



УЧЕБНИКИ КУЗГТУ

Л.С. Таганов, А.Г. Пимонов

УЧЕБНОЕ ПОСОБИЕ

ИНФОРМАТИКА

РЕКОМЕНДОВАНО

**СИБИРСКИМ РЕГИОНАЛЬНЫМ УЧЕБНО-МЕТОДИЧЕСКИМ ЦЕНТРОМ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
ДЛЯ МЕЖВУЗОВСКОГО ИСПОЛЬЗОВАНИЯ В КАЧЕСТВЕ УЧЕБНОГО ПОСОБИЯ
ДЛЯ СТУДЕНТОВ ТЕХНИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ И НАПРАВЛЕНИЙ**

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«КУЗБАССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Л. С. ТАГАНОВ, А. Г. ПИМОНОВ

ИНФОРМАТИКА

Учебное пособие

Рекомендовано Сибирским региональным учебно-методическим центром высшего профессионального образования для межвузовского использования в качестве учебного пособия для студентов технических специальностей и направлений

Кемерово 2010

УДК 004

РЕЦЕНЗЕНТЫ

К. Е. Афанасьев, доктор физико-математических наук, профессор, проректор по информационным технологиям и открытому образованию, заведующий кафедрой ЮНЕСКО по новым информационным технологиям Государственного образовательного учреждения высшего профессионального образования «Кемеровский государственный университет»

Кафедра технологии автоматизированной обработки информации института информационных и библиотечных технологий Государственного образовательного учреждения высшего профессионального образования «Кемеровский государственный университет культуры и искусств» (заведующий кафедрой *Н. И. Колкова*, кандидат педагогических наук, доцент)

Таганов, Л. С.

Информатика : учеб. пособие / Л. С. Таганов, А. Г. Пимонов ; Кузбас. гос. техн. ун-т. – Кемерово, 2010. – 330 с.
ISBN 978-5-89070-770-3

В учебном пособии изложены в систематизированном виде теоретические основы, обеспечивающие единую методическую базу для изучения информатики, и представлены все необходимые материалы для освоения дисциплины в объеме учебного курса вуза в соответствии с государственным образовательным стандартом.

Пособие предназначено для студентов технических специальностей и направлений, а также может быть рекомендовано для преподавателей и студентов других специальностей.

Печатается по решению редакционно-издательского совета Кузбасского государственного технического университета.

УДК 004

© Кузбасский государственный
технический университет, 2010

© Таганов Л. С., Пимонов А. Г., 2010

ISBN 978-5-89070-770-3

© Дизайн обложки. Юдинкова А. В., 2010

Предисловие

Настоящее пособие предназначено для студентов, изучающих дисциплину «Информатика». Пособие включает предисловие, введение, семь глав и заключение. В первой главе рассмотрены понятия данных и информации, изложены основные положения алгебры логики, основные понятия позиционных систем счисления, рассмотрены способы кодирования и структуры данных, приведены единицы представления, измерения, хранения и передачи данных. Во второй главе изложена информация о технических средствах реализации информационных процессов. Третья глава содержит информацию о программном обеспечении информационных процессов, уровнях программного обеспечения и основные сведения об операционных системах. В ней дана характеристика программы MS Excel и структуры электронных таблиц. Изложены сведения о текстовых процессорах семейства MS Word, а также обсуждаются некоторые технологии работы с документами. Средства создания и использования презентаций, Средства компьютерной графики. Основные понятия теории баз данных и технология работы с базами данных MS Access в режиме конструктора. В четвертой главе приведена информация о моделях решения функциональных и вычислительных задач. В пятой главе рассмотрены основные понятия алгоритмизации и базовые структуры алгоритмов. В этой главе содержится информация о программном обеспечении и технологиях программирования, приведен обзор языков программирования, изложена структура языка программирования Visual Basic for Applications (VBA) с множеством примеров, иллюстрирующих применение его различных возможностей. В шестой главе представлены общие понятия о локальных и глобальных компьютерных сетях. В седьмой главе рассмотрены угрозы информационной безопасности и методы защиты информации.

Несмотря на наличие на российском рынке учебной литературы по дисциплине «Информатика», издания, освещающие в полном объеме все разделы Государственного образовательного стандарта по информатике, отсутствуют. Это определяет актуальность и практическую значимость учебного пособия.

Введение

Информатика – это прикладная техническая наука, систематизирующая приемы создания, хранения, воспроизведения, обработки и передачи данных (информации) средствами вычислительной техники, а также принципы функционирования этих средств и методы управления ими.

В рамках дисциплины «Информатика» изучаются следующие понятия:

- аппаратное обеспечение средств вычислительной техники;
- программное обеспечение средств вычислительной техники;
- средства взаимодействия аппаратного и программного обеспечения;
- средства взаимодействия человека с аппаратным и программным обеспечением.

Основной задачей информатики является систематизация приемов и методов работы с аппаратными и программными средствами вычислительной техники.

Цель систематизации состоит в выделении, внедрении и развитии передовых, наиболее эффективных технологий, в автоматизации этапов работы с данными, а также в методическом обеспечении новых технологических исследований.

В составе основной задачи информатики выделяются следующие направления для практических приложений:

- приемы и методы построения систем, предназначенных для автоматической обработки данных;
- приемы и методы управления аппаратным и программным обеспечением;
- приемы, методы и средства разработки компьютерных программ;
- компьютерная графика;
- приемы и методы преобразования структур данных;
- обобщение приемов, разработка методов и средств защиты информации;
- функционирование программно-аппаратных средств без участия человека;
- обеспечение совместимости между аппаратными и программными средствами, а также между форматами

представления данных (информации), относящихся к различным типам вычислительных систем.

Основопологающим понятием информатики является информационная модель.

Основной метод, используемый в информатике, это моделирование информационных процессов с помощью компьютера.

Предмет «Информатика» – это учебная дисциплина, в рамках которой изучаются технологии создания, хранения, воспроизведения и обработки данных (информации) средствами вычислительной техники, а также принципы функционирования этих средств и методы управления ими.

Изучение дисциплины «Информатика» с помощью предложенного учебного пособия позволит студентам, обладающим непреодолимой тягой к знаниям, способным проявить минимальную любознательность и недюжинную усидчивость, решить следующие задачи:

- сформировать общие представления о возможностях использования средств вычислительной техники;
- ознакомиться с основами современных информационных технологий (сбора, обработки, хранения и передачи информации) и тенденциями их развития;
- научиться использовать современные информационные технологии в профессиональной деятельности и в выполнении анализа полученных результатов,
- развить навыки алгоритмического мышления;
- овладеть приемами работы с современными типовыми пакетами прикладных программ (MS Excel, MS Word и MS Access), обеспечивающих широкие возможности обработки информации.

Глава 1. Понятие информации, общая характеристика процессов сбора, передачи, обработки и накопления информации

1.1. Понятие данных и информации

Все существующие объекты, процессы или явления обладают определенными свойствами, которые можно регистрировать теми или иными средствами и способами. Зарегистрированные свойства характеризуют объекты, процессы или явления. Следовательно, зарегистрированные свойства – это данные об объектах, процессах или явлениях.

Итак, *данные* – это факты (зарегистрированные сигналы), характеризующие объекты, процессы, явления.

Обработка данных адекватными им методами создает новый продукт – информацию. Таким образом, информация возникает и существует в момент взаимодействия объективных данных и субъективных методов.

Итак, *информация* – это продукт взаимодействия данных и адекватных им методов или мера устранения неопределенности в отношении, интересующего нас события.

1.2. Общая характеристика процессов сбора, передачи, обработки и накопления информации

В ходе информационного процесса данные преобразуются из одного вида в другой с помощью методов. Обработка данных включает в себя множество различных операций. В структуре возможных операций с данными и информацией можно выделить следующие операции:

- сбор данных и информации – накопление информации с целью обеспечения достаточной полноты для принятия решения;
- форматизация данных и информации – приведение данных, поступающих из разных источников, к одинаковой форме, чтобы сделать их сопоставимыми между собой, то есть повысить их уровень доступности;
- фильтрация данных и информации – отсеивание тех данных, в которых нет необходимости для принятия решения; при

этом должны возрастать достоверность и адекватность информации;

- сортировка данных и информации – упорядочение данных по определенному признаку с целью удобства их использования; при этом должна повышаться доступность информации;
- архивация данных и информации – организация хранения данных в удобной и легкодоступной форме; служит для снижения экономических затрат по хранению данных и повышает общую надежность информационного процесса в целом;
- преобразование данных и информации – перевод данных из одной формы в другую или из одной структуры в другую;
- защита данных (информации) – комплекс мер, направленных на предотвращение утраты, воспроизведения и модификации данных;
- транспортировка данных и информации – прием и передача данных между удаленными участниками информационного процесса.

1.3. Виды и типы данных

Данные могут быть представлены следующими видами:

- числами;
- текстом;
- мультимедиа (графическими объектами, звуковыми сигналами, цветными изображениями).

Виды данных подразделяются на следующие типы:

а) числовой вид:

- байтовый тип;
- целые числа простой точности;
- длинные целые числа;
- вещественные числа простой точности;
- вещественные числа двойной точности;

б) текстовый вид:

- строковый тип;
- текстовый тип (в базах данных);

в) мультимедийный вид самостоятельных типов не имеет, однако для представления графических объектов может использоваться тип объектов, а свойства этого вида задаются целыми числами.

Дата и время могут представляться в виде чисел, однако к числовому виду они не относятся. Для них применяется тип даты и времени.

Наряду с названными типами, относящимися к тем или иным видам, существуют также:

- логический тип;
- тип объектов.

Кроме того, в некоторых языках программирования предусмотрена возможность создания пользовательского типа, а также имеется тип `Variant` (например, в объектно-ориентированном языке `Visual Basic for Applications`), совместимый со всеми другими типами.

1.4. Алгебра логики

Алгебра логики (математическая логика или булева алгебра, или алгебра высказываний) используется в информатике для того, чтобы можно было определять истинность или ложность составных высказываний, не вникая в их содержание.

Каждое составное высказывание можно выразить в виде формулы (логического выражения), в которую войдут логические переменные, обозначающие высказывания, и знаки логических операций, обозначающие логические функции. Для записи составных высказываний в виде логических выражений на формальном языке (языке алгебры логики) в составном высказывании нужно выделить простые высказывания и логические связи между ними. Истинность или ложность составных высказываний можно определять чисто формально, руководствуясь законами алгебры высказываний, не обращаясь к смысловому содержанию высказываний.

