b	гипотенуза c	
2	5,8	4,2	=КОРЕНЬ(СУММКВ(A2:B2))	
3				

Рис. 5.40

Заметим, что можно вести расчет и по формуле =КОРЕНЬ(A2^2+B2^2).



1. Что может быть аргументом функции в Excel?
2. По каким основным категориям сгруппированы функции в Excel?
3. Какими способами можно вызвать список категорий функций?

Упражнения

1. Откройте рабочую книгу ur15.xls. Выполните задание по указанию учителя.

1.1. По таблице определите численность населения, площадь территории и среднюю плотность населения Республики Беларусь, а также плотность населения областей.

	А	В	С	Д
1	область	численность	площадь	плотность
2		тыс чел	тыс кв км	чел на кв км
3	Брестская	1 398,7	32,8	42,8
4	Витебская	1 228,6	40,0	30,6
5	Гомельская	1 438,3	40,4	35,6
6	Гродненская	1 069,6	25,1	42,6
7	Минская	1 418,9	39,9	35,3
8	Могилевская	1 091,9	29,1	37,5
9	Минск	1 834,2	0,3055	6003,9
10				

1.2. По таблице «Реки Беларуси» определите, сколько процентов от общей длины реки приходится на территорию Беларуси. Определите самую длинную и самую короткую реку на территории Беларуси.

	А	В	С	Д
1	река	общая длина, в Беларуси,		
2		км	км	%
3	Березина	613	613	
4	Вилия	498	264	
5	Днепр	2145	689	
6	Западная Двина	1020	328	
7	Западный Буг	772	154	
8	Нёман	937	459	
9	Припять	761	500	
10	Птичь	421	421	
11	Свислочь	285	285	
12	Сож	648	493	
13	Щара	325	325	

1.3. По таблице «Озера Беларуси» определите самое глубокое и самое большое по площади озеро.

	А	В	С
1	Озеро	площадь, кв.км	глубина, м
2	Выгощанское	26	2,5
3	Дривяты	36,1	12
4	Лукомское	37,7	11,5
5	Мядель	16,2	24,6
6	Нарочь	79,6	24,8
7	Нещердо	24,6	8,1
8	Освейское	52,8	7,5
9	Свирь	22,3	8,7
10	Снуды	22	16,5
11	Червоное	40,8	2,9
12	Чёрное	17,3	3

1.4. По таблице «Расстояния между городами Беларуси» определите два самых удаленных города. Определите время движения автомобиля между ними при средней скорости 70 км/ч.

	А	В	С	Д	Е	Ф	Г
1	Расстояния между городами Беларуси, км						
2	город	Минск	Брест	Витебск	Гомель	Гродно	Могилев
3	Минск						
4	Брест	349					
5	Витебск	277	625				
6	Гомель	308	535	342			
7	Гродно	268	259	545	576		
8	Могилев	210	524	164	182	478	



2. Создайте таблицу для перевода градусов в радианы. Аргумент изменяется от 0 до 90° с шагом 5°. Вычислите значения тригонометрических функций sin и cos.

	А	В	С	Д
1	угол		функция	
2	град	рад	sin	cos
3	0			
4	5			

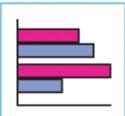
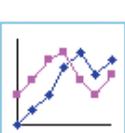
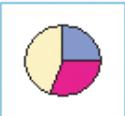


3. Определите, в какой день недели вы родились (функция ДЕНЬНЕД). Вставьте в ячейку С2 текущую дату и время (функция ТДАТА). Просмотрите справки по этим функциям.

	А	В	С
1	Дата рождения	День недели	Текущая дата и время
2	01.09.1995		01.05.2011 21:19

§ 16. ПОСТРОЕНИЕ ДИАГРАММ

Важной особенностью электронных таблиц является возможность наглядного представления и анализа данных с помощью диаграмм. *Диаграммой* называют графическое отображение числовых данных и соотношений между ними. Табличный процессор Excel позволяет строить диаграммы различных *типов*.

<p>В <i>гистограмме</i> каждое значение отображается вертикальным столбиком соответствующей высоты. Гистограммы используют для сравнения дискретных величин.</p>	
<p><i>Линейчатая диаграмма</i> — это гистограмма, повернутая на 90°. В ней легче располагать и читать подписи, особенно если они длинные.</p>	
<p><i>Графики</i> используются для иллюстрации непрерывных изменений. Они полезны для выявления тенденций, например, роста населения или убывания радиации с течением времени.</p>	
<p>Для демонстрации зависимостей между двумя переменными используются <i>точечные диаграммы</i> (в математике их называют <i>графиками зависимостей</i>).</p>	
<p><i>Круговая диаграмма</i> иллюстрирует соотношение величин. Представляет собой круг, разделенный на секторы пропорционально отображаемым значениям.</p>	
<p>В <i>лепестковой диаграмме</i> значения отмечаются на осях, направленных наружу от центра. Для каждой категории определена своя ось. В розе ветров, например, категориями служат направления ветров, а значениями — количество дней с преобладанием ветров этих направлений.</p>	

Каждый тип диаграмм в свою очередь содержит несколько *видов*, что позволяет выбрать наиболее наглядный способ графического представления данных. Эти данные вводятся в таблицу до построения диа-

граммы. При изменении данных в таблице диаграмма автоматически обновляется.

Диаграмма в Excel содержит несколько различных объектов (рис. 5.41).

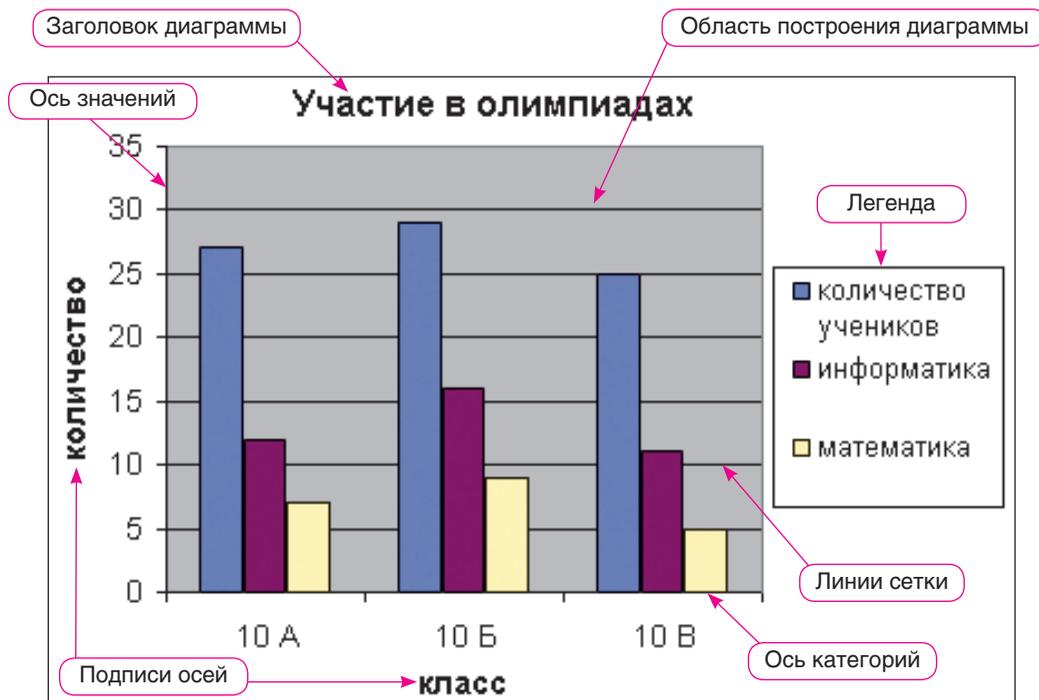


Рис. 5.41

Выполнять построения удобно с помощью **Мастера диаграмм**, который вызывается из меню **Вставка** → **Диаграмма** или кнопки  на панели инструментов. Работа с **Мастером диаграмм** состоит из четырех шагов. Нужно пошагово выполнять его указания и нажимать кнопку **Далее**.

Пример 1. Таблица содержит данные об участии учеников 10 классов в олимпиадах по информатике и математике (рис. 5.42). Построить гистограмму, иллюстрирующую эти данные.

Выделим ячейки таблицы, включая строку заголовков (в этом случае они будут использованы при построении диаграммы).

Правообладатель Издательский центр БГУ

	А	В	С	Д
1	класс	количество учеников	информатика	математика
2	10 А	27	12	7
3	10 Б	29	16	9
4	10 В	25	11	5

Рис. 5.42

Шаг 1. Выберем тип и вид диаграммы (рис. 5.43). При этом полезно посмотреть результат выбора (рис. 5.44), нажав левой кнопкой мыши кнопку **Просмотр результата** и удерживая ее в нажатом состоянии. Нажмем кнопку **Далее**.

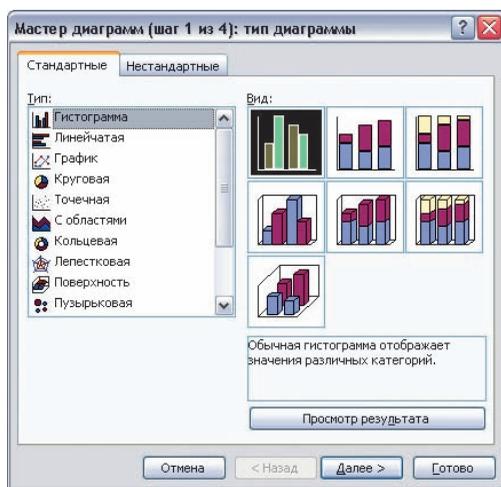


Рис. 5.43

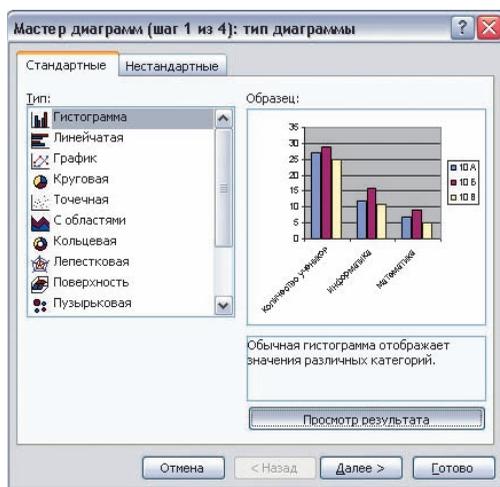


Рис. 5.44

Шаг 2. Поскольку диапазон ячеек был выделен, то в открывшемся окне появляется вариант гистограммы. Данные сгруппированы в соответствии с установленным флажком **Ряды в строках**: заголовки столбцов таблицы отражены в качестве подписей категорий, а нумерация классов — в легенде (рис. 5.45).

Установим флажок **Ряды в столбцах**. Теперь данные сгруппированы по классам, а в легенде показаны заголовки столбцов (рис. 5.46).

Правообладатель Издательский центр БГУ



Рис. 5.45



Рис. 5.46

При необходимости диапазон данных можно изменить.

Шаг 3. Установим параметры нашей диаграммы. На вкладке **Заголовки** введем **Название диаграммы**: Участие в олимпиадах. Подпишем оси: **Ось X (категорий)** — класс, **Ось Y (значений)** — количество (рис. 5.47). Остальные параметры на всех вкладках оставим без изменений.

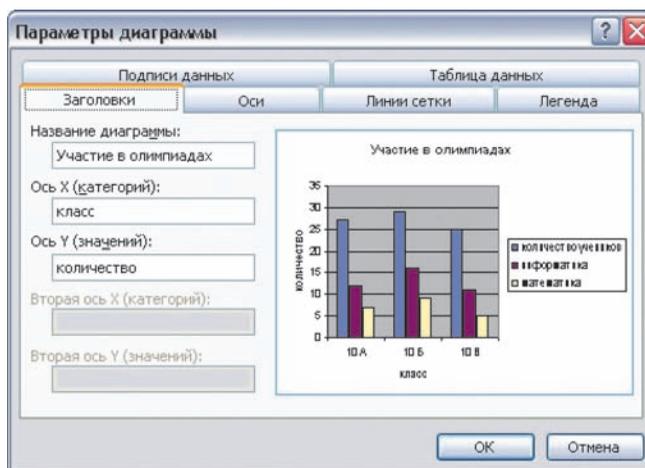


Рис. 5.47

Шаг 4. На последнем шаге подтвердим, что диаграмма будет размещена на имеющемся листе (рис. 5.48).

В результате получим диаграмму, показанную на рис. 5.41.

Правообладатель Издательский центр БГУ

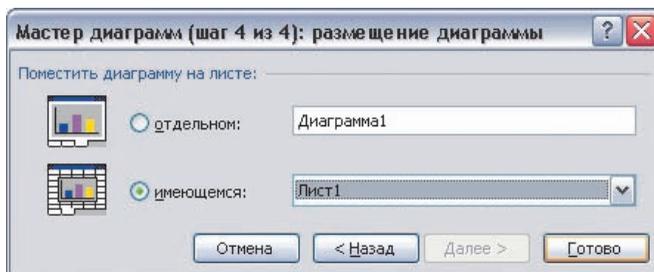


Рис. 5.48

Изменить вид и другие параметры созданной диаграммы можно с помощью панели **Диаграммы** (рис. 5.49), которая открывается из меню **Вид** → **Панели инструментов**.



Рис. 5.49

В выпадающем списке **Область диаграммы** выбирают элемент диаграммы, который требует изменения, например область построения диаграммы (рис. 5.50). Любой элемент диаграммы, так же как и диаграмма в целом, является объектом. Двойным щелчком по этому элементу можно вызвать соответствующее диалоговое окно и произвести необходимые установки, например, в окне **Формат области построения** выбрать светло-зеленый цвет (рис. 5.51).

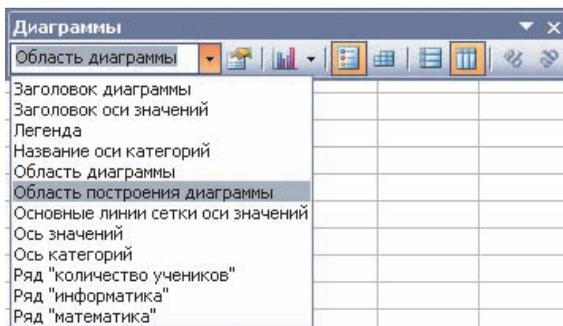


Рис. 5.50

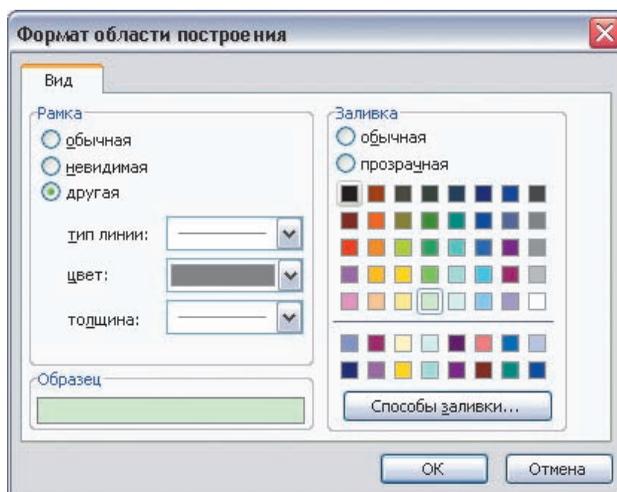


Рис. 5.51

Многообразие типов диаграмм обусловлено тем, что каждый тип наиболее наглядно отражает определенные особенности данных в таблице. Так, для демонстрации вклада каждого значения в общую сумму используют круговые диаграммы.

	А	В
1	категория	тыс кв км
2	Леса	70,6
3	Болота	22,8
4	Пашня	58,1
5	Сенокосы	33,2
6	Др земли	22,8

Пример 2. По данным приведенной таблицы построить круговую диаграмму, иллюстрирующую структуру земель Беларуси.

- Выделим диапазон ячеек А1:В6. Вызовем **Мастер диаграмм**.

- На первом шаге выберем тип диаграммы — **Круговая**.

- Параметры на втором шаге будут выбраны автоматически.

- На третьем шаге введем заголовок: **Распределение земель** и установим флажок **Доли** на вкладке **Подписи данных**.



Рис. 5.52

Завершив построение диаграммы, с помощью панели **Диаграммы** изменим цвет фона и размер символов (рис. 5.52).

Правообладатель Издательский центр БГУ

Для построения графиков функциональных зависимостей используется тип **Точечная диаграмма**.

Пример 3. Построить график функции $y = x^2 - 3x - 10$ на промежутке $[-4; 6]$ с шагом 1.

Сначала создадим таблицу значений аргумента и функции (рис. 5.53).

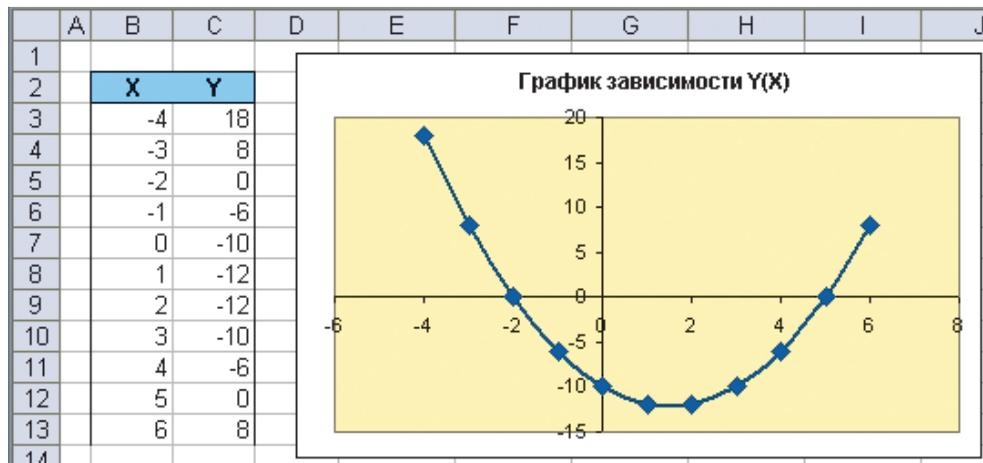


Рис. 5.53

Затем выделим таблицу вместе с заголовками (диапазон ячеек B2:C13) и вызовем **Мастер диаграмм**. Установим тип диаграммы — **Точечная**, вид — **Со значениями, соединенными сглаживающими линиями**.

На третьем шаге в окне **Мастера диаграмм** введем название диаграммы — **График зависимости Y(X)**. На вкладке **Линии сетки** сбросим все флажки. Удалим легенду (она необходима, если нужно различать несколько диаграмм).

С помощью панели **Диаграммы** установим подходящий цвет фона. Результат показан на рис. 5.53.



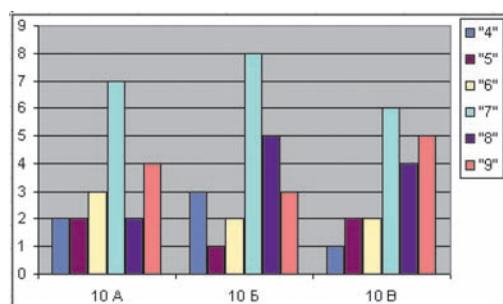
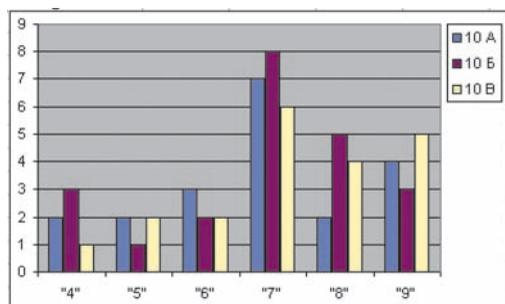
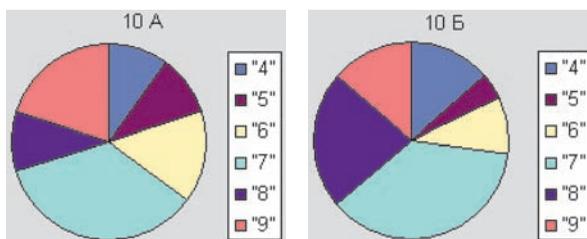
1. Для чего используют диаграммы?
2. Каковы наиболее распространенные типы диаграмм?
3. Из каких объектов состоит диаграмма?

Упражнения

1. Откройте рабочую книгу urg16.xls. Выполните задание по указанию учителя.

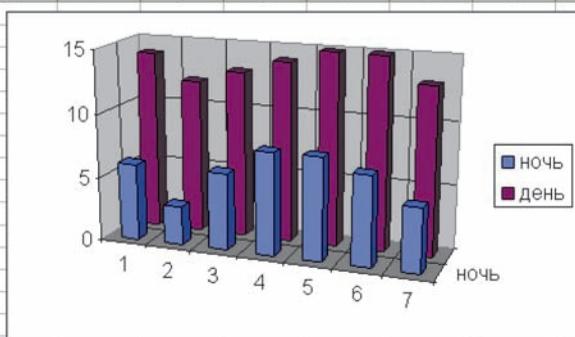
1.1. В таблице приведены оценки учащихся трех классов. Постройте гистограммы и круговые диаграммы. Выберите наилучшее представление.

	A	B	C	D	E
1					
2		класс			
3	оценка	10 А	10 Б	10 В	
4	"4"	2	3	1	
5	"5"	2	1	2	
6	"6"	3	2	2	
7	"7"	7	8	6	
8	"8"	2	5	4	
9	"9"	4	3	5	



1.2. В таблице приведены средние дневные и ночные температуры. Вычислите средние температуры за неделю. Постройте объемные гистограммы.

	A	B	C	D	E	F	G	H	I	J	K
1											
2		температура									
3		дата	ночь	день							
4		1 мая	6	14							
5		2 мая	3	12							
6		3 мая	6	13							
7		4 мая	8	14							
8		5 мая	8	15							
9		6 мая	7	15							
10		7 мая	5	13							
11		ср знач									
12											
13											
14											
15											
16											

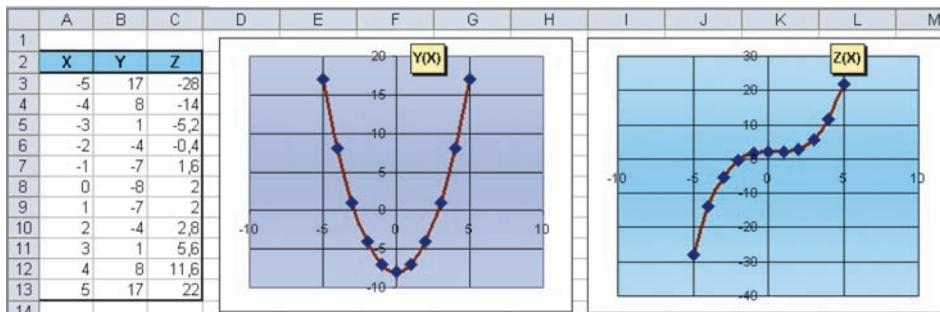


1.3. По таблице ветров постройте лепестковую диаграмму (розу ветров).

	А	В	С
1	Направление ветра	март	апрель
2	Северный	2	3
3	Северо-восточный	3	2
4	Восточный	4	3
5	Юго-восточный	3	2
6	Южный	3	5
7	Юго-западный	6	7
8	Западный	6	5
9	Северо-западный	4	3



2. Постройте графики функций (точечные диаграммы) $y(x) = x^2 - 8$ и $z(x) = (x^3 - x^2)/5 + 2$ на промежутке $[-5; 5]$ с шагом 1.



3. Постройте графики функций:

- а) $y = 0,5x^2 - 3x + 5$ на промежутке $[-3; 3]$;
- б) $y = x^3 + x + 2$ на промежутке $[-5; 5]$;
- в) $y = |3x - 4|$ на промежутке $[-10; 10]$;
- г) $y = \cos x$ на промежутке $[-10; 10]$;
- д) $y = x \sin x$ на промежутке $[-10; 10]$.

§ 17. СОРТИРОВКА ДАННЫХ В ТАБЛИЦЕ

Сортировкой называют упорядочение данных в таблице. Числовые данные, время и даты можно расположить в порядке возрастания или убывания, а текстовые — по алфавиту или в порядке, обратном алфавитному.