В алгебре логики (высказываний) суждениям (простым высказываниям) ставятся в соответствие переменные (заглавные буквы латинского алфавита), которые могут принимать только два значения "истина (`true`)" (1) и "ложь (`false`)" (0).

Пример: А – " $2 \times 2 = 4$ " – истина (1) А = 1
В – " $2 \times 2 = 5$ " – ложь (0) В = 0

Простые высказывания называются *переменными*, а сложные – *логическими функциями*.

1.4.1. Основные логические операции

Логическое умножение (конъюнкция) – объединение 2-х (или нескольких высказываний) в одно с помощью союза "**И**". Конъюнкция истинна тогда и только тогда, когда истинны входящие в нее простые высказывания. Обозначения: **&**, **^**, *****, **and**.

Например, $F = A \& B$, $F = A \wedge B$, $F = A * B$, $F = A \text{ and } B$. Ниже приведена таблица истинности для операции логического умножения.

| A | B | F = A and B |
|----------|----------|--------------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Логическое сложение (дизъюнкция) – объединение 2-х (или нескольких) высказываний в одно с помощью союза "**ИЛИ**". Дизъюнкция истинна тогда, когда истинно хотя бы одно из входящих в выражение простых высказываний. Обозначения: **v**, **+**, **or**.

Например, $F = A \vee B$, $F = A + B$, $F = A \text{ or } B$. Ниже приведена таблица истинности для операции логического сложения.

| A | B | F = A or B |
|----------|----------|-------------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Логическое отрицание (инверсия) – присоединение частицы "**НЕ**" к высказыванию. Инверсия делает истинным ложное высказывание, ложным – истинное высказывание. Обозначения: **¬**, **not**.

Например, $\neg F = A$, $F = \neg A$, $F = \text{not } A$. Ниже приведена таблица истинности для операции логического отрицания.

| A | F = not A |
|----------|------------------|
| 0 | 1 |
| 1 | 0 |

Обычный приоритет выполнения логических операций:
1) инверсия, 2) конъюнкция, 3) дизъюнкция.

Значение логической функции для разных сочетаний значений входных переменных или наборов входных переменных обычно задается *таблицей истинности*. Таблицы истинности для всех логических операций приведены выше.

1.4.2. Свойства операций

Логические операции по свойствам во многом аналогичны привычным математическим, но имеют и свою специфику.

Коммутативность – «от перестановки слагаемых сумма не изменяется»

$$A \text{ and } B = B \text{ and } A$$

$$A \text{ or } B = B \text{ or } A$$

Идемпоентность

$$X \text{ and } X = X$$

$$X \text{ or } X = X$$

Ассоциативность – порядок выполнения операций не важен

$$(A \text{ and } B) \text{ and } C = A \text{ and } (B \text{ and } C)$$

$$(A \text{ or } B) \text{ or } C = A \text{ or } (B \text{ or } C)$$

Дистрибутивность – раскрытие скобок

$$C \text{ or } (A \text{ and } B) = (C \text{ or } A) \text{ and } (C \text{ or } B)$$

$$C \text{ and } (A \text{ or } B) = (C \text{ and } A) \text{ or } (C \text{ and } B)$$

Законы де Моргана (ударение на «о»)

$$\text{Not } (A \text{ and } B) = (\text{Not } A) \text{ or } (\text{Not } B)$$

$$\text{Not } (A \text{ or } B) = (\text{Not } A) \text{ and } (\text{Not } B)$$

Законы поглощения

$$A \text{ and } (A \text{ or } B) = A$$

$$A \text{ or } (A \text{ and } B) = A$$

Другие полезные закономерности, в которых фигурируют константы `true` и `false` выглядят следующим образом:

$$A \text{ and } \text{true} = A$$

$$A \text{ or } \text{true} = \text{true}$$

$$A \text{ and } \text{false} = \text{false}$$

$$A \text{ or } \text{false} = A$$

$$A \text{ and } (\text{not } A) = \text{false}$$

$$A \text{ or } (\text{not } A) = \text{true}$$

Пример: Выполнить некоторое действие с файлом, «если дата его создания T меньше времени L , а если время T больше L , то требуется выполнение дополнительного условия P ». Это можно записать так:

$$T < L \text{ or } (T > L \text{ and } P)$$

Используем дистрибутивность – раскрываем скобки:

$$(T < L \text{ or } T > L) \text{ and } (T < L \text{ or } P)$$

Первая скобка всегда «истина», а «истина и что-то – равно что-то».

Получаем выражение:

$$T < L \text{ or } P$$

Даже в таком простом выражении нам удалось вдвое уменьшить количество операций.

Строго говоря, в алгебре логики рассматриваются не только операции «И», «ИЛИ» и «НЕ». Есть и другие операции, не обладающие всеми перечисленными свойствами. В прикладном программировании же используются преимущественно только эти три операции, а любые другие можно выразить с их помощью.

Пример: «ИСКЛЮЧАЮЩЕЕ ИЛИ (Xor)» можно выразить так:

$$A \text{ Xor } B = (A \text{ or } B) \text{ and } (\text{not } A \text{ or } \text{not } B)$$

или

$$A \text{ Xor } B = (A \text{ and } \text{not } B) \text{ or } (B \text{ and } \text{not } A)$$

1.4.3. Равносильные логические выражения

Логические выражения, у которых таблицы истинности совпадают, называются равносильными (эквивалентными). Обозначение – знак "=".

Пример: Докажем, что $\neg A \& \neg B = \neg(A \vee B)$.

| A | B | $\neg A$ | $\neg B$ | $\neg A \& \neg B$ | $A \vee B$ | $\neg(A \vee B)$ |
|---|---|----------|----------|--------------------|------------|------------------|
| 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 |

Таблицы истинности совпадают, следовательно, логические выражения равносильны: $\neg A \& \neg B = \neg(A \vee B)$.

Импликация и эквиваленция. В обыденной и научной речи кроме базовых логических связок «И», «ИЛИ», «НЕ», используются и

некоторые другие: «ЕСЛИ..., ТО...», «ТОГДА... И ТОЛЬКО ТОГДА, КОГДА...» и др. Некоторые из них имеют свое название и свой символ и им соответствуют определенные логические функции.

Логическое следование (импликация) образуется соединением двух высказываний в одно с помощью оборота речи «ЕСЛИ..., ТО...». Логическая операция импликации «ЕСЛИ А, ТО В», обозначается $A \textcircled{R} B$. Таблица истинности логической функции импликация приведена ниже.

| A | B | A \textcircled{R} B |
|----------|----------|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Составное высказывание, образованное с помощью операции логического следования (импликации), ложно тогда и только тогда, когда из истинной предпосылки (первого высказывания) следует ложный вывод (второе высказывание).

Например:

1) высказывание «Если число делится на 10, то оно делится на 5» истинно, так как истинны и первое высказывание (предпосылка), и второе высказывание (вывод);

2) высказывание «Если число делится на 10, то оно делится на 3» ложно, так как из истинной предпосылки делается ложный вывод.

Однако операция логического следования несколько отличается от обычного понимания слова «следует». Если первое высказывание (предпосылка) ложно, то вне зависимости от истинности или ложности второго высказывания (вывода) составное высказывание истинно. Это можно понимать таким образом, что из неверной предпосылки может следовать что угодно.

В алгебре высказываний все логические функции могут быть сведены путем логических преобразований к трем базовым: логическому умножению, логическому сложению и логическому отрицанию.

Докажем с помощью сравнения таблиц истинности, что операция импликации $A \textcircled{R} B$ равносильна логическому выражению

$\neg A \vee B$. Таблица истинности логических выражений $\neg A \vee B$ и $A \oplus B$ приведена ниже.

| A | B | $\neg A$ | $\neg A \vee B$ | $A \oplus B$ |
|----------|----------|----------------------------|-----------------------------------|--------------------------------|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |

Из вышеприведенной таблицы видно, что $\neg A \vee B = A \oplus B$, что и требовалось доказать.

Логическое равенство (эквивалентность) образуется соединением двух высказываний в одно с помощью оборота речи «... **ТОГДА И ТОЛЬКО ТОГДА, КОГДА...**». Логическая операция эквивалентности «**A ЭКВИВАЛЕНТНО B**» обозначается $A \sim B$ и выражается с помощью логической функции, которая задается соответствующей таблицей истинности.

| A | B | $A \sim B$ |
|----------|----------|------------------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Составное высказывание, образованное с помощью логической операции эквивалентности, истинно тогда и только тогда, когда оба высказывания одновременно либо ложны, либо истинны.

Рассмотрим, например, два высказывания $A =$ «Компьютер может производить вычисления» и $B =$ «Компьютер включен».

Составное высказывание, полученное с помощью операции эквивалентности, истинно, когда оба высказывания либо истинны, либо ложны.

1) «Компьютер может производить вычисления тогда и только тогда, когда компьютер включен».

2) «Компьютер не может производить вычисления тогда и только тогда, когда компьютер не включен».

Составное высказывание, полученное с помощью операции эквивалентности, ложно, когда одно высказывание истинно, а другое – ложно.

1) «Компьютер может производить вычисления тогда и только тогда, когда компьютер не включен».

2) «Компьютер не может производить вычисления тогда и только тогда, когда компьютер включен».

1.4.4. Основные законы логики

Законы логики отражают наиболее важные закономерности логического мышления. В алгебре высказываний законы логики записываются в виде формул, которые позволяют проводить эквивалентные преобразования логических выражений в соответствии с этими законами.

Закон тождества. Всякое высказывание тождественно самому себе: $A = A$.

Закон непротиворечия. Высказывание не может быть одновременно истинным и ложным. Если высказывание A истинно, то его отрицание $\neg A$ должно быть ложным. Следовательно, логическое произведение высказывания и его отрицания должно быть ложно, т. е. $A \& \neg A = 0$.

Закон исключенного третьего. Высказывание может быть либо истинным, либо ложным, третьего не дано. Это означает, что результат логического сложения и его отрицания всегда принимает значение истина, т. е. $A \vee \neg A = 1$.

Закон двойного отрицания. Если дважды отрицать некоторое высказывание, то в результате мы получим исходное высказывание, т. е. $\neg \neg A = A$.

1.5. Позиционные системы счисления. Основные определения

Определение 1. Система счисления или нумерация – это способ записи (обозначения) чисел.

Определение 2. Символы, при помощи которых записываются числа, называются цифрами, а их совокупность – алфавитом системы счисления. Количество цифр, составляющих алфавит, называется его размерностью.