Проще всего произвести сортировку с помощью кнопок на панели инструментов Сортировка по возрастанию или Сортировка по убыванию.

Пример 1. Отсортировать данные в таблице, изображенной на рис. 5.54, а, расположив названия городов в алфавитном порядке.

	А	В
1	город	основан
2	Минск	1067
3	Гомель	1142
4	Могилев	1267
5	Витебск	974
6	Полоцк	862
7	Гродно	1127
8	Брест	1019

Рис. 5.54, а

	А	В
1	город	основан
2	Брест	1019
3	Витебск	974
4	Гомель	1142
5	Гродно	1127
6	Минск	1067
7	Могилев	1267
8	Полоцк	862

Рис. 5.54, б

	А	В
1	город	основан
2	Полоцк	862
3	Витебск	974
4	Брест	1019
5	Минск	1067
6	Гродно	1127
7	Гомель	1142
8	Могилев	1267

Рис. 5.54, в

Откроем рабочую книгу `prim17-1.xls`. Выделим ячейку с названием любого города, например А3, и щелкнем мышью по кнопке  **Сортировка по возрастанию**. В результате в алфавитном порядке названий городов будут отсортированы данные обоих столбцов, т. е. целые строки таблицы (рис. 5.54, б). При этом первая строка таблицы не подвергается сортировке. По умолчанию Excel автоматически выделяет диапазон ячеек и считает, что в первой строке находятся названия столбцов.

	А	В	С	Д	Е
1	Фамилия	Информатика	Математика	Физика	Химия
2	Гончарова Ольга	9	7	8	6
3	Романов Петр	7	6	5	5
4	Васильков Антон	9	7	6	6
5	Александров Павел	9	8	7	7
6	Котов Дмитрий	9	8	8	7
7	Красовская Оксана	7	7	7	6
8	Смусин Виктор	9	9	9	7
9	Батиков Андрей	8	9	8	7
10	Шацкий Владислав	8	8	7	6
11	Буянов Сергей	9	9	9	8
12	Захарова Ольга	9	9	9	8
13	Амелина Екатерина	9	9	8	7
14	Крошин Владимир	8	8	8	6
15	Кудинов Виктор	7	6	6	5
16	Сонина Елена	8	7	7	6
17	Петровский Максим	9	9	8	7
18	Яковенко Илья	7	6	4	5
19	Миროнова Мария	7	6	6	6
20	Нефедова Наталья	9	7	8	7

Рис. 5.55, а

Аналогично выполняется сортировка строк таблицы по году основания города. Для этого выделяется ячейка второго столбца, например В5 (рис. 5.54, в).

Более сложные условия сортировки можно задать в диалоговом окне **Сортировка диапазона**, которое вызывается из меню **Данные**. В нем можно выбрать от одного до трех столбцов (или строк) сортировки

и определить свой порядок сортировки по каждому из них.

Пример 2. В таблицу занесены оценки учеников 10 класса по четырем предметам. Отсортировать учеников по убыванию балла по информатике, затем по математике и, наконец, по фамилиям в алфавитном порядке.

Откроем рабочую книгу prim17-2.xls. Фрагмент исходной таблицы приведен на рис. 5.55, а (всего в ней хранятся данные для 25 учеников).

Выделим диапазон ячеек A1:E26. Из меню **Данные** вызовем диалоговое окно **Сортировка диапазона**. Зададим условия: **Сортировать по** — **Информатика** (по убыванию), **Затем по** — **Математика** (по убыванию), **В последнюю очередь, по** — **Фамилия** (по возрастанию) (рис. 5.56). При этом должен быть установлен флажок **Идентифицировать диапазон данных по подписям** (первая строка диапазона).

Результат сортировки показан на рис. 5.55, б (для наглядности отсортированные оценки отмечены цветом).

	A	B	C	D	E
	Фамилия	Информатика	Математика	Физика	Химия
1					
2	Амелина Екатерина	9	9	8	7
3	Буянов Сергей	9	9	9	8
4	Захарова Ольга	9	9	9	8
5	Петровский Максим	9	9	8	7
6	Смусин Виктор	9	9	9	7
7	Александров Павел	9	8	7	7
8	Котов Дмитрий	9	8	8	7
9	Васильков Антон	9	7	6	6
10	Гончарова Ольга	9	7	8	6
11	Нефедова Наталья	9	7	8	7
12	Батиков Андрей	8	9	8	7
13	Крошин Владимир	8	8	8	6
14	Шацкий Владислав	8	8	7	6
15	Сонина Елена	8	7	7	6
16	Лемешева Ольга	8	6	6	6
17	Красовская Оксана	7	7	7	6
18	Полещук Татьяна	7	7	6	6
19	Кудинов Виктор	7	6	6	5
20	Миронова Мария	7	6	6	6

Рис. 5.55, б

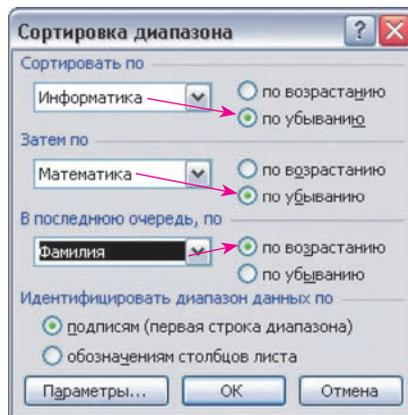


Рис. 5.56



Операция **фильтрации**, в отличие от сортировки, не меняет порядок строк. В отфильтрованном списке отображаются только строки, отвечающие **условиям** отбора данных, а остальные временно скрываются. Отфильтрованную таблицу можно редактировать, форматировать, выводить на печать; для нее можно создавать диаграммы, не изменяя порядок строк и не перемещая их.



Пример 3. В таблице примера 2 отобразить информацию только о тех учениках, которые получили по математике 8 баллов.

Выделим ячейку в первой строке с заголовками столбцов, например B1 (Информатика). В меню **Данные** → **Фильтр** выберем **Автофильтр**. В строке заголовков появятся кнопки фильтров столбцов со стрелками ▼ (рис. 5.57).

Нажмем на кнопку фильтра столбца **Математика** и в выпадающем меню выберем значение 8 (рис. 5.58). В отфильтрованной таблице покажутся 4 строки (рис. 5.59). Продолжим фильтрацию, выбрав, например, оценку 8 по физике. В результате будут показаны только две строки.

Для отмены действия фильтра достаточно нажать на его кнопку и в выпадающем меню выбрать пункт **(Все)** (см. рис. 5.58). Вернуться к отображению всей таблицы можно и с помощью меню **Данные** → **Фильтр** → **Отобразить все**.

	A	B	C	D	E
1	Фамилия	Информатика	Математика	Физика	Уч. мия
2	Гончарова Ольга	9	7	8	6
3	Романов Петр	7	6	5	5
4	Васильков Антон	9	7	6	6

Рис. 5.57

	A	B	C	D	E
1	Фамилия	Информатика	Математика	Физика	Уч. мия
2	Г	Сортировка по возрастанию		8	6
3	Р	Сортировка по убыванию		5	5
4	Е	(Все)		6	6
5	А	(Первые 10...)		7	7
6	К	(Условие...)		8	7
7	К	5		7	6
8	С	7		9	7
9	Е	8		8	7
10	Щацкий Владислав	9		7	6

Рис. 5.58

	A	B	C	D	E
1	Фамилия	Информатика	Математика	Физика	Уч. мия
5	Александров Павел	9	8	7	7
6	Котов Дмитрий	9	8	8	7
10	Щацкий Владислав	8	8	7	6
14	Крошин Владимир	8	8	8	6

Рис. 5.59



1. Что называют сортировкой?
2. По каким параметрам можно сортировать данные электронной таблицы?
3. Какой порядок сортировки можно задать для числовых данных? Для текстовых данных?

Упражнения

1. Откройте рабочую книгу `urp17.xls`. Выполните задание по указанию учителя.

1.1. В таблице представлены данные о площади и глубине озер Беларуси. Отсортируйте данные таблицы по следующим условиям:

- а) в алфавитном порядке — области, затем — названия озер;
- б) площадь — по убыванию;
- в) глубина — по возрастанию.

1.2. В таблице представлены данные о численности населения городов Беларуси. Отсортируйте данные таблицы по следующим условиям:

- а) в алфавитном порядке — области, затем — названия городов;
- б) в алфавитном порядке — области, затем численность населения — по убыванию.

1.3. В таблице представлены результаты соревнований по двум видам спорта. Отсортируйте данные таблицы по следующим условиям:

- а) в алфавитном порядке — фамилии спортсменов;
- б) результаты бега — по возрастанию;
- в) результаты прыжков в длину — по убыванию.



2. В таблице представлена информация о странах Европы. С помощью сортировки и автофилтра найдите:

- а) страну, в которой протекает река длиной 600 км;
- б) страны, названия которых начинаются на букву «И»;
- в) страны, площадь которых составляет от 100 до 300 тыс. кв. км;
- г) страны, население которых составляет от 10 до 50 млн чел.

§ 18. ВЫПОЛНЕНИЕ ПРАКТИЧЕСКИХ ЗАДАНИЙ ИЗ РАЗЛИЧНЫХ ПРЕДМЕТНЫХ ОБЛАСТЕЙ

18.1. РЕШЕНИЕ РАСЧЕТНЫХ ЗАДАЧ

Рассмотрим примеры использования электронных таблиц для выполнения практических заданий из различных предметных областей.

Использование электронных таблиц существенно облегчает решение расчетных задач, особенно с громоздкими вычислениями, например, в химии при приготовлении растворов или в физике тепловых явлений. Последовательность выполняемых при этом действий соот-

ответствует этапам традиционного способа решения задачи: анализ условия и применимых закономерностей, ввод исходных данных, ввод требуемых формул, получение и анализ результатов.

Пример 1. Вычислить площадь треугольника, если известны длины трех сторон.

Решение задачи основано на применении формулы Герона

$$S = \sqrt{p(p-a)(p-b)(p-c)}, \text{ где полупериметр } p = (a + b + c) / 2.$$

Введем исходные данные a , b и c в ячейки А3, В3 и С3 соответственно.

Для лучшего понимания решения и упрощения формул полезно отображать в ячейках промежуточные результаты. В ячейке D3 вычислим p по формуле $= (A3+B3+C3) / 2$. Площадь рассчитаем в ячейке E3 по формуле $= \text{КОРЕНЬ}(D3*(D3-A3)*(D3-B3)*(D3-C3))$.

Результат показан на рис. 5.60.

	А	В	С	D	E
1	Дано				результат
2	a	b	c	p	S
3	12	7	9	14	31,3

Рис. 5.60

Пример 2. В 20 %-ный раствор кислоты массой 200 г добавляют 100 г воды. Определить массовую долю кислоты в полученном растворе.

Введем исходные данные: массу раствора — в ячейку В2, массовую долю кислоты — в В3, а массу добавленной воды — в В4.

В химии массовой долей вещества в растворе называют отношение массы растворенного вещества к массе раствора $w = m_b / m_p$ и выражают в процентах. Отсюда получим формулу расчета массы кислоты: $m_b = m_p \cdot w / 100$. Введем эту формулу в ячейку В5: $=B2*B3/100$. В ячейке С2 вычислим суммарную массу раствора: $=B2+B4$. Наконец, в ячейке С3 вычислим массовую долю кислоты в полученном растворе: $=100*B5/C2$.

Результат показан на рис. 5.61.



Пример 3. Пузырек воздуха поднимается со дна озера глубиной 20 м на поверхность. Как изменится объем пузырька? Считать температуру у дна и поверхности одинаковой. Атмосферное давление 10^5 Па.

	А	В	С
1	Дано		результат
2	масса раствора, г	200	300
3	массовая доля, %	20	13,3
4	добавлено воды, г	100	
5	масса кислоты, г	40	

Рис. 5.61

Решение задачи основано на применении закона изотермического расширения газа $P_1 \cdot V_1 = P_2 \cdot V_2$. Давление на глубине h равно сумме атмосферного и гидростатического давления $P + \rho gh$.

Отсюда $V_2 / V_1 = 1 + \rho gh / P$.

Введем исходные данные: глубину озера h — в ячейку В2, атмосферное давление P , плотность воды ρ и ускорение силы тяжести g — в ячейки В3, В4, В5 соответственно. В ячейку В6 введем расчетную формулу $=1+B4*B5*B2/B3$.

Результат показан на рис. 5.62.

Многие задачи требуют перевода данных в одну систему единиц (СИ). Для этого удобно использовать соседний столбец или строку.

	А	В
1	Дано	
2	Глубина озера h , м	20
3	Атмосферное давление P , Па	1,00E+05
4	Плотность воды ρ , кг/м ³	1000
5	Ускорение силы тяжести g , м/с ²	9,8
6	результат V_2/V_1	2,96

Рис. 5.62



Пример 4. В теплоизолированный сосуд, в котором находится 500 г воды при температуре 30 °С, бросают кусок льда массой 50 г, имеющий температуру 0 °С. Какой станет температура воды после плавления льда и установления теплового равновесия?

Введем данные условия задачи в ячейки В2:В5. В ячейках С2:С5 приведем их к системе СИ: 1 кг = 1000 г, 1 К = 1 °С + 273. Искомую температуру определим из уравнения теплового баланса

$$Cm_{\text{в}}(T_{\text{в}} - T) = Lm_{\text{л}} + Cm_{\text{л}}(T - T_{\text{л}}).$$

Массу воды после плавления льда вычислим в ячейке D2 := C2+C4, а температуру в К — в ячейке D3 по формуле $=(C6*C2*C3+C6*C4*C5-C7*C4)/(C6*D2)$.

Полученную температуру пересчитаем в градусы Цельсия в ячейке E3.

Результат показан на рис. 5.63.

	А	В	С	Д	Е
1	Дано		единицы СИ	результаты	
2	Масса воды $m_в$	500	0,5	0,55	
3	Температура воды $t_в$	30	303	293,13	20,13
4	Масса льда $m_л$	50	0,05		
5	Температура льда $t_л$	0	273		
6	Теплоемкость воды C , Дж/(кг К)		4,20E+03		
7	Уд. теплота плавления L , Дж/кг		3,30E+05		

Рис. 5.63

Упражнения

1. Сколько граммов растворенного вещества и растворителя потребуется, чтобы получить 200 г раствора с массовой долей 12 %?

2. Из 400 г 20 %-ного раствора при охлаждении выделилось 50 г растворенного вещества. Определите массовую долю вещества в оставшемся растворе.

	А	В
1	Дано	
2	Масса раствора, г	200
3	Массовая доля, %	12
4	Масса вещества, г	?
5	Масса растворителя, г	?

	А	В	С
1	Дано		результат
2	Масса раствора, г	400	?
3	Массовая доля, %	20	?
4	выделилось, г	50	
5	осталось вещества, г	?	

3. При температуре 27 °С и давлении 10^5 Па объем заполненного гелием воздушного шара равен 500 м³. Каким станет объем этого шара при подъеме на высоту, где давление равно $5 \cdot 10^4$ Па, а температура — 33 °С? Массу шара считать неизменной.

4. В теплоизолированном сосуде смешивают 100 г воды, имеющей температуру 20 °С, и 300 г — температуру 80 °С. Какой станет температура воды после установления теплового равновесия?



5. Определите массовую долю вещества в растворе, полученном в результате смешивания двух растворов, имеющих массу 200 г и 300 г и массовые доли 30 % и 5 %.

	A	B	C	D
1	Дано	раствор1	раствор2	результат
2	Масса раствора, г	200	100	?
3	Массовая доля, %	30	5	?
4	Масса растворенного вещества, г	?	?	?



6. В сосуд, в котором находится 200 г воды при температуре 20 °С, бросают медный шарик массой 20 г, нагретый до температуры 70 °С. Какой станет температура воды после установления теплового равновесия? Теплоемкость воды 4200 Дж/(кг·К), меди — 390 Дж/(кг·К).

18.2. ИССЛЕДОВАНИЕ ЗАВИСИМОСТЕЙ

Использование электронных таблиц оказывается незаменимым при выполнении многократно повторяющихся вычислений и исследовании зависимостей в различных предметных областях.

Пример 1. Создать таблицу для расчета стоимости проезда на автомобиле в зависимости от расстояния, цены и расхода топлива.

Введем значения расхода топлива и его цены в ячейки A2 и B2. Диапазон ячеек A5:A16 заполним рядом значений расстояний от 10 до 120 км с шагом 10. В ячейку B5 введем формулу для расчета стоимости проезда = A\$2*B\$2*A5/100. Распространим эту формулу на диапазон B5:B16.

Полученный результат показан на рис. 5.64.

Изменяя значения расхода топлива и цены, будем получать стоимость проезда в зависимости от расстояния.

Пример 2. Используя таблицу (рабочая книга primer18.xls), вычислить долю (в %) запасов и добычи нефти

	A	B
1	Расход, л на 100 км	Цена, усл. ед.
2	5,6	2,8
3		
4	расстояние, км	стоимость
5	10	1,568
6	20	3,136
7	30	4,704
8	40	6,272
9	50	7,84
10	60	9,408
11	70	10,976
12	80	12,544
13	90	14,112
14	100	15,68
15	110	17,248
16	120	18,816

Рис. 5.64

основных нефтедобывающих стран от общемировых. Оценить, на сколько лет хватит нефти в этих странах при указанном уровне ее добычи.

В ячейке В19 определим суммарные общемировые запасы: =СУММ(В3:В18), а в D19 — ежедневную добычу нефти: =СУММ(D3:D18).

В ячейке С3 вычислим долю запасов Саудовской Аравии: =100*В3/\$В\$19, а в Е3 — долю этой страны в ежедневной добыче: =100*D3/\$D\$19. Распространим эти формулы на соответствующие диапазоны: на С3:С19 и Е3:Е19.

В ячейке F3 вычислим количество лет: =1000*В3/D3/365 (запасы в млрд баррелей переводим в млн и делим на ежедневную добычу и 365 дней в году). Распространим эту формулу на диапазон F3:F19.

Результаты показаны на рис. 5.65.

Анализ полученных значений показывает, что при данных условиях дольше всего нефти хватит в Ираке (большие запасы и низкая добыча).

	А	В	С	Д	Е	Ф
1		Запасы		Ежедневная добыча		
2	страна	млрд бар	доля, %	млн бар/день	%	на сколько лет хватит
3	Саудовская Аравия	264,1	20,99	10,846	13,26	67
4	Иран	137,6	10,94	4,325	5,29	87
5	Ирак	115	9,14	2,423	2,96	130
6	Кувейт	101,5	8,07	2,784	3,40	100
7	Венесуэла	99,4	7,90	2,566	3,14	106
8	ОАЭ	97,8	7,77	2,98	3,64	90
9	Россия	79	6,28	9,886	12,08	22
10	Ливия	43,7	3,47	1,846	2,26	65
11	Казахстан	39,8	3,16	1,554	1,90	70
12	Нигерия	36,2	2,88	2,17	2,65	46
13	США	30,5	2,42	6,736	8,23	12
14	Канада	28,6	2,27	3,238	3,96	24
15	Катар	27,3	2,17	1,378	1,68	54
16	Китай	15,5	1,23	3,795	4,64	11
17	Ангола	13,5	1,07	1,875	2,29	20
18	Остальные страны	128,5	10,21	23,418	28,62	15
19	Весь мир	1258	100,00	81,82	100,00	42

Рис. 5.65

В США приведенные запасы могут быть исчерпаны уже через 12 лет, а в России — через 22 года (средние запасы и большая добыча). Общемировых запасов может хватить примерно на 42 года.

В заключение интересно исследовать, до каких значений надо уменьшить ежедневную добычу, чтобы мировых запасов хватило на 100 лет.

Упражнения

1. Откройте рабочую книгу upr18-1.xls с таблицей расстояний от Минска до столиц европейских стран. Вычислите время в пути при средней скорости автобуса 65 км/ч, а самолета 750 км/ч. С какой скоростью должен двигаться автобус, чтобы доехать до Вильнюса за то же время, что самолет долетит до Берлина?

	А	В	С	Д
1	город	расстояние, км	время, час	
2			автобус	самолет
3	Москва	720	11,1	0,96
4	Киев	560	8,6	0,75
5	Вильнюс	190	2,9	0,25
6	Рига	530	8,2	0,71
7	Варшава	570	8,8	0,76
8	Берлин	1100	16,9	1,47
9	Вена	1300	20,0	1,73
10	Прага	1150	17,7	1,53
11	Париж	2180	33,5	2,91
12	Рим	2370	36,5	3,16
13				
14	средн. скорость	км/ч	65	750

2. Откройте рабочую книгу upr18-2.xls с таблицей расстояний между городами Беларуси. Создайте таблицу и исследуйте зависимость времени нахождения в пути от средней скорости автобуса.

	A	B	C	D	E	F	G
1	Расстояния между городами Беларуси, км						
2	город	Минск	Брест	Витебск	Гомель	Гродно	Могилев
3	Минск						
4	Брест	349					
5	Витебск	277	625				
6	Гомель	308	535	342			
7	Гродно	268	259	545	576		
8	Могилев	210	524	164	182	478	
9							
10	скорость, км/ч	60					
11	Время в пути, час						
12	город	Минск	Брест	Витебск	Гомель	Гродно	Могилев
13	Минск						
14	Брест	5,82					
15	Витебск	4,62	10,42				
16	Гомель	5,13	8,92	5,70			
17	Гродно	4,47	4,32	9,08	9,60		
18	Могилев	3,50	8,73	2,73	3,03	7,97	

3. Ежедневно уровень радиации в зоне заражения уменьшается на 3 % по отношению к предыдущему дню. Постройте таблицу и исследуйте, через сколько дней уровень радиации упадет до безопасного значения 10 единиц. Начальное значение задавайте от 100 до 1000 единиц.



4. Постройте таблицу, в которой вычисляется:

а) через сколько дней количество атомов радиоактивного изотопа йод-131 уменьшается в 100 раз, если период полураспада (уменьшения количества атомов вдвое) составляет 8,14 дня;

б) во сколько раз уменьшится количество ядер радиоактивного изотопа цезия (период полураспада 30,2 года) через 1, 2, 3, ..., 20 периодов. Сколько пройдет лет?

18.3. ИСПОЛЬЗОВАНИЕ ГРАФИКОВ

Замечательные возможности представления данных в виде разнообразных графиков и диаграмм существенно повышают наглядность исследования законов природы и общества.

Пример 1. Первоначально участок леса содержит 90 тыс. м³ древесины. Ежегодная вырубка составляет $D = 8$ тыс. м³, а естественный при-

Правообладатель Издательский центр БГУ

рост — $P = 5\%$. Построить таблицу и график изменения количества древесины по годам за 20 лет. Исследовать, на каком году объем древесины уменьшится до 20 тыс. м³. При какой максимальной вырубке не наступает уменьшение объема древесины?

Введем начальный объем древесины в ячейку В2, объем ежегодной вырубки — в Е1, а процент прироста — в Е2.

Диапазон ячеек А2:А22 заполним рядом значений лет от 0 до 20 с шагом 1. Значение объема древесины в любой год будем выражать через значение предыдущего года. Для этого в ячейку В3 введем формулу $=B2+B2*E\$2/100-E\1 и распространим ее на весь диапазон В2:В22. Построим график изменения объема древесины (рис. 5.66).

Из таблицы и графика видно, что при заданных начальных условиях объем древесины уменьшится до 20 тыс. м³ на 15-м году, а к 17-му году весь лес будет вырублен.