Определение 3. Система счисления называется позиционной, если количественный эквивалент цифры зависит от ее положения в записи числа.

В десятичной системе значение числа образуется следующим образом: значения цифр умножаются на «веса» соответствующих разрядов и все полученные значения складываются.

Например, $4945 = 4 \times 1000 + 9 \times 100 + 4 \times 10 + 5 \times 1$. Такой способ образования значения числа называется аддитивно-мультипликативным.

Определение 4. *Последовательность чисел, каждое из которых задает «вес» соответствующего разряда, называется базисом позиционной системы счисления.*

Основное достоинство любой позиционной системы счисления – возможность записи произвольного числа ограниченным количеством символов.

Определение 5. *Позиционную систему счисления называют традиционной, если ее базис образуют члены геометрической прогрессии, а значения цифр есть целые неотрицательные числа.*

Так, базисы десятичной, двоичной и восьмеричной систем счисления образуют геометрические прогрессии со знаменателями 10, 2 и 8 соответственно. В общем виде базис традиционной системы счисления можно записать так: $\dots p^{-2}, p^{-1}, p^0, p^1, p^2, \dots p^n$.

Определение 6. *Знаменатель p геометрической прогрессии, члены которой образуют базис традиционной системы счисления, называется основанием этой системы. Традиционные системы счисления с основанием p иначе называют p -ичными. В p -ичных системах счисления размерность алфавита равна основанию системы счисления.*

Так, алфавит десятичной системы составляют цифры 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Алфавитом произвольной системы счисления с основанием p служат числа 0, 1, ..., $p-1$, каждое из которых должно быть записано с помощью одного уникального символа, младшей цифрой всегда является 0.

В класс позиционных систем счисления также входят системы, в которых либо базис не является геометрической прогрессией, а цифры есть целые неотрицательные числа, либо базис является геометрической прогрессией, но цифры не являются целыми неотрицательными числами.

К первым можно отнести *факториальную* и *фибоначчиеву* системы счисления, ко вторым – *уравновешенные* системы счисления. Такие системы называются *нетрадиционными*. Алфавитом фибоначчиевой системы являются цифры 0 и 1, а ее базисом последовательность чисел Фибоначчи 1, 2, 3, 8, 13, 21, 34, 55,

Базисом факториальной системы счисления является последовательность $1!, 2!, 3!, \dots, n!, \dots$. В отношении алфавита этой системы можно сделать замечание: количество цифр, используемых в разряде, увеличивается с ростом разряда.

В общем случае, если система счисления устроена таким образом, что основание как таковое в ней отсутствует, а базис представляет собой последовательность чисел $\dots p_0, p_1, \dots p_n, \dots$, то количество N_k цифр, используемых в k -м разряде, определяется так:

$$N_k = \frac{p_{k-1}}{p_k}, \text{ если } p_{k-1} \bmod p_k; \text{ в противном случае } N_k = \frac{p_{k-1}}{p_k} + 1.$$

Операция «mod» здесь означает целочисленное деление.

Ниже приведена сводная таблица, характеризующая некоторые позиционные системы.

| Система счисления | Основание | Размерность алфавита | Цифры |
|--------------------------|------------------|---------------------------------------|---|
| Двоичная | 2 | 2 | 0, 1 |
| Троичная | 3 | 3 | 0, 1, 2 |
| Восьмеричная | 8 | 8 | 0, 1, 2, 3, 4, 5, 6, 7 |
| Шестнадцатеричная | 16 | 16 | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F |
| Факториальная | Нет | Увеличивается с ростом номера разряда | 1 разр.: 0, 1 2 разр.: 0, 1, 2 3 разр.: 0, 1, 2, 3... |
| Фибоначчиева | Нет | 2 | 0, 1 |

Основанием p -ичной системы счисления может быть любое натуральное число, большее единицы. Системой счисления с минимальным основанием является двоичная система, все числа в которой записываются с помощью цифр 0 и 1.

Примеры записи некоторых десятичных чисел в различных нетрадиционных позиционных системах счисления приведены в следующей таблице.

| Десятичная | Факториальная | Фибоначчиева |
|------------|---------------|--------------|
| 10 | 120 | 10010 |
| 25 | 1001 | 1000101 |
| 100 | 4020 | 1000010100 |

1.5.1. Некоторые понятия, вопросы и ответы

1) Какое множество понятий однозначно определяет позиционную систему счисления?

- базис, алфавит, основание
- базис, алфавит
- базис.

Ответ: базис.

Если в качестве цифр в системе счисления используются числа, отличные от целых неотрицательных, то для определения системы счисления необходимо описать еще и алфавит.

2) Какая последовательность может быть использована в качестве базиса позиционной системы счисления?

Ответ: Последовательность чисел может являться базисом позиционной системы счисления только тогда, когда в соответствующей этому базису системе может быть представлено любое число (если система предназначена только для нумерации целых чисел, то любое целое число).

3) Какие символы могут использоваться в качестве цифр системы счисления?

Ответ: Любые символы.

Если основание системы счисления p меньше 10, то для символического представления в ней, как правило, используются первые p десятичных цифр (от 0 до $p-1$). Например, в пятеричной системе счисления будут использоваться пять цифр: 0, 1, 2, 3, 4.

Для $10 < p < 37$ в качестве первых десяти цифр также используют их десятичное представление, а для остальных цифр буквы латинского алфавита.

Для систем счисления с основаниями, большими 36, единых правил для формы записи цифр не существует.

Если при описании произвольной p -ичной системы счисления не будет указан вид ее цифр, то принято считать, что первые десять цифр совпадают с десятичными, а следующие 26 – с латинскими буквами. Остальные цифры записываются в виде их десятичных представлений, заключенных в квадратные скобки. Так [50] в системах счисления с основаниями, большими 50, будет обозначать 51-ю по счету от 0 цифру. Максимальную цифру в произвольной p -ичной системе счисления обозначают $[p-1]$.

Любое натуральное число можно записать единственным образом, в какой угодно p -ичной системе счисления.

Пример: Десятичное число 14 можно записать:

- в двоичной системе 111_2 ($14 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1$);
- в троичной системе 112_3 ($14 = 1 \times 3^2 + 1 \times 3^1 + 2 \times 3^0$);
- в четверичной системе 32_4 ($14 = 3 \times 4^1 + 2 \times 4^0$);
- в 14-ричной системе 10_{14} ($14 = 1 \times 4^1 + 0 \times 4^0$).

В системах счисления с основанием, большим 14, данное число будет представлено одной цифрой (это будет буква латинского алфавита E или некий другой символ).

Любую правильную дробь можно представить в виде конечной или бесконечной суммы отрицательных степеней любого натурального числа $p > 1$.

Например:

$$0,123 = 1 \times 10^{-1} + 2 \times 10^{-2} + 3 \times 10^{-3} = 0,1 + 0,02 + 0,003.$$

1.5.2. Алгоритм получения дополнительного k -разрядного кода отрицательного числа

Определение 7. В p -ичной системе счисления любое неотрицательное вещественное число можно записать в виде

$$\sum_{i=-\infty}^n a_i p^i, 0 \leq a_i < p, n > 0,$$

где $p > 1$ – основание позиционной системы счисления, a_i – цифры числа a в p -ичной системе счисления.

Отрицательные числа в p -ичных системах счисления представляются с помощью знака минус перед выражением для модуля отрицательного числа.

Определение 8. Представление числа в привычной для человека форме «знак-величина», при которой старший раз-

ряд ячейки отводится под знак, остальные $k-1$ разрядов – под цифры числа, называется прямым кодом.

Например, прямые коды двоичных чисел 11001_2 и -11001_2 для восьмиразрядной ячейки соответственно равны 00011001 и 10011001 .

Положительные целые числа представляются в компьютере с помощью прямого кода.

Отрицательные целые числа представляются в компьютере с помощью дополнительного кода.

Определение 9. *k -разрядный дополнительный код отрицательного числа m – это запись в k разрядах положительного числа $2^k - |m|$, где $|m|$ – модуль отрицательного числа m , $|m| \leq 2^{k-1}$.*

Алгоритм получения этого кода выглядит следующим образом:

- 1) модуль отрицательного числа представить прямым кодом в k двоичных разрядах;
- 2) значения всех разрядов инвертировать (все нули заменить единицами, а единицы – нулями), получив, таким образом, k -разрядный обратный код исходного числа;
- 3) к полученному обратному коду, трактуемому как k -разрядное неотрицательное двоичное число, прибавить единицу.

Пример: Найдем дополнительный код числа -52 .

Для восьмиразрядной ячейки:

$0011\ 0100$ – прямой код числа $|-52| = 52$;

$1100\ 1011$ – обратный код числа 52 ;

$1100\ 1100$ – дополнительный код числа -52 .

Для шестнадцатиразрядной ячейки:

$0000\ 0000\ 0011\ 0100$ – прямой код числа $|-52| = 52$;

$1111\ 1111\ 1100\ 1011$ – обратный код числа 52 ;

$1111\ 1111\ 1100\ 1100$ – дополнительный код числа -52 .

Описанный выше алгоритм получения дополнительного кода для отрицательного числа знаковую единицу в левом разряде образует автоматически при $|m| \leq 2^{k-1}$.

Если же $2^{k-1} < |m| < 2^k$, то попытка реализации данного алгоритма приведет к тому, что в левом разряде будет находиться цифра 0 , соответствующая компьютерному представлению положитель-

ных чисел, что неверно.

Восстановить модуль отрицательного исходного десятичного числа по его дополнительному коду можно двумя способами.

Способ 1 (обратная цепочка преобразований):

- 1) вычесть единицу из дополнительного кода;
- 2) инвертировать полученный код;
- 3) перевести полученное двоичное представление числа в десятичное.

Способ 2: по приведенному выше алгоритму построить дополнительный код для имеющегося дополнительного кода искомого числа и представить результат в десятичной системе счисления.

Пример получения десятичного значения числа по его дополнительному коду 10010111_2 .

Способ 1:

- 1) из дополнительного кода вычитаем единицу
 $10010111 - 1 = 10010110$ (получили обратный код);
- 2) инвертируем полученный обратный код 01101001
(получили модуль отрицательного числа);
- 3) переводим полученное двоичное значение в десятичную систему счисления:
 $01101001_2 = 2^6 + 2^5 + 2^3 + 1 = 64 + 32 + 8 + 1 = 105$.