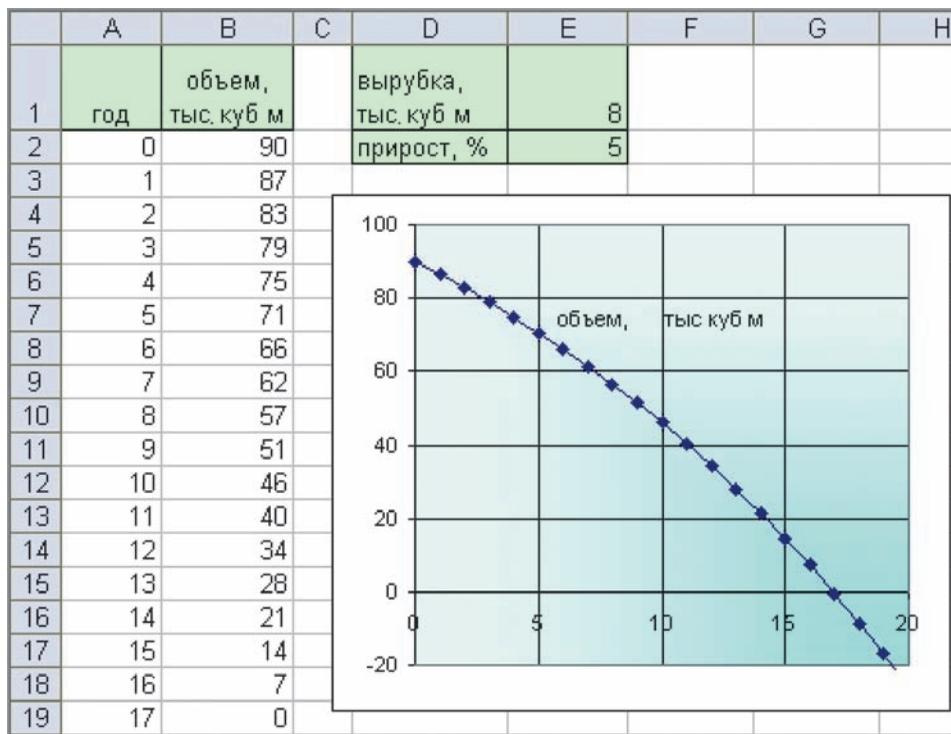


Рис. 5.66

Будем уменьшать объемы вырубки. При значении $D = 4,5$ тыс. м³ вырубка компенсируется естественным приростом и график принимает вид горизонтальной прямой (рис. 5.67).

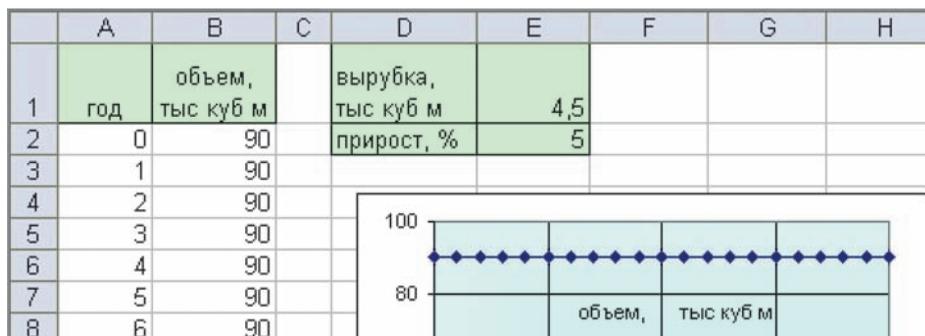


Рис. 5.67

Еще одна из самых распространенных областей применения электронных таблиц — финансовые и экономические расчеты.



Пример 2. Имеется 100 тыс. руб., которые можно поместить в банк. В банке *A* ежегодно начисляют 30 % от первоначального вклада. В банке *B* ежегодно начисляют 20 % от суммы, находящейся на счете. С помощью таблицы и графика исследовать, услугами какого банка и какое время выгоднее пользоваться. Зависит ли это от величины вклада? От начисляемых процентов?

Поместим начальное значение вклада в ячейку *B1*, а начисляемые проценты двух банков — в ячейки *B3* и *C3*. Столбец *A* заполним рядом значений от 0 до 8 лет с шагом 1.

Текущее значение вклада в банке *A* будем выражать через первоначальный вклад и количество лет (простые проценты). Для этого в ячейку *B6* введем формулу $=B\$1+A6*B\$1*B\$3/100$ и распространим ее на диапазон *B6*:*B13*.

Текущее значение вклада в банке *B* будем выражать через предыдущее (сложные проценты). Для этого в ячейку *C6* введем формулу $=C5+C5*C\$3/100$ и распространим ее на диапазон *C6*:*C13*. Построим графики (точечные диаграммы) зависимости величины вкладов от времени (рис. 5.68).

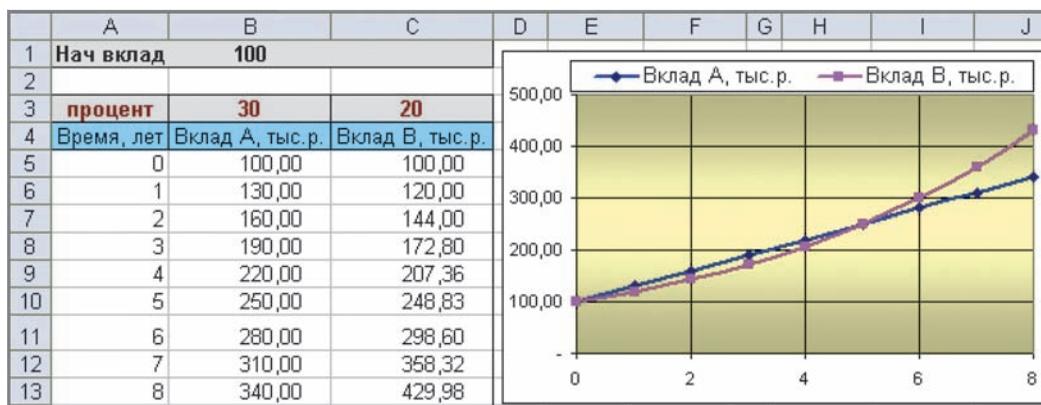


Рис. 5.68

Сравнивая значения вкладов (столбцы В и С и графики), приходим к выводу, что краткосрочные (до 5 лет) вклады выгоднее хранить в первом банке, а долгосрочные — во втором. Так, на 8-м году вклад в банке *В* составит около 430 тыс. руб., т. е. на 90 тыс. руб. превысит вклад в банке *А* (340 тыс. руб.). Равенство текущих вкладов при указанных процентах (30 % годовых в банке *А* и 20 % в банке *В*) наступает на 5-м году и не зависит от первоначального вклада, в чем легко убедиться, изменяя его значение в ячейке В1. Несложно исследовать и зависимости вкладов от начисляемых банками процентов, изменяя их значения в ячейках В3 и С3.

Упражнения

1. Постройте таблицу и график, которые позволяют определять плотность водного раствора серной кислоты по ее массовой доле (рис. 5.69).

2. Постройте таблицу и график, которые позволяют определить:

а) сколько лет нужно хранить S рублей в банке под P % годовых, чтобы утроить вклад. Задаются начальное значение суммы вклада и процент;

б) под какой процент необходимо положить вклад S , чтобы через T лет получить удвоение вклада. Задаются начальное значение суммы вклада и срок T лет;

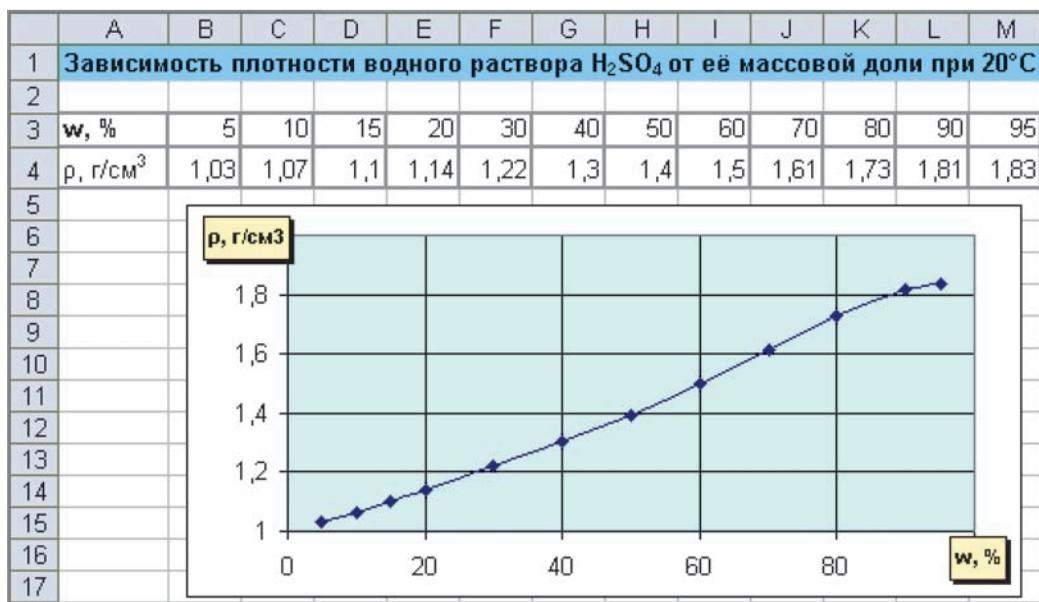


Рис. 5.69

в) какую сумму потребуется выплатить по кредиту через N месяцев при начисляемых ежемесячно процентах P . Задаются величина кредита, количество месяцев N и процент P .

КОМПЬЮТЕРНЫЕ КОММУНИКАЦИИ И ИНТЕРНЕТ

§ 19. РАЗНОВИДНОСТИ ЭЛЕКТРОННЫХ КОММУНИКАЦИЙ

Информация может передаваться с помощью различных электронных средств, например телефона, радио, телевидения. Под электронной коммуникацией понимают общение с помощью электронных средств. На современном этапе развития электронных коммуникаций ключевую роль стали играть компьютерные сети.

Передача и прием информации в компьютерной сети могут происходить различными способами. Различают два режима обмена сообщениями.

В первом случае передача и прием сообщений разделены во времени. В этом режиме осуществляются, например, отправка и получение писем по электронной почте.

Во втором случае общение происходит в режиме реального времени. Общение в этом режиме обеспечивают средства мгновенного обмена сообщениями — программы-месенджеры (от англ. messenger — связной, курьер). Такие программы позволяют обмениваться через компьютерные сети текстовыми, голосовыми и даже видеосообщениями.

Наиболее популярные месенджеры — NetMeeting, Skype, ICQ.

19.1. ИСПОЛЬЗОВАНИЕ ПРОГРАММЫ NETMEETING

Программа NetMeeting позволяет производить обмен информацией между компьютерами в локальных и глобальных сетях, а также организовывать сеансы совместной работы.

Пример 1. Произвести обмен файлами между двумя компьютерами сети.

- На обоих компьютерах запустим программу NetMeeting с помощью меню **Пуск** → **Программы** или ярлыка  на Рабочем столе.
- На компьютере 1 введем IP-адрес вызываемого абонента или имя компьютера в локальной сети, например comp1322 (рис. 6.1).
- Щелкнем мышью по кнопке **Вызов**.

Правообладатель Издательский центр БГУ

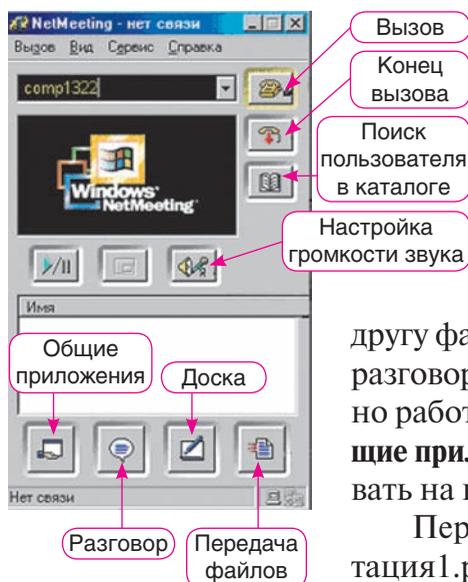


Рис. 6.1

• Вызываемый абонент на компьютере 2 должен подтвердить получение вызова щелчком мыши по кнопке **Принять** (рис. 6.2).

При успешном соединении компьютеров в окне программы появляются их имена (рис. 6.3).

Теперь можно отправлять друг другу файлы (кнопка **Передача файлов** ) , вести разговор (кнопка **Разговор** ) , а также совместно работать с общим приложением (кнопка **Общие приложения** ) , например, совместно рисовать на графической доске (кнопка **Доска** ) .

Перешлем по сети файл, например Презентация1.ppt.

• На компьютере 1 щелкнем левой клавишей мыши по кнопке **Передача файлов**  .
 • В появившемся окне **Передача файлов** (рис. 6.4) выполним цепочку действий: **Файл** → **Добавить файлы** и укажем файл для передачи и адресата.



Рис. 6.3

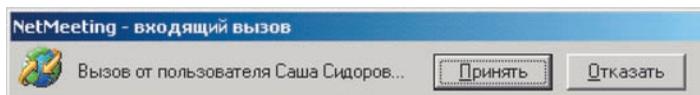


Рис. 6.2

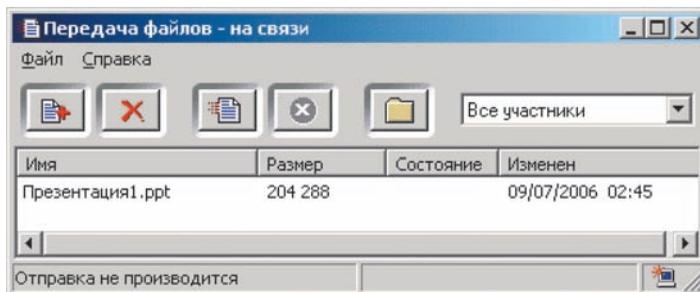


Рис. 6.4

- Для подтверждения отправки выполним действия: **Файл** → **Отправить файл** (или **Отправить все**).
- На компьютере 2 примем полученный файл. Для его просмотра щелкнем мышью по кнопке **Открыть** (рис. 6.5).

Аналогичным способом выполняется обмен текстовыми сообщениями.

Пример 2. Организовать сеанс совместного рисования с помощью программы NetMeeting.

- На компьютерах локальной сети запустим программу NetMeeting.
- Установим связи между подключенными к сети парами рядом стоящих компьютеров: например, с четными номерами (ЧН) и нечетными номерами (НН).

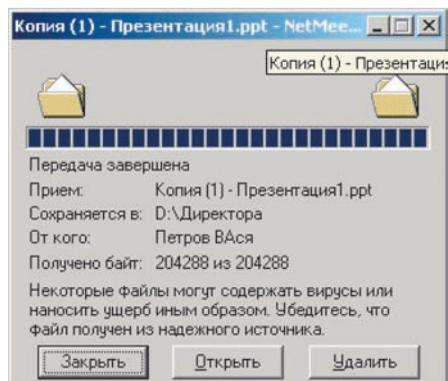


Рис. 6.5

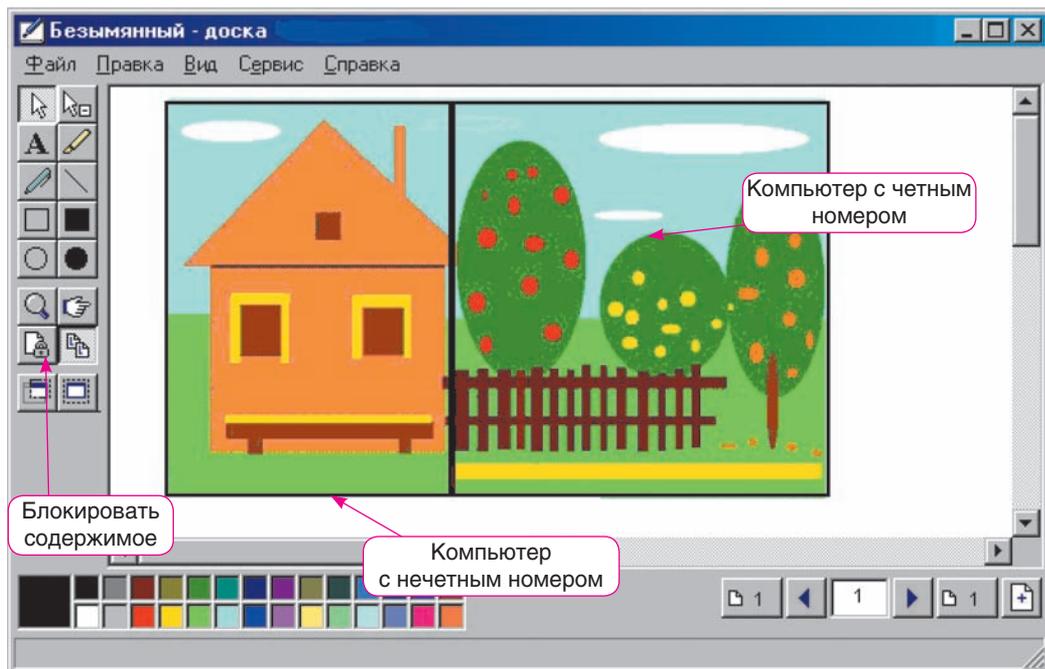


Рис. 6.6

- Щелчком мыши по кнопке **Доска**  программы NetMeeting вызовем графический редактор. По виду и возможностям он подобен редактору Paint. Отличие заключается в том, что во время совместной работы можно блокировать работу соседнего компьютера щелчком мыши по кнопке **Блокировать содержимое**.
- Выполним совместно рис. 6.6. Левая часть рисунка выполняется на компьютерах с нечетными номерами, а правая — с четными.

Пример 3. Организовать сеанс совместной работы в редакторе MS Word.

На всех компьютерах сети откроем редактор MS Word.

- В программе NetMeeting щелкнем мышью по кнопке **Общие приложения** . В окне **Общий доступ** укажем редактор MS Word и щелкнем мышью по кнопке **Общий доступ** (рис. 6.7).
- В редакторе Word наберем строки из поэмы Якуба Коласа «Новая зямля»:

*Мой родны кут, як ты мне мілы,
Забыць цябе не маю сілы!*

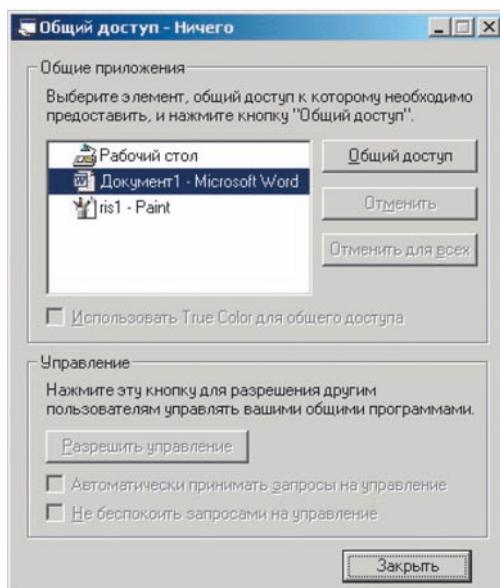


Рис. 6.7

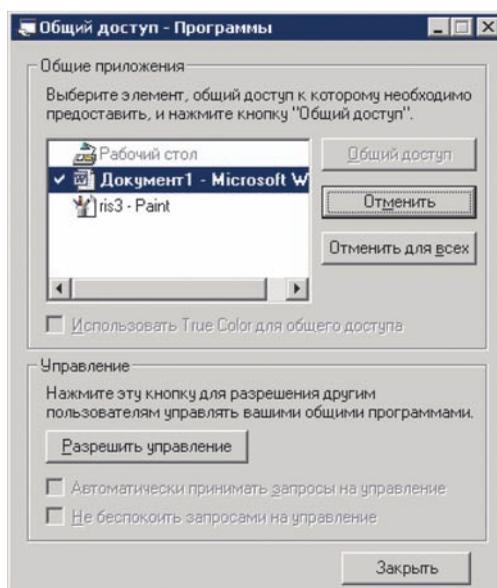


Рис. 6.8

- Щелкнем мышью по кнопке **Разрешить управление** (рис. 6.8). На экране своего компьютера понаблюдаем за работой соседа.
- Выполним цепочку действий из надстройки над окном своего редактора Word: **Управление** → **Запросить управление**.
- На другом компьютере необходимо щелкнуть мышью по кнопке **Принять** (рис. 6.9).

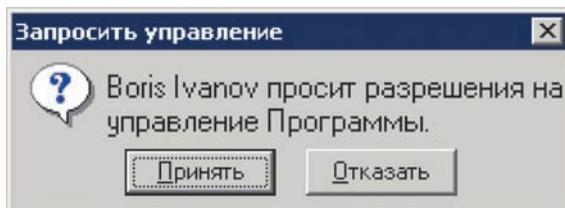


Рис. 6.9

- Для завершения связи нажмем кнопку **Конец вызова** .

19.2. ОБЩЕНИЕ С ПОМОЩЬЮ ПРОГРАММЫ SKYPE

Одной из наиболее популярных программ интернет-общения является Skype. Эта программа позволяет в режиме реального времени вести диалог с помощью текстовых сообщений, пересылать файлы, осуществлять голосовое и видеообщение (интернет-телефония), вести телеконференции.

Для голосового общения в программе Skype пользователю необходимо иметь на своем компьютере средства мультимедиа: звуковую карту, микрофон, колонки или наушники, а для видеообщения — еще и веб-камеру.

После установки программы Skype на компьютере необходимо зарегистрироваться в окне **Создать пользователя**, указав имя, пароль и некоторые сведения о себе (рис. 6.10).

Для добавления контакта с новым абонентом необходимо выполнить в основном окне Skype следующую последовательность действий: **Контакты** → **Новый контакт**. Затем в окне **Добавить контакт** ввести имя, логин или адрес электронной почты нового абонента и нажать кнопку **Поиск** (рис. 6.11).

Зарегистрировать нового пользователя

Полное имя

Выбери логин для Skype

Примечание. Эта информация открыта для всех. Пожалуйста, заполни все поля.

Пароль

Повтори пароль

Эл. почта

Введи адрес эл. почты еще раз

Да, я хочу получать рассылки с новостями и специальными предложениями от Skype

Примечание. Эти сведения будут видны только тебе. Пожалуйста, заполни все поля.

Рис. 6.10

Skype™ - Добавить контакт

Добавить контакт

Добавить контакт Skype
 Найди людей, которые пользуются Skype, и добавь их в свой список контактов. Введи **логин**, **полное имя** или **адрес эл. почты** и нажми 'Поиск'.

Ты также можешь добавлять контакты из MySpaceIM. Введи их имена и нажми "Найти".

Выбери нужного пользователя и нажми «Добавить контакт»:

Полное имя	Логин Skype	Страна/регион	Личные
	marinasolod		<input type="checkbox"/>
Марина Солодышева	marina-solod	Россия, Санкт-Петербург	<input type="checkbox"/>

Рис. 6.11

После этого программа Skype предложит указанному вами абоненту дать согласие на общение.

Щелкнув правой кнопкой мыши по выбранному контакту, можно **Позвонить** абоненту, осуществить **Видеозвонок** или **Начать чат** (рис. 6.12).

Правообладатель Издательский центр БГУ

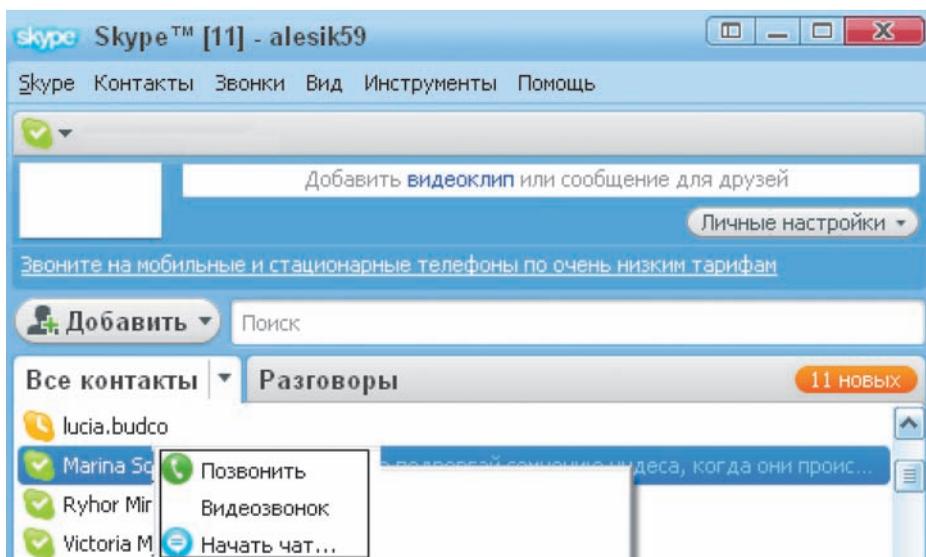


Рис. 6.12

Программа Skype позволяет вести телеконференции. Под **телеконференцией** в сети Интернет понимается одновременное общение нескольких абонентов с помощью голосовой или видеосвязи.