Способ 2:

- 1) инвертируем имеющийся дополнительный код: 01101000 ;
- 2) прибавляем единицу: $01101000 + 1 = 01101001$ (получили модуль отрицательного числа);
- 3) переводим полученное двоичное значение в десятичную систему счисления:
 $01101001_2 = 2^6 + 2^5 + 2^3 + 1 = 64 + 32 + 8 + 1 = 105$.

В таблице, расположенной ниже, приведены значения границ диапазонов для знаковых представлений в ячейках с различной разрядностью.

| Разрядность | Минимальное число | Максимальное число |
|-------------|----------------------|---------------------|
| 8 | -128 | 127 |
| 16 | -32768 | 32767 |
| 32 | -2147483648 | 2147483647 |
| 64 | -9223372036854775808 | 9223372036854775807 |

1.6. Кодирование данных двоичным кодом

Для автоматизации работы с данными, относящимися к различным типам, очень важно унифицировать их форму представления. Для этого используется прием кодирования, то есть выражение данных одного типа через данные другого типа. В вычислительной технике применяется система кодирования двоичным кодом. Она основана на представлении данных последовательностью всего двух знаков **0** и **1**. Эти знаки называются двоичными цифрами, каждая из которых представляет 1 бит. Одним битом могут быть выражены два понятия: 0 или 1 (да или нет, черное или белое, истина или ложь и т. п.). Двумя битами можно выразить четыре различных понятия: 00, 01, 10, 11. Тремя битами можно закодировать 8 различных значений: 000, 001, 010, 011, 100, 101, 110, 111. При увеличении количества разрядов в системе двоичного кодирования на единицу количество кодируемых значений увеличивается в два раза. Общая формула расчета имеет вид: $N = 2^m$, где N – количество независимых кодируемых значений; m – количество разрядов двоичного кодирования.

Кроме системы двоичного кодирования в программировании используются системы восьмеричного и шестнадцатеричного кодирования, а вообще-то в качестве основания системы счисления, используемой для кодирования, может выступать любое число.

1.6.1. Кодирование целых и вещественных чисел

Двоичный код целого числа можно получить путем деления числа на 2 до тех пор, пока частное не будет равно 1. Совокупность остатков от каждого деления, записанная справа налево вместе с последним частным, и образует двоичный аналог десятичного целого числа.

Примеры:

1) $47: 2 = 23 + 1$; 2) $23: 2 = 11 + 1$; 3) $11: 2 = 5 + 1$;

4) $5: 2 = 2 + 1$; 5) $2: 2 = 1 + 0$; 6) итак: $47_{10} = 111101_2$.

1) $252: 2 = 126 + 0$; 2) $126: 2 = 63 + 0$; 3) $63: 2 = 31 + 1$;

4) $31: 2 = 15 + 1$; 5) $15: 2 = 7 + 1$; 6) $7: 2 = 3 + 1$;

7) $3: 2 = 1 + 1$; 8) итак: $252_{10} = 00111111_2$.

Для кодирования целых чисел от 0 до 255 достаточно иметь 8 разрядов двоичного кода (8 бит). Для кодирования чисел от 0 до

65535 потребуется 16 разрядов (16 бит). Используя 24 разряда (24 бита), можно закодировать более 16,5 миллиона разных значений.

Для кодирования вещественных чисел используется 80 разрядов (80 бит). При этом действительное число предварительно преобразуется в нормализованную форму:

$$41,2346785 = 0,412346785 \times 10^2.$$

Первая часть нормализованного числа называется мантисой, а вторая – характеристикой. При этом значительная часть из 80 бит задействуется для хранения мантисы (вместе со знаком числа) и некоторое фиксированное количество бит отводится для хранения характеристики (тоже со знаком степени).

При восьмеричном кодировании десятичное число последовательно делится на 8, а при шестнадцатеричном кодировании десятичное число последовательно делится на 16. Формирование восьмеричного кода осуществляется аналогично двоичному кодированию.

Кодирование десятичных чисел двоичным, восьмеричным и шестнадцатеричным кодами, а также обратное преобразование достаточно легко можно выполнить с помощью калькулятора, имеющегося в составе пакета программ «Стандартные». При этом необходимо использовать «Инженерный» режим работы калькулятора.

1.6.2. Кодирование текстовых данных

Если каждому символу присвоить порядковый номер (целое число), то с помощью двоичного кода можно кодировать любые текстовые данные. Восми двоичных разрядов достаточно для кодирования 256 различных символов. Этого хватит, чтобы закодировать комбинациями 8 битов (1 байт) все символы английского и русского алфавитов (строчные и прописные), арабские цифры, знаки препинания, символы арифметических действий и некоторые общепринятые специальные символы.

С этой целью институт стандартизации США (ANSI – American National Standard Institute) ввел в действие систему кодирования ASCII (American Standard Code for Information Interchange – стандартный код информационного обмена США). В системе ASCII закреплены две таблицы кодирования – базовая и расширенная. Ба-

зовая таблица содержит значения кодов от 0 до 127, а расширенная относится к символам с номерами от 128 по 255.

Первые 32 кода (от 0 до 31) базовой таблицы выделены производителям аппаратных средств (в первую очередь компьютеров и печатающих устройств). Это управляющие коды, которым не соответствуют никакие символы, зато с их помощью можно управлять работой технических устройств.

Коды от 32 по 127 предназначены для кодирования символов английского алфавита, знаков препинания, цифр, арифметических действий и некоторых вспомогательных символов. Символы русского алфавита и другие специальные символы кодируются кодами расширенной таблицы от 128 по 255.

Однако рассмотренная выше система кодирования ASCII не обеспечивает кодирование алфавитов многих других языков планеты. С целью устранения этого недостатка в настоящее время используется универсальная система – **UNICODE**, основанная на 16-разрядном (2 байта) кодировании символов. Эта система позволяет обеспечить уникальные коды для 65536 различных символов. Этого количества достаточно для размещения в одной таблице символов практически всех различных алфавитов планеты.

1.6.3. Кодирование графических данных

Если графическое изображение рассматривать как комбинацию мельчайших точек (пикселей), образующих определенный узор, называемый растром, то для его кодирования, используя систему линейных координат и индивидуальные свойства каждой точки, записываемые с помощью целых чисел, можно применить двоичную систему. К индивидуальным свойствам точки относятся яркость и цвет.

Черно-белые иллюстрации представляются в виде комбинации точек с 256 градациями серого цвета. Таким образом, для кодирования яркости любой точки достаточно 8 разрядов двоичного числа.

Кодирование цветных графических изображений осуществляется с помощью декомпозиции произвольного цвета на основные составляющие. В качестве таких составляющих используются три цвета: красный (**Red, R**), зеленый (**Green, G**) и синий (**Blue, B**). Такой принцип кодирования называется системой **RGB**. При этом если для кодирования яркости каждой из основных составляющих

использовать по 256 значений (8 двоичных разрядов), то на кодирование цвета одной точки требуется 24 разряда. Такая система кодирования обеспечивает 16,5 миллионов цветов. Эта система является полноцветной и называется **True Color**. Если уменьшить количество двоичных разрядов, используемых для кодирования цвета каждой точки, то можно сократить объем данных, но при этом заметно сократится диапазон кодируемых цветов. Кодирование цветной графики двоичными числами, содержащими 16 разрядов, называется **High Color**.

На практике применяется индексный метод кодирования информации о цвете. При этом код каждой точки раstra выражает не цвет сам по себе, а только его номер (индекс) в справочной таблице, называемой *палитрой*, которая прилагается к графическим данным.

1.6.4. Кодирование звука

Для кодирования звуковой информации применяется метод таблично-волнового синтеза (**Wave-Table**). Сущность этого метода состоит в том, что используются заранее подготовленные таблицы образцов звуков. В технике такие образцы называют *эмплами*. Числовые коды звуковой информации выражают тип инструмента и номер его модели, высоту тона, продолжительность, интенсивность звука и динамику его изменения, а также некоторые параметры среды, в которой происходит звучание, и прочие параметры, характеризующие особенности звука.

1.7. Основные структуры данных

Работа с большими наборами данных автоматизируется проще, когда данные упорядочены, то есть образуют заданную структуру. При работе с данными чаще всего используются *линейная, табличная, иерархическая* структуры. При создании любой структуры данных необходимо обеспечить решение двух задач: разделение элементов данных между собой и поиск нужных элементов.

Линейные структуры – это хорошо знакомые списки. Список – это простейшая структура данных, отличающаяся тем, что каждый элемент данных однозначно определяется своим уникальным номером в массиве (списке).

Табличные структуры данных подразделяются на двумерные и многомерные.

Двумерные табличные структуры данных (матрицы) – это упорядоченные структуры, в которых адрес элемента определяется номером столбца и номером строки, на пересечении которых находится ячейка, содержащая искомый элемент.

Многомерные таблицы – это упорядоченные структуры данных, в которых адрес элемента определяется тремя и более измерениями. Для отыскания нужного элемента в таких таблицах необходимо знать параметры всех измерений (размерностей).

Линейные и табличные структуры являются простыми. Ими легко пользоваться, поскольку адрес каждого элемента задается числом (для списка), двумя числами (для двумерной таблицы) или несколькими числами для многомерной таблицы. Они также легко упорядочиваются. Основным методом упорядочения таких данных является сортировка. Недостатком простых структур данных является трудность их обновления. При добавлении, например, произвольного элемента в упорядоченную структуру возникает необходимость изменения адресных данных у других элементов.

Иерархические структуры – это структуры, объединяющие нерегулярные данные, которые трудно представить в виде списка или таблицы. В иерархической структуре адрес каждого элемента определяется маршрутом, ведущим от вершины структуры к данному элементу. Эти структуры по форме сложнее, чем линейные и табличные, но они не создают проблем с обновлением данных. Их легко развивать путем создания новых уровней. Недостатками иерархических структур являются относительная трудоемкость записи адреса элемента данных и сложность упорядочения. Поэтому для упорядочения в таких структурах применяется метод предварительной индексации. При этом каждому элементу данных присваивается свой уникальный индекс, который используется при поиске, сортировке и других действиях по обработке информации. В качестве примера иерархической структуры может служить система почтовых адресов.

1.8. Единицы представления, измерения, хранения и передачи данных

Одной из систем представления данных, принятых в информатике и вычислительной технике является система двоичного кодирования. Наименьшей единицей такого представления является *бит* (двоичный разряд).