Для проведения телеконференции в программе Skype участники объединяются в группу с помощью последовательности действий **Разговоры** → **Добавить** (рис. 6.13), затем выполняется звонок щелчком мыши по пиктограмме 📞.

В аудиоконференции могут принимать участие до 5—10 участников.

Общение по телефону через Интернет (IP-телефония) позволяет осуществить звонок с компьютера на обычный телефон, в том числе и на мобильный. Такие услуги в основном являются платными. Однако стоимость таких звонков в несколько раз ниже обычных телефонных тарифов.



1. Какими сообщениями позволяют обмениваться программы-месенджеры?
2. Каковы основные возможности программы NetMeeting?
3. Каковы основные возможности программы Skype?

Правообладатель Издательский центр БГУ

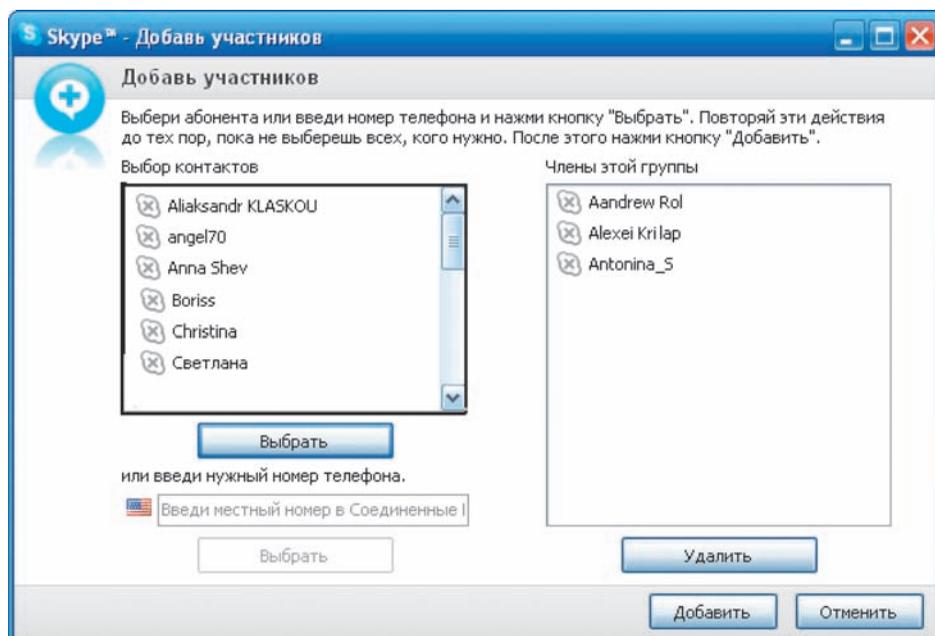


Рис. 6.13

Упражнения

1. С помощью программы NetMeeting:

- а) произведите обмен текстовыми сообщениями и файлами (по указанию учителя) между двумя компьютерами локальной сети;
- б) проведите сеанс совместного рисования разноцветных геометрических фигур (круг, квадрат, треугольник) на одной графической доске. Для каждого компьютера в сети выберите свою фигуру и цвет.



2. С помощью программы Skype проведите телеконференцию с участием компьютеров локальной сети класса (школы).

§ 20. БЛОГИ. ФОРУМЫ. ЧАТЫ. СОЦИАЛЬНЫЕ СЕТИ

Популярными средствами обмена информацией и общения в сети Интернет являются блоги, форумы, чаты.

Блог — это сетевой дневник, или журнал событий. Термин «блог» произошел от сочетания двух английских слов — **Web Log** (Blog).

Правообладатель Издательский центр БГУ

Блог представляет собой сайт или раздел, содержимое которого представлено в виде небольших заметок или записей и оперативно обновляется. Материалы блога создает автор или группа авторов по определенной тематике или интересам.

Отличительная особенность блога состоит в том, что записи в нем могут комментироваться всеми, кто регистрируется и получит соответствующие права доступа.

Блоги обычно размещают на специально отведенных сервисных сайтах. На таких сайтах может зарегистрироваться и начать вести свой блог любой пользователь. Записи в блоге создаются в интерактивном режиме и впоследствии могут редактироваться его владельцем (рис. 6.14).

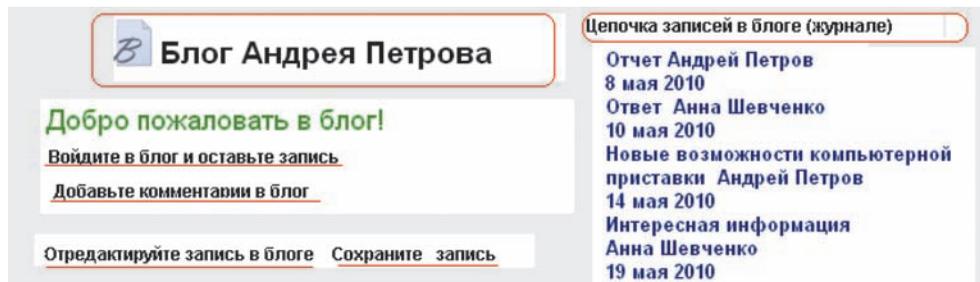


Рис. 6.14

В настоящее время блоги начинают широко использоваться в обучении, например, в качестве журналов для записи и обсуждения проделанной за определенный период работы.

Микроблог, или **твиттер** (от англ. tweet — щебетать, болтать), представляет собой сервис, позволяющий отправлять короткие текстовые заметки (до 140 символов), используя веб-интерфейс, SMS, службы мгновенных сообщений или программы-месенджеры.

Форум (веб-форум) — это специальное пространство сайта или целый сайт для публичного обмена сообщениями между посетителями.

Форумы по своему назначению напоминают блоги. Однако в отличие от них в основу структуры форума положены тематические линии (ветки). При этом первое сообщение задает тематику всей ветки (тему для обсуждения), например, графические возможности редакто-

ров растрового и векторного типов (рис. 6.15). Затем идут комментарии к теме обсуждения, как показано на рис. 6.16.

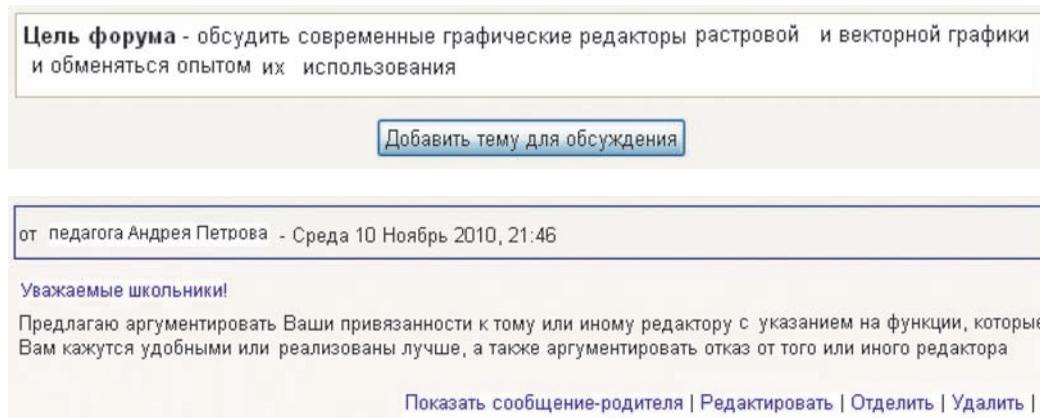


Рис. 6.15

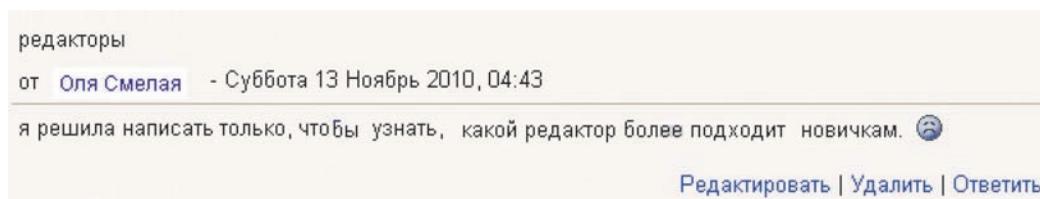


Рис. 6.16

Для участия в форуме требуется регистрация, а при пользовании форумом необходимо выполнять правила, определенные в нем.

Одним из способов общения в сети является ведение диалога (разговора) в *чатах*. Английское слово *chat* переводится как «разговор» или «болтовня». Этот вид сетевого общения очень похож на разговор по телефону, но пользователи не говорят в трубку телефона, а набирают вопросы и ответы на компьютере. В чатах можно выбрать тематический канал для общения с учетом своих интересов. Такие тематические каналы называют «комнатами».

Для общения в чатах пользователю, как правило, необходимо зарегистрироваться, т. е. ввести свой Nickname (идентификатор, напри-

мер имя), Password (пароль) и заполнить, если требуется, регистрационную анкету.

Вести разговоры в чатах можно непосредственно на веб-страницах, а также с помощью программ-месенджеров. Программы ICQ, NetMeeting позволяют участвовать в разговоре сразу несколькими собеседникам.

Пример. С помощью программы NetMeeting установить связь между несколькими компьютерами сети и провести на них сеанс одновременного разговора (чат).

- На всех компьютерах локальной сети запустим программу NetMeeting.
- Установим связи между несколькими компьютерами (по указанию учителя).
- Щелкнем мышью по кнопке **Разговор** . Распахнем окно разговора на весь экран.
- В области ввода сообщения введем текст приветствия выбранному собеседнику и нажмем кнопку **Отправка сообщения**. Продолжим разговор. Будем задавать друг другу различные вопросы и отвечать на них.
- В конце работы завершим сеанс связи.

В настоящее время в сети Интернет активно используются различные *социальные сервисы*, которые дают возможность их пользователям обмениваться информацией и вести совместную деятельность в сети.

К социальным сервисам Интернета относят: социальные сети, сервисы для хранения мультимедийных ресурсов, социальные закладки, сервисы для совместной работы с документами.

Социальные сети представляют собой интерактивные многопользовательские веб-сайты, содержимое которых наполняется самими участниками сети. Социальные сети являются пространством для неформального знакомства и общения людей по интересам. Общение осуществляется обычно в тематических форумах. Наиболее популярные социальные сети посвящены прежде всего поиску друзей — бывших одноклассников и однокурсников, а также учителей.

Сервисы для хранения мультимедийных ресурсов — фотографий, презентаций, видео дают возможность размещать, просматривать и оценивать размещенные ресурсы, а также оставлять свои комментарии.

Социальные закладки — это средства для хранения в Интернете закладок с адресами веб-страниц. В отличие от закладок в браузере пользователя (которые доступны только ему), доступ к социальным закладкам возможен для всех пользователей глобальной сети.

Сервисы для совместной работы с документами различных типов предоставляют многие поисковые системы.

Сервисы для совместной работы с документами дают возможность пользователям:

- иметь совместный доступ к электронным материалам разного характера;
- выполнять задания по совместному редактированию текста (например, написание реферата или плана исследования);
- разрабатывать совместные презентации;
- организовывать совместное решение задач.

Следует отметить, что все средства обмена информацией и общения в компьютерных сетях бурно развиваются и видоизменяются.

К сожалению, культура сетевого общения, особенно в чатах и на форумах, в настоящее время невысока. Это показывает анализ диалогов как на белорусских чатах, так и на многих российских. Появились сайты, пропагандирующие насилие и жестокость, расовую и национальную нетерпимость. Участились случаи финансовых афер.

При общении в сети следует соблюдать меры безопасности и правила сетевого этикета.

- Помните, что вы чаще всего общаетесь с анонимными собеседниками, среди которых могут оказаться непорядочные люди.
- Не торопитесь давать новому собеседнику в чате, блоге, на форуме свой телефон или домашний адрес.
- Если, с вашей точки зрения, собеседник навязывается в общение, то постарайтесь заблокировать его доступ к вам.
- Общение в некоторых тематических каналах может запрещаться правовыми нормами и законодательством государства, а также моральными и этическими правилами поведения религиозной или этнической группы, к которой вы принадлежите.