Совокупность двоичных разрядов, выражающих числовые или иные данные, образует некий битовый рисунок. С битовым представлением удобнее работать, если этот рисунок имеет регулярную форму. В качестве таких форм используются группы из 8 битов, каждая из которых называется *байтом*. Однако во многих случаях целесообразно использовать 16-разрядное, 24-разрядное, 32-разрядное, 64-разрядное кодирование. Байт является наименьшей единицей измерения количества данных (информации).

Более крупные единицы измерения данных образуются добавлением префиксов: кило-, мега-, гига-, тера-, пета-, экса-, зетта-, йотта-.

1 килобайт (Кбайт) = 1024 байт = 2^{10} байт.

1 мегабайт (Мбайт) = 1024 Кбайт = 2^{20} байт.

1 гигабайт (Гбайт) = 1024 Мбайт = 2^{30} байт.

1 терабайт (Тбайт) = 1024 Гбайт = 2^{40} байт.

1 петабайт (Пбайт) = 1024 Тбайт = 2^{50} байт.

1 эксабайт (Эбайт) = 1024 Пбайт = 2^{60} байт.

1 зеттабайт (Збайт) = 1024 Эбайт = 2^{70} байт.

1 йоттабайт (Йбайт) = 1024 Збайт = 2^{80} байт.

В качестве единицы хранения данных (информации) принят объект переменной величины, называемый файлом.

Файл – это последовательность произвольного числа байтов, обладающая уникальным собственным именем.

Поскольку в определении файла нет ограничений на его размер, то можно представить себе файл, имеющий 0 байтов (пустой файл), и файл, имеющий любое число байтов. В определении файла особое внимание уделяется имени. Имя файла фактически несет в себе адресные данные, без которых данные, хранящиеся в файле, не станут информацией из-за отсутствия методов доступа к ним. Кроме адресной информации в имени файла могут содержаться сведения о типе данных, заключенных в нем.

Требование уникальности имени файла в вычислительной технике обеспечивается автоматически – создать файл с именем, тождественным уже существующему, не может ни пользователь, ни автоматика.

Хранение файлов организуется в иерархической структуре, которая называется файловой структурой. В качестве вершины структуры служит имя носителя, на котором сохраняются файлы. Далее файлы группируются в каталоги (папки), внутри которых могут быть созданы вложенные каталоги (папки). *Путь доступа* к файлу начинается с имени устройства и включает все имена каталогов (папок), через которые проходит. В качестве разделителя используется символ «\» (обратная косая черта). Все эти сведения образуют *спецификацию файла*, которая записывается следующим образом:

Имя носителя: \ Имя каталога 1 \ ... \ Имя каталога N \ Составное имя файла.

Пример: *C:\Игры\Стрелялки\Кролики.exe.*

Передача данных в компьютерных системах происходит с разной скоростью. Единицей измерения скорости передачи данных через последовательные порты является бит в секунду (бит/с, Кбит/с, Мбит/с). Единицей измерения скорости передачи данных через параллельные порты является байт в секунду (байт/с, Кбайт/с, Мбайт/с).

1.9. Контрольные вопросы

- 1) Назовите одно из основных направлений развития информатики.
- 2) Перечислите средства, не изучаемые в информатике.
- 3) Назовите основополагающее понятие информатики.
- 4) Назовите основной метод, используемый в информатике.
- 5) Информатика и программирование – это понятия равнозначные или неравнозначные?
- 6) Что объединяет прикладная информатика?
- 7) Как называется соответствие информации текущему моменту времени?
- 8) Определите понятие данных.
- 9) Определите понятие информации.
- 10) Как называются сведения об объектах, процессах и явлениях, повышающие степень достоверности?

- 11) Сколько символов позволяет закодировать таблица Unicode?
- 12) Каким понятием определяется информация, пригодная для переработки автоматизированными или автоматическими средствами?
- 13) Сканирование книги является операцией?
- 14) Перечислите виды данных.
- 15) Назовите числовые типы данных.
- 16) Что относится к мультимедийным видам данных?
- 17) В каком виде данные входят в состав команд компьютера?
- 18) Какого вида известна информация, если задан целочисленный тип данных?
- 19) Перечислите типовые структуры данных.
- 20) Назовите простые структуры данных.
- 21) Какие наименьшие единицы представления, измерения и хранения данных вы знаете?
- 22) Как называются наименьшие единицы передачи данных?
- 23) Перечислите основные цвета модели RGB.
- 24) Как связаны между собой виды информации?
- 25) Чему равна разность десятичного числа 205 и восьмеричного числа 105?
- 26) Найдите произведение десятичного числа 101 на двоичное число 110.
- 27) Какому двоичному числу соответствует десятичное число 255?
- 28) Какие коды используются для кодирования букв русского алфавита в ASCII?
- 29) Какие коды используются для кодирования букв английского алфавита, арабских цифр и специальных символов в ASCII?
- 30) Какие коды используются для кодирования команд управления в ASCII?

Глава 2. Технические средства реализации информационных процессов

Компьютер – это электронный прибор, предназначенный для автоматизации создания, хранения, обработки и транспортировки данных (информации).

Принципы построения вычислительной машины впервые были разработаны английским математиком и изобретателем Чарльзом Беббиджем.

ЭВМ впервые была создана американцем фон Нейманом.

Состав вычислительной системы называют конфигурацией. Конфигурация вычислительной системы включает аппаратные и программные средства, которые представляют собой отдельно аппаратную и программную конфигурации.

2.1. Аппаратная конфигурация вычислительной системы

Современные компьютеры и вычислительные комплексы имеют блочно-модульную конструкцию – аппаратную конфигурацию (**Hardware**), необходимую для исполнения конкретных видов работ и собираемую из готовых блоков и узлов. По способу расположения различают устройства внутренние и внешние. Внешними, как правило, являются большинство устройств ввода-вывода данных и некоторые устройства, предназначенные для длительного хранения данных.

Согласование между отдельными узлами и блоками выполняют с помощью аппаратных интерфейсов.

Аппаратными интерфейсами называют переходные аппаратно-логические устройства.

Стандарты на аппаратные интерфейсы называют протоколами.

Протокол – это совокупность технических условий, обеспечивающих взаимное согласование различных устройств при их совместной работе.

Многочисленные интерфейсы, присутствующие в любой вычислительной системе, можно условно разделить на последовательные и параллельные. Через последовательные интерфейсы данные передаются последовательно бит за битом, а через параллельные – одновременно группами битов. При этом количест-

во битов, участвующих в одной посылке, определяется разрядностью интерфейса (8, 16, 24, 32, 64-разрядные).

Поскольку обмен данными через последовательные интерфейсы производится битами, их производительность измеряют битами в секунду (бит/с, Кбит/с, Мбит/с). Последовательные интерфейсы применяют для подключения «медленных» устройств, когда нет существенных ограничений на продолжительность обмена данными.

Так как обмен данными через параллельные интерфейсы производится группами битов (байтами), то их производительность измеряется байтами в секунду (байт/с, Кбайт/с, Мбайт/с). Параллельные интерфейсы применяют для подключения быстродействующих устройств там, где важна скорость передачи данных.

2.2. Базовая аппаратная конфигурация компьютера

Компьютер – это электронный прибор (универсальная техническая система), предназначенный для автоматизации создания, хранения, обработки и транспортировки данных.

Конфигурацию компьютера (состав оборудования) можно гибко изменять по мере необходимости. Однако существует понятие базовой конфигурации, которую считают типовой. Понятие базовой конфигурации по мере развития техники может меняться. В настоящее время в состав базовой конфигурации включают системный блок, монитор, клавиатуру и мышь.

Системный блок является основным узлом, внутри которого установлены наиболее важные компоненты.

1) *Материнская плата* – основная плата персонального компьютера. На ней размещаются:

- процессор – основная микросхема, выполняющая большинство математических и логических операций;
- микропроцессорный комплект (чипсет) – набор микросхем, управляющих работой внутренних устройств компьютера и определяющих функциональные основные возможности материнской платы;
- шины – наборы проводников, по которым происходит обмен сигналами между внутренними устройствами компьютера;

- оперативная память (оперативное запоминающее устройство – ОЗУ) – набор микросхем, предназначенных для временно-го хранения данных, когда компьютер включен;
- постоянное запоминающее устройство (ПЗУ) – микросхема, предназначенная для длительного хранения данных и даже при выключенном компьютере;
- разъемы для подключения дополнительных устройств (слоты).

2) *Жесткий диск* – основное устройство для долговременного хранения больших объемов данных и программ. Управление работой жесткого диска выполняет аппаратно-логическое устройство – контроллер жесткого диска. К основным параметрам жестких дисков относятся емкость и производительность. Емкость современных жестких дисков может достигать нескольких Тбайт. Производительность диска оценивается скоростью внутренней передачи данных, которая может достигать 30÷80 Мбайт/с. С производительностью диска, кроме скорости внутренней передачи данных, напрямую связан параметр среднего времени доступа. Он определяет интервал времени, необходимый для поиска нужных данных. Этот показатель может составлять 5÷10 микросекунд (мкс) в зависимости от скорости вращения диска.

3) *Дисковод гибких дисков* – специальный накопитель для оперативного переноса небольших объемов информации на гибкие магнитные диски (дискеты) или с дискет на жесткий диск или в ОЗУ.

4) *Дисковод компакт-дисков* (постоянное запоминающее устройство на основе компакт-диска) – устройство для считывания больших объемов числовых данных с помощью лазерного луча. Основным параметром дисководов является скорость чтения данных. Она измеряется в кратных долях. За единицу измерения была принята скорость чтения 150 Кбайт/с. Двукратная скорость чтения 300 Кбайт/с, четырехкратная – 600 Кбайт/с и так далее

5) *Видеокарта (видеоадаптер)* – это устройство, образующее совместно с монитором видеоподсистему компьютера. Видеоадаптер выполнен в виде отдельной дочерней платы, которая вставляется в один из слотов материнской платы и называется видеокартой. Видеоадаптер выполняет функции видеоконтроллера, видеопроцессора и видеопамати. Одним из важнейших параметров видеосистемы является разрешение экрана. Для каждого размера

монитора существует свое оптимальное разрешение экрана, которое должен обеспечивать видеоадаптер. Оптимальное разрешение экрана для монитора размером 15 дюймов – 800×600 , 17 дюймов – 1024×768 , 19 дюймов – 1280×1024 (1 дюйм равен 2,54 см). Цветовое разрешение (глубина цвета) определяет количество различных оттенков, которые может принимать отдельная точка экрана. Максимально возможное цветовое разрешение зависит от свойств видеоадаптера и, в первую очередь, от количества установленной на нем видеопамяти.

б) *Звуковая карта* – устройство, выполняющее вычислительные операции, связанные с обработкой звука, речи, музыки. Звуковая карта подключается к одному из слотов материнской платы в виде дочерней карты. Основным параметром звуковой карты является разрядность, определяющая количество битов, используемых при преобразовании сигналов из аналоговой формы в цифровую форму и наоборот.

Монитор – устройство визуального представления данных. Его основными потребительскими параметрами являются размер и шаг маски экрана, максимальная частота регенерации изображения, класс защиты. Стандартные размеры мониторов 14, 15, 17, 19, 20, 21 и 22 дюйма. *Маска* – это панель с регулярно расположенными отверстиями или щелями, которая расположена перед люминофором. Шаг маски – это расстояние между отверстиями или щелями. Чем меньше шаг маски, тем четче и точнее изображение. В современных мониторах шаг маски составляет $0,25 \div 0,27$ мм. Частота регенерации (обновления) изображения показывает, сколько раз в течение секунды монитор может полностью сменить изображение. Минимальная величина частоты регенерации составляет 75 Гц, нормальная – 85 Гц и комфортная – 100 и более Гц. Класс защиты монитора определяется стандартом, которому соответствует монитор с точки зрения требований техники безопасности. В настоящее время самые жесткие нормы по параметрам, определяющим качество изображения (яркость, контрастность, мерцание, антибликовые свойства покрытия) установлены в стандарте ТСО-03.

Клавиатура – клавишное устройство управления компьютером. Служит для ввода алфавитно-цифровых (знаковых) данных, а также команд управления. Комбинация монитора и клавиатуры (терминал) обеспечивает простейший интерфейс (взаимодействие)

пользователя. С помощью клавиатуры управляют компьютерной системой, а с помощью монитора получают от нее отклик.

Мышь – устройство управления манипуляторного типа. Перемещение мыши по плоской поверхности синхронизировано с перемещением графического объекта (указателя мыши) на экране монитора. Работу мыши обеспечивает специальная системная программа – драйвер мыши. Драйвер мыши предназначен для интерпретации сигналов, поступающих через порт, и обеспечивает механизм передачи информации о положении и состоянии мыши операционной системе и работающим программам. Комбинация монитора и мыши обеспечивает полноценный тип интерфейса пользователя.

Кроме базовой конфигурации компьютера в его состав могут входить и периферийные устройства. Периферийные устройства компьютера подключаются к его интерфейсам и предназначены для выполнения вспомогательных операций. Благодаря периферийным устройствам компьютерная система (вычислительный комплекс) приобретает гибкость и универсальность.

Классификация периферийных устройств по назначению.

1) Устройства ввода данных:

- специальные клавиатуры;
- специальные манипуляторы;
- планшетные сканеры;
- ручные сканеры;
- барабанные сканеры;
- сканеры форм;
- штрих-сканеры;
- графические планшеты (дигитайзеры);
- цифровые фотокамеры.

2) Устройства вывода данных:

- матричные принтеры;
- лазерные принтеры;
- светодиодные принтеры;
- струйные принтеры.

3) Устройства хранения данных:

- стримеры;
- ZIP-накопители;

- накопители HiFD;
 - накопители JAZ;
 - магнитооптические устройства.
- 4) Устройства обмена данными (модемы).

2.3. Контрольные вопросы

- 1) Назовите автора принципов построения цифровых вычислительных машин.
- 2) Кто автор первой ЭВМ?
- 3) К какому устройству в первую очередь обращается процессор при включении компьютера?
- 4) Каковы основные характеристики процессора?
- 5) Перечислите основные характеристики монитора.
- 6) Перечислите набор компонентов в составе компьютера.
- 7) Какое устройство формирует и подает сигналы управления?
- 8) Что служит для подключения к ПК других устройств?
- 9) Что составляет основу базового программного обеспечения?
- 10) Во что преобразуются данные (информация), вводимые в компьютер?
- 11) Перечислите состав базовой конфигурации компьютера, принятый в настоящее время.
- 12) Назовите устройства памяти компьютера.
- 13) Перечислите устройства ввода информации.
- 14) Назовите устройства вывода информации.
- 15) Какие существуют режимы работы видеомонитора?
- 16) Каково назначение шин компьютера?
- 17) Как называется минимальная адресуемая ячейка оперативной памяти?
- 18) Какое количество кластеров займет файл на диске, если его размер 768 байт (размер кластера 512 байт)?
- 19) В чем заключается принцип записи данных на магнитный диск?
- 20) Как называется наименьший элемент поверхности визуализации, которому независимым образом могут быть заданы цвет, интенсивность и другие параметры?
- 21) Какого монитора не существует?
- 22) В каком устройстве не сохраняются данные при выключении компьютера?
- 23) Как называется экран монитора компьютера, работающего на основе прикосновений?

Глава 3. Программные средства реализации информационных процессов

3.1. Программная конфигурация вычислительной системы

Программа – это упорядоченная последовательность команд. Конечная цель любой компьютерной программы – управление аппаратными средствами. Программное и аппаратное обеспечение в компьютере работают в неразрывной связи и в непрерывном взаимодействии. Состав программного обеспечения (**Software**) вычислительной системы называют программной конфигурацией. В программной конфигурации между ее программами существует взаимосвязь, то есть имеет место межпрограммный интерфейс. Возможность существования такого интерфейса основана на базе технических условий и протоколов взаимодействия. На практике межпрограммный интерфейс (взаимодействие) реализуется путем распределения программного обеспечения по нескольким взаимодействующим между собой уровням. Эти уровни представляют собой пирамидальную конструкцию. Каждый следующий уровень опирается на программное обеспечение предшествующих уровней. Уровни программного обеспечения подразделяются на базовый, системный, служебный и прикладной.

Базовый уровень – самый низкий уровень программного обеспечения представляет базовое программное обеспечение. Оно обеспечивает взаимодействие с базовыми аппаратными средствами. Как правило, эти программные средства входят непосредственно в состав базового оборудования и хранятся в специальных микросхемах ПЗУ. Программы и данные записываются в микросхемы ПЗУ на этапе производства и могут быть изменены в процессе эксплуатации только при помощи специальных средств.

Системный уровень – переходной. Программы, работающие на этом уровне, обеспечивают взаимодействие прочих программ компьютерной системы с программами базового уровня и непосредственно с аппаратным обеспечением, то есть выполняют «посреднические» функции. Конкретные программы, обеспечивающие взаимодействие с конкретными устройствами, называются драйверами устройств. Они входят в состав программного обеспечения системного уровня. Программы, отвечающие за взаимодействие с пользователем, называют средствами обеспечения

пользовательского интерфейса. Совокупность программного обеспечения системного уровня образует ядро операционной системы компьютера. Если компьютер оснащен программным обеспечением системного уровня, то он уже подготовлен к установке программ более высоких уровней, к взаимодействию программных средств с оборудованием и с пользователем. Наличие ядра операционной системы – непереносимое условие для возможности практической работы человека с вычислительной системой.

Служебный уровень – это служебные программы, обеспечивающие взаимодействие с программами базового и системного уровней. Служебные программы (утилиты) предназначены для автоматизации работ по проверке, наладке и настройке компьютерной системы.

Классификация служебных программ:

- диспетчеры файлов (файловые менеджеры), которые выполняют операции, связанные с обслуживанием файловой структуры: копирование, перемещение и переименование файлов, создание каталогов (папок), удаление файлов и каталогов, поиск файлов и навигацию в файловой структуре;
- средства сжатия данных (архиваторы), которые предназначены для создания архивов. Архивирование данных упрощает их хранение, повышает эффективность использования носителя (устройства памяти) за счет того, что архивные файлы обычно имеют повышенную плотность записи информации;
- средства просмотра и воспроизведения, предназначенные для просмотра и воспроизведения документов без загрузки их в «родительскую» прикладную систему (среду разработки);
- средства диагностики, предназначенные для автоматизации процессов диагностики аппаратного и программного обеспечения;
- средства контроля (мониторинга), предназначенные для того, чтобы следить за процессами, происходящими в вычислительном комплексе;
- мониторы установки, предназначенные следить за тем, чтобы не происходило нарушений работоспособности прочих программ при установке и удалении программного обеспечения;

- средства коммуникации (коммуникационные программы), предназначенные для установления соединений с удаленными компьютерами. Они служат для обеспечения передачи сообщений электронной почты, пересылки факсимильных сообщений и множества других операций в компьютерных сетях;
- средства обеспечения компьютерной безопасности – это средства пассивной и активной защиты данных от повреждения, несанкционированного доступа, просмотра и изменения.

Прикладной уровень – комплекс прикладных программ, с помощью которых на рабочем месте обеспечивается выполнение конкретных задач.

Классификация прикладных программ:

- текстовые редакторы, предназначенные для ввода и редактирования текстовых данных;
- текстовые процессоры, обеспечивающие ввод, редактирование текста и форматирование (оформление) документов, предназначенных для печати, а также электронных документов, предназначенных для отображения на экране;
- графические редакторы, предназначенные для создания и (или) обработки графических изображений;
- системы управления базами данных (СУБД), предназначенные для создания структуры базы данных, предоставления средств для заполнения этой структуры или импорта данных из таблиц других баз данных, обеспечения возможности доступа к данным, а также предоставления средств поиска и фильтрации данных;
- электронные таблицы – это комплексные средства для хранения и обработки различных типов данных, представляемых в виде таблиц;
- **CAD**-системы (системы автоматизированного проектирования), предназначенные для проектно-конструкторских работ;
- настольные издательские системы, предназначенные для автоматизации процесса верстки полиграфических изданий;

- экспертные системы, предназначенные для анализа данных, содержащихся в базах знаний, и выдачи рекомендаций по запросу пользователя;
- редакторы **HTML** (веб-редакторы), предназначенные для создания и редактирования веб-документов (веб-страниц интернета);
- браузеры – это программные средства, предназначенные для просмотра электронных документов, выполненных в формате HTML;
- интегрированные системы делопроизводства, предназначенные для автоматизации рабочего места руководителя (создания, редактирования и форматирования простейших документов, централизации функций электронной почты, факсимильной и телефонной связи, диспетчеризации и мониторинга документооборота предприятия, координации деятельности подразделений, оптимизации административно-хозяйственной деятельности и поставки по запросу оперативной и справочной информации);
- бухгалтерские системы – это специализированные системы, сочетающие в себе функции текстовых и табличных редакторов, электронных таблиц и систем управления базами данных;
- финансовые аналитические системы, предназначенные для банковских и биржевых структур;
- геоинформационные системы (**ГИС**), предназначенные для автоматизации картографических и геодезических работ на основе информации, полученной топографическими или аэрокосмическими методами;
- системы видеомонтажа, предназначенные для цифровой обработки видеоматериалов (монтажа, создания видеоэффектов, устранения дефектов, наложения звука, титров и субтитров);
- обучающие, развивающие, справочные и развлекательные системы и программы, представляющие отдельные категории прикладных программных средств и обладающие своими развитыми внутренними системами классификации.