- При общении в сети будьте предельно корректными и вежливыми. При обсуждении всякая критика или несогласие с собеседником должны быть аргументированы.



1. Что понимается под блогом? Для чего используются блоги?
2. Что такое веб-форум?
3. Что такое чат? Для чего используют чаты?
4. Что представляют собой социальные сети?
5. Каковы основные правила сетевого общения?

Приложение 1

Кодовая таблица символов ASCII (32—127)

Код	Символ	Код	Символ	Код	Символ	Код	Символ
32		56	8	80	P	104	h
33	!	57	9	81	Q	105	i
34	"	58	:	82	R	106	j
35	#	59	;	83	S	107	k
36	\$	60	<	84	T	108	l
37	%	61	=	85	U	109	m
38	&	62	>	86	V	110	n
39	'	63	?	87	W	111	o
40	(64	@	88	X	112	p
41)	65	A	89	Y	113	q
42	*	66	B	90	Z	114	r
43	+	67	C	91	[115	s
44	,	68	D	92	\	116	t
45	-	69	E	93]	117	u
46	.	70	F	94	^	118	v
47	/	71	G	95	_	119	w
48	0	72	H	96	`	120	x
49	1	73	I	97	a	121	y
50	2	74	J	98	b	122	z
51	3	75	K	99	c	123	{
52	4	76	L	100	d	124	
53	5	77	M	101	e	125	}
54	6	78	N	102	f	126	~
55	7	79	O	103	g	127	□

Приложение 2

Кодовая таблица символов CP-1251 (128—255)

Код	Символ	Код	Символ	Код	Символ	Код	Символ
128	Ђ	160		192	А	224	а
129	Ѓ	161	Ў	193	Б	225	б
130	,	162	ў	194	В	226	в
131	ѓ	163	Ј	195	Г	227	г
132	„	164	Ѡ	196	Д	228	д
133	...	165	Г	197	Е	229	е
134	†	166	‡	198	Ж	230	ж
135	‡	167	§	199	З	231	з
136	€	168	Ё	200	И	232	и
137	‰	169	©	201	Й	233	й
138	Љ	170	Є	202	К	234	к
139	<	171	«	203	Л	235	л
140	Њ	172	¬	204	М	236	м
141	Ќ	173		205	Н	237	н
142	Ђ	174	®	206	О	238	о
143	Џ	175	İ	207	П	239	п
144	ђ	176	°	208	Р	240	р
145	‘	177	±	209	С	241	с
146	’	178	І	210	Т	242	т
147	“	179	і	211	У	243	у
148	”	180	г	212	Ф	244	ф
149	•	181	μ	213	Х	245	х
150	—	182	¶	214	Ц	246	ц

Окончание прил. 2

Код	Символ	Код	Символ	Код	Символ	Код	Символ
151	—	183	·	215	Ч	247	ч
152		184	ë	216	Ш	248	ш
153	™	185	№	217	Щ	249	щ
154	ль	186	€	218	Ъ	250	ъ
155	>	187	»	219	Ы	251	ы
156	нь	188	j	220	Ь	252	ь
157	ќ	189	S	221	Э	253	э
158	ћ	190	s	222	Ю	254	ю
159	ц	191	ï	223	Я	255	я

Стандартные процедуры и функции

Имя и параметры	Типы параметров	Тип возвращаемого значения	Действие
Процедура Read(a,b,...)	Простой тип или тип string		Вводит с клавиатуры значения переменных a, b, ...
Процедура Write(a,b,...)	Простой тип, тип string или указатели		Выводит значения a, b, ... в окно вывода
Процедура Readln(a,b,...)	Простой тип или тип string		Вводит с клавиатуры значения переменных a, b, ..., после чего пропускает все оставшиеся символы в текущей строке ввода. Если параметры процедуры не указаны, то осуществляет паузу в выполнении программы до нажатия клавиши Enter

Окончание прил. 3

Имя и параметры	Типы параметров	Тип возвра- щаемого значения	Действие
Процедура Writeln(a,b,...)	Простой тип, тип string или указатели		Выводит зна- чения a, b, ... в окно вывода и осуществляет переход на сле- дующую строку. Если параметры процедуры не указаны, то вы- полняет переход на следующую строку
Функция Ord(s)	Порядковый тип	integer	Возвращает код символа
Функция Chr(n)	Тип integer	char	Возвращает символ с кодом n
Функция Pred(s)	Порядковый тип	Совпадает с типом параметра	Возвращает предыдущий символ
Функция Succ(s)	Порядковый тип	Совпадает с типом параметра	Возвращает сле- дующий символ

Приложение 4

Стандартные процедуры и функции для работы со строками

Имя и параметры	Типы параметров	Тип возвращаемого значения	Действие
Функция Length(<i>s</i>)	<i>s</i> — string	integer	Возвращает длину строки <i>s</i>
Функция Copy(<i>s</i> , <i>p</i> , <i>n</i>)	<i>s</i> — string; <i>p</i> и <i>n</i> — integer	string	Возвращает подстроку строки <i>s</i> длины <i>n</i> , начиная с позиции <i>p</i>
Процедура Delete(<i>s</i> , <i>p</i> , <i>n</i>)	<i>s</i> — string; <i>p</i> и <i>n</i> — integer		Удаляет в строке <i>s</i> <i>n</i> символов, начиная с позиции <i>p</i>
Процедура Insert(<i>subs</i> , <i>s</i> , <i>p</i>)	<i>s</i> , <i>subs</i> — string; <i>p</i> — integer		Вставляет подстроку <i>subs</i> в строку <i>s</i> , начиная с позиции <i>p</i>
Функция Pos(<i>subs</i> , <i>s</i>)	<i>s</i> , <i>subs</i> — string	integer	Возвращает первую позицию подстроки <i>subs</i> в строке <i>s</i> (или 0, если подстрока не найдена)
Процедура SetLength(<i>s</i> , <i>n</i>)	<i>s</i> — string; <i>n</i> — integer		Устанавливает длину строки <i>s</i> равной <i>n</i>

Продолжение прил. 4

Имя и параметры	Типы параметров	Тип возвра- щаемого значения	Действие
Процедура Str(x,s) Str(x:n,s) Str(x:n:m,s)	s — string; x — integer, real; n, m — integer		Преобразовывает число x в строку (во втором и третьем слу- чаях — согласно формату вывода, устанавливаемо- му n и m)
Процедура Val(s,v,code)	s — string; v — integer, real; code — integer		Преобразовывает строку s к чис- ловому представ- лению v. Если преобразование возможно, то в переменную code возвра- щается 0, если не- возможно, то не- нулевое значение
Функция Concat(s1, ...,sn)	s1, ..., sn — string	string	Возвращает стро- ку, являющуюся результатом соединения строк s1, ..., sn. Ре- зультат тот же, что у выражения s1+s2+...+sn

Продолжение прил. 4

Имя и параметры	Типы параметров	Тип возвра- щаемого значения	Действие
Функция IntToStr(i)	i — integer	string	Преобразовыва- ет целое число к строке
Функция StrToInt(s)	s — string	integer	Преобразовывает строку в целое число. Если пре- образование невозможно, то возникает ошиб- ка времени вы- полнения
Функция FloatToStr(r)	r — real	string	Преобразовыва- ет вещественное число к строке
Функция StrToFloat(s)	s — string	real	Преобразовывает строку в веще- ственное число. Если преобразо- вание невозмож- но, то возникает ошибка времени выполнения
Функция UpCase(c)	c — char	char	Возвращает сим- вол c, преобразо- ванный к верхне- му регистру

Окончание прил. 4

Имя и параметры	Типы параметров	Тип возвращаемого значения	Действие
Функция LowerCase(c)	c — char	char	Возвращает символ c, преобразованный к нижнему регистру
Функция UpperCase(s)	s — string	string	Возвращает строку s, преобразованную к верхнему регистру
Функция LowerCase(s)	s — string	string	Возвращает строку s, преобразованную к нижнему регистру
Функция Trim(s)	s — string	string	Возвращает копию строки s с удаленными лидирующими и заключительными пробелами
Функция TrimLeft(s)	s — string	string	Возвращает копию строки s с удаленными лидирующими пробелами
Функция TrimRight(s)	s — string	string	Возвращает копию строки s с удаленными заключительными пробелами

СОДЕРЖАНИЕ

От авторов.....	3
ГЛАВА 1. Хранение информации	4
§ 1. Архивация данных	4
ГЛАВА 2. Цифровые устройства для обработки информации	12
§ 2. Совместное использование цифровых устройств и компьютера	12
§ 3. Работа со сканером	17
ГЛАВА 3. Основы алгоритмизации и программирования	22
§ 4. Символьные и строковые величины. Операции над символьными и строковыми величинами	22
4.1. Символьный тип данных	22
4.2. Процедуры и функции обработки символьных переменных	28
4.3. Строковый тип данных	31
§ 5. Процедуры и функции обработки строковых переменных	36
§ 6. Выполнение практических заданий с использованием символьных и строковых величин	41
ГЛАВА 4. Информационные модели	50
§ 7. Понятие и назначение информационной модели	50
§ 8. Структурирование информации с использованием информационных моделей	54
§ 9. Компьютерное моделирование	57
ГЛАВА 5. Обработка информации в электронных таблицах	61
§ 10. Понятие электронной таблицы.....	61
10.1. Назначение табличного процессора	61
10.2. Структура таблицы: ячейки, столбцы, строки	62
10.3. Типы данных	65
§ 11. Ввод и редактирование данных. Сохранение рабочей книги	67

§ 12. Работа с диапазонами ячеек	77
§ 13. Форматирование электронной таблицы	82
§ 14. Использование формул. Относительные и абсолютные ссылки	92
§ 15. Использование функций	96
§ 16. Построение диаграмм	103
§ 17. Сортировка данных в таблице	111
§ 18. Выполнение практических заданий из различных предметных областей	115
18.1. Решение расчетных задач	115
18.2. Исследование зависимостей	119
18.3. Использование графиков	122
ГЛАВА 6. Компьютерные коммуникации и Интернет	127
§ 19. Разновидности электронных коммуникаций	127
19.1. Использование программы NetMeeting	127
19.2. Общение с помощью программы Skype	131
§ 20. Блоги. Форумы. Чаты. Социальные сети	134
Приложения	140

(Название и номер школы)

Учебный год	Имя и фамилия ученика	Состояние учебного пособия при получении	Оценка ученику за пользование учебным пособием
20 /20			
20 /20			
20 /20			
20 /20			
20 /20			
20 /20			

- Заборовский, Г. А.**
3-12 Информатика : учеб. пособие для 10-го кл. учреждений общ. сред. образования с рус. яз. обучения / Г. А. Заборовский, А. Е. Пупцев. — Минск : Изд. центр БГУ, 2011. — 151 с. : ил.
ISBN 978-985-476-907-3.

УДК 004(075.3=161.1)

ББК 32.81я721

Учебное издание

Заборовский Георгий Александрович
Пупцев Александр Евгеньевич

ИНФОРМАТИКА

**Учебное пособие для 10 класса
учреждений общего среднего образования
с русским языком обучения**

Главный редактор *Т. Е. Янчук*
Редактор *Г. А. Ребенкова*
Технический редактор *Е. В. Романчик*
Компьютерная верстка *Н. Е. Батюковой*
Корректоры *В. М. Иванов, Н. Б. Кучмель, Е. В. Семенчукова*

Подписано в печать 24.06.2011. Формат 70×90 ¹/₁₆. Бумага офсетная № 1. Печать офсетная.
Усл. печ. л. 11,11. Уч.-изд. л. 7,43. Тираж 109 840 экз. Заказ

Республиканское унитарное предприятие
«Издательский центр Белорусского государственного университета».
ЛИ № 02330/0494361 от 16.03.2009. Ул. Красноармейская, 6, 220030, Минск.

Производственное республиканское унитарное предприятие
«Минская фабрика цветной печати».
ЛП № 02330/0494156 от 03.04.2009. Ул. Корженевского, 20, 220024, Минск.

Правообладатель Издательский центр БГУ