3.2. Операционные системы персональных компьютеров

3.2.1. Общие сведения об операционных системах

Операционная система представляет комплекс системных и служебных программных средств. С одной стороны, она опирается на базовое программное обеспечение компьютера, входящее в его систему **BIOS** (базовая система ввода-вывода), с другой стороны, она сама является опорой для программного обеспечения более высоких уровней – прикладных и большинства служебных приложений. Приложениями операционной системы принято называть программы, предназначенные для работы под ее управлением.

Основная функция всех операционных систем – посредническая. Она заключается в обеспечении нескольких видов интерфейса:

- интерфейса между пользователем и программно-аппаратными средствами компьютера (интерфейс пользователя);
- интерфейса между программным и аппаратным обеспечением (аппаратно-программный интерфейс);
- интерфейса между разными видами программного обеспечения (программный интерфейс).

Все операционные системы способны обеспечивать как пакетный, так и диалоговый режим работы с пользователем.

В пакетном режиме операционная система автоматически исполняет заданную последовательность команд.

В диалоговом режиме операционная система находится в состоянии ожидания команды пользователя и, получив ее, приступает к исполнению, а, исполнив, возвращает отклик и ждет очередной команды.

По реализации интерфейса пользователя различают неграфические и графические операционные системы.

Неграфические операционные системы реализуют интерфейс командной строки. Основным устройством управления в данном случае является клавиатура. Именно эти системы и обеспечивают диалоговый режим работы.

Графические операционные системы реализуют более сложный тип интерфейса, в котором в качестве органа управления кроме клавиатуры используется мышь или адекватное устройство позиционирования. Работа с графической операционной системой основана на взаимодействии активных и пассивных экранных эле-

ментов управления. В качестве активного элемента управления выступает указатель мыши – графический объект, перемещение которого на экране синхронизировано с перемещением мыши. В качестве пассивных элементов управления выступают графические элементы управления приложений (экранные кнопки, значки, переключатели, флажки, раскрывающиеся списки, строки меню и многие другие).

Все операционные системы (их ядра) загружаются в оперативную память автоматически при включении компьютера.

3.2.2. Файловая структура операционных систем

Все современные дисковые операционные системы обеспечивают создание файловой структуры, предназначенной для хранения данных на дисках и обеспечения доступа к ним. Принцип организации файловой структуры – табличный. Поверхность жесткого диска рассматривается как трехмерная матрица, измерениями которой являются номера поверхности, цилиндра и сектора. Под цилиндром понимается совокупность всех дорожек, принадлежащих разным поверхностям и находящихся на равном удалении от оси вращения. Данные о том, в каком месте диска записан тот или иной файл, хранятся в системной области диска в специальных таблицах размещения файлов (FAT-таблицах). К FAT-таблице предъявляются очень высокие требования по ее надежности, поскольку нарушение FAT-таблицы приводит к нарушению доступа к данным, записанным на диске. Поэтому FAT-таблица создается в двух экземплярах, идентичность которых регулярно контролируется средствами операционной системы.

Наименьшей физической единицей хранения данных является сектор. Емкость сектора составляет 512 Кбайт. Поскольку емкость FAT-таблицы ограничена, то для дисков, емкость которых превышает 32 Мбайта, обеспечить адресацию к каждому отдельному сектору невозможно. С целью устранения этого недостатка секторы условно объединяются в кластеры. Кластер – это наименьшая единица адресации к данным. Емкость кластера не фиксирована и зависит от емкости диска.

Несмотря на то, что сведения о местоположении файлов хранятся в табличной структуре, пользователю они представляются в

виде иерархической структуры, а все необходимые преобразования берет на себя операционная система.

Под управлением операционной системы осуществляются следующие функции обслуживания файловой структуры:

- создание файлов и присвоение им имен;
- создание каталогов (папок) и присвоение им имен;
- копирование и перемещение файлов между дисками и между каталогами (папками) одного диска;
- удаление файлов и каталогов (папок);
- навигация по файловой структуре с целью доступа к заданному файлу, каталогу (папке);
- управление атрибутами файлов.

Файл – это именованная последовательность байтов произвольной длины. Поскольку файл может иметь нулевую длину, то создание файла состоит в присвоении ему имени и регистрации его в файловой структуре – это одна из функций операционной системы. По способам именования файлов различают «короткое» (собственно имя) и «длинное» (полное или составное) имя. Полное имя файла состоит из двух частей: собственно имени и расширения имени. Имя от расширения отделяется точкой. Как имя, так и расширение могут состоять только из алфавитно-цифровых символов латинского (английского) алфавита. Составное имя может содержать до 256 символов. Этого вполне достаточно для создания содержательных имен файлов. Имя файла может содержать любые символы, кроме девяти специальных: \, /, :, ,, *, ?, “, <, >, |. В имени разрешается использовать пробелы и несколько точек. Расширением имени считаются все символы, идущие после последней точки. В современных операционных системах семейства MS Windows использование длинных имен файлов имеет ряд особенностей:

- в корневой папке диска (на верхнем уровне иерархической файловой структуры) нежелательно хранить файлы с длинными именами, так как в этой папке ограничено количество единиц хранения, поэтому, чем длиннее имена, тем меньше файлов можно разместить в корневой папке;
- существует жесткое ограничение на длину спецификации файла (она не может быть длиннее 260 символов), которая включает кроме составного имени путь доступа к файлу, начиная от вершины иерархической структуры;

- разрешается использовать символы любых алфавитов, в том числе и русского;
- прописные и строчные буквы не различаются операционной системой;
- во многих случаях выбор расширения имени файла не является частным делом пользователя (приложения операционных систем предлагают выбрать только основную часть имени и указать тип файла, а соответствующее расширение имени создается автоматически).

Кроме имени файла операционная система хранит для каждого файла дату его создания (изменения) и его атрибуты – это дополнительные параметры (характеристики), определяющие свойства файлов. Операционная система позволяет их контролировать и изменять. Состояние атрибутов учитывается при проведении автоматических операций с файлами. К основным атрибутам, которые обозначаются буквами английского алфавита, относятся следующие четыре:

- только для чтения (**Read only**), то есть файл не предназначен для внесения изменений;
- скрытый (**Hidden**), то есть информация о файле не отображается на экране при проведении файловых операций (это мера защиты против его случайного повреждения);
- системный (**System**) – это файлы, обладающие важными функциями в работе самой операционной системы;
- архивный (**Archive**) в прошлом использовался для работы программ резервного копирования (в современных операционных системах утратил практическое значение, так как используются другие средства для резервного копирования).

3.2.3. Базовые функции операционных систем

Основная функция операционной системы заключается в обеспечении интерфейса приложений с аппаратными и программными средствами вычислительной системы, а также с пользователем. С точки зрения управления исполнением приложений различают однозадачные и многозадачные операционные системы.

Однозадачные операционные системы передают все ресурсы вычислительной системы одному исполняемому приложению и не допускают ни параллельного выполнения другого приложения (пол-

ная многозадачность), ни его приостановки и запуска другого приложения (вытесняющая многозадачность). В то же время параллельно с однозадачными операционными системами возможна работа специальных программ, называемых резидентами. Такие программы не опираются на операционную систему, а непосредственно работают с процессором, используя его систему прерываний.

Большинство современных графических операционных систем являются многозадачными. Они управляют распределением ресурсов вычислительной системы между задачами и обеспечивают следующие возможности:

- одновременную или поочередную работу нескольких приложений;
- обмен данными между несколькими приложениями;
- совместное использование программных, аппаратных, сетевых и прочих ресурсов вычислительной системы несколькими приложениями.

От того, как операционная система управляет работой приложений, во многом зависит надежность всей вычислительной системы.

Операционная система должна предоставлять возможность прерывания работы приложений по желанию пользователя и снятия сбойной задачи без ущерба для работы других приложений. При этом надежность системы может вступать в противоречие с требованием ее универсальности.

Для правильной работы приложений на компьютере они должны пройти операцию, называемую установкой. Привязка приложения к существующей программно-аппаратной среде компьютера и его настройка на работу осуществляется именно в этой среде.

Современные графические операционные системы берут на себя управление установкой приложений. Они управляют распределением ресурсов вычислительной системы между приложениями, обеспечивают доступ устанавливаемых приложений к драйверам устройств вычислительной системы, формируют общие ресурсы, которые могут использоваться разными приложениями, выполняют регистрацию установленных приложений и выделенных им ресурсов.

Средства аппаратного обеспечения вычислительной техники отличаются широким многообразием. Существуют сотни различных моделей видеоадаптеров, звуковых карт, мониторов,

принтеров, сканеров и прочего оборудования. При таком многообразии технических устройств ни один разработчик программного обеспечения не в состоянии предусмотреть все возможные варианты взаимодействия своей программы с соответствующим устройством. Гибкость аппаратных и программных конфигураций вычислительных систем поддерживается за счет того, что каждый разработчик оборудования прикладывает к нему специальные программные средства управления – драйверы. Драйверы имеют точки входа для взаимодействия с прикладными программами, а диспетчеризация обращений прикладных программ к драйверам устройств – это одна из функций операционной системы. Современные операционные системы позволяют управлять не только установкой и регистрацией программных драйверов устройств, но и процессом аппаратно-логического подключения.

3.2.4. Прочие функции операционных систем

Кроме базовых функций операционные системы могут предоставлять различные дополнительные функции. Прочие функции операционных систем позволяют осуществлять следующие действия:

- поддерживать функционирование локальной компьютерной сети без специального программного обеспечения;
- обеспечивать доступ к основным службам интернета средствами, интегрированными в состав операционной системы;
- создавать системными средствами сервер интернета, его обслуживание и управление, в том числе и дистанционное посредством удаленного соединения;
- защищать данные от несанкционированного доступа, просмотра и внесения изменений;
- обеспечивать комфортной поочередной работой различных пользователей на одном персональном компьютере с сохранением персональных настроек рабочей среды каждого из них;
- автоматически исполнять операции обслуживания компьютера и операционной системы по заданному расписанию или под управлением удаленного сервера;
- обеспечивать работу с компьютером лицам, имеющим физические недостатки, связанные с органами зрения, слуха и другими.

Современные операционные системы могут также включать ми-

нимальный набор прикладного программного обеспечения, которое можно использовать для исполнения простейших практических задач:

- чтения, редактирования и печати текстовых документов;
- создания и редактирования простейших рисунков;
- выполнения арифметических и математических расчетов;
- ведения дневников и служебных блокнотов;
- создания, передачи и приема сообщений электронной почты;
- создания и редактирования факсимильных сообщений;
- воспроизведения и редактирования звукозаписи;
- воспроизведения видеозаписи;
- разработки и воспроизведения комплексных электронных документов, включающих текст, графику, звукозапись и видеозапись.

В дальнейшем, по мере развития аппаратных средств, неизбежно будут развиваться и функции операционных систем.

3.2.5. Контрольные вопросы

- 1) Перечислите уровни программного обеспечения.
- 2) Назовите функции операционных систем.
- 3) Каков состав системного программного обеспечения?
- 4) Как называется программное обеспечение, в котором могут размещаться антивирусные программы?
- 5) Какие действия необходимо выполнить при установке нового программного продукта?
- 6) Перечислите виды операционных систем в зависимости от реализации пользовательского интерфейса.
- 7) Что необходимо задать для доступа к файлу?

3.3. Электронные таблицы MS Excel

3.3.1. Назначение и возможности электронных таблиц

Программа MS Excel (электронные таблицы) предназначена для решения любых задач, входные и выходные данные (преимущественно числовые) которых представляются в виде таблиц.

Применение электронных таблиц упрощает работу с данными и позволяет получать результаты без проведения расчетов вручную, а также при решении многих задач обойтись без программирования.

Особенность электронных таблиц заключается в возможности применения формул для описания связи между значениями различных ячеек. Расчет по заданным формулам выполняется автоматически. Изменение содержимого какой-либо ячейки приводит к пересчету значений всех ячеек, которые с ней связаны формульными отношениями и, тем самым, к обновлению всей таблицы в соответствии с изменившимися данными.

Основные возможности электронных таблиц:

- проведение расчетов над большими наборами данных;
- автоматизация итоговых вычислений;
- решение задач путем подбора значений параметров;
- реализация любых формульных расчетов;
- обработка результатов экспериментов;
- проведение поиска оптимальных значений параметров;
- подготовка табличных документов;
- построение диаграмм и графиков по имеющимся данным.

Способы загрузки программы MS Excel:

- двойным щелчком по ярлыку Microsoft Excel на рабочем столе, если ярлык там находится;
- выполнением последовательности команд *Пуск, Программы, Стандартные, ярлык Microsoft Excel*;
- выполнением последовательности команд *Пуск, Найти, Файлы и папки*. В появившемся диалоговом окне в строке *Имя* ввести *Microsoft Excel* (имя файла ярлыка программы MS Excel) и щелкнуть по кнопке *Найти*. После окончания поиска выполнить двойной щелчок по ярлыку MS Excel.

Загрузка программы заканчивается появлением на экране окна программы с открытым рабочим листом по имени Лист1 стандартной рабочей книги с именем Книга1.

Для создания своей рабочей книги необходимо выполнить следующие действия.

- Щелчком левой кнопки мышки развернуть меню *Сервис*, щелкнуть левой кнопкой мышки по строке *Параметры* и в появившемся диалоговом окне щелкнуть мышкой по закладке *Общие*. В окошечке *Листов в новой книге* установить требуемое число листов и щелкнуть по кнопке *Ок*.
- На панели инструментов щелкнуть по кнопке *Создать*.

- Щелчком левой кнопки мышки развернуть меню *Файл* и щелкнуть мышкой по строке *Сохранить как...*. В появившемся окне щелкнуть по стрелке окна *Мои документы*. В раскрывшемся меню щелкнуть по строке с адресом вашего каталога, а затем в нижнем окне *Имя файла* вместо стандартного имени записать свое название рабочей книги, после чего щелкнуть по кнопке *Сохранить*. В последующем при работе с этим файлом такие действия выполнять не надо, если не требуется замена имени файла, а достаточно периодически щелкать по кнопке *Сохранить* на панели инструментов *Стандартная*.

3.3.2. Главное окно MS Excel

Окно MS Excel представлено на рис. 3.1.

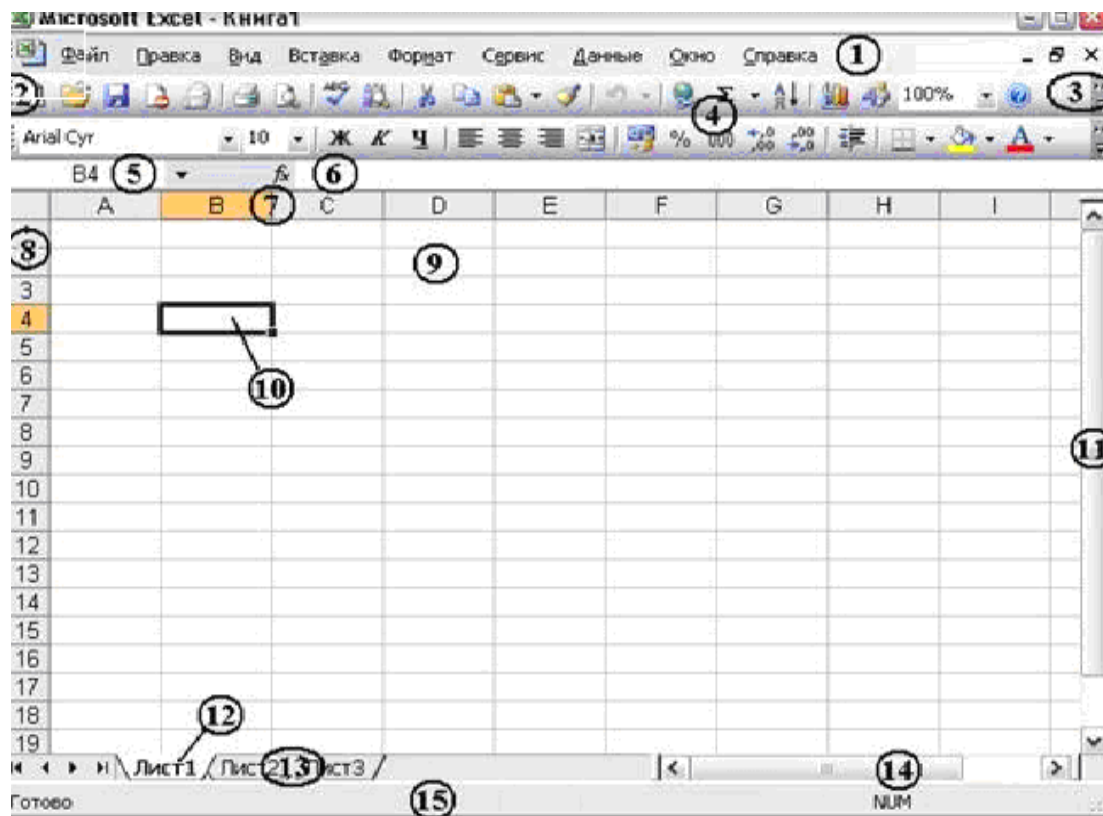


Рис. 3.1. Главное окно MS Excel

На рисунке цифрой 1 обозначено горизонтальное меню, а цифрой 15 – строка состояния данной программы. Книга состоит из *листов*. Каждый лист – отдельная электронная таблица, которая состоит из строк и столбцов, и имеет уникальное имя (области с именами листов – ярлыки листов – обозначены на рисунке цифрами

12 и 13 соответственно для активного и неактивных листов). По аналогии с активной ячейкой, активным листом называют тот лист, с которым пользователь работает в данный момент. Для того чтобы сделать лист активным, достаточно кликнуть мышью по имени листа либо выбрать нужный лист последовательным нажатием сочетания клавиш **Ctrl + PgUp** (листы будут последовательно перебираться от последнего к первому) или **Ctrl + PgDn** (листы будут последовательно перебираться от первого к последнему).

Строки нумеруются цифрами от 1 до 65536 (вертикальная область, обозначенная на рисунке цифрой 8), а столбцы – английскими буквами (горизонтальная область, обозначенная на рисунке цифрой 7). Максимальное количество строк равно 65536, а столбцов – 256. Так как в английском языке всего 26 букв, то первые 26 столбцов обозначаются буквами от «A» до «Z» (как показано на рисунке), а дальше используются их сочетания. Например, AB, AC, AD, ...

На пересечении строк и столбцов листа располагаются *ячейки* (например, одна из ячеек обозначена на рисунке цифрой 9). Каждая ячейка MS Excel имеет адрес, состоящий из индекса столбца и номера строки. Например, A1, B12, AZ83 и так далее. Ячейка, выделенная на рисунке цифрой 9, имеет адрес D2.

Иногда удобно поставить в соответствие некоторой ячейке заданного листа собственное имя и в дальнейшем обращаться к ней по этому имени. Для присвоения ячейке имени следует перейти в область отображения адреса (на рисунке область 5) и ввести имя, которое будет доступно с любого листа книги. Именованная ссылка считается абсолютной. MS Excel строго регламентирует то, как можно и как нельзя именовать ячейки и диапазоны. Приведем эти правила:

- имена ячеек и диапазонов могут иметь длину не более 255 символов;
- имя должно начинаться с буквы или с символа «_»;
- в имени могут использоваться только буквы, цифры, точка и символ подчеркивания;
- пробелы использовать запрещено;
- имя ячейки не может совпадать с ее адресом или содержащимся в ней значением;
- в качестве имени нельзя использовать одну букву или число без каких-либо знаков пунктуации или дополнительных символов.

Одна ячейка на листе является *активной* (на рисунке это ячейка В4, которая обозначена цифрой 10). Она обведена жирной рамкой, и ее адрес высвечивается в специальном поле *Имя* (на рисунке область 5). Ввод данных всегда осуществляется только в активную ячейку. При этом ее содержимое дублируется в области, обозначенной на рисунке цифрой 6 и называемой *Строка формул*. Изменить активную ячейку можно несколькими способами:

- с помощью клавиш перемещения курсора;
- кликом мыши по нужной ячейке;
- с помощью поля *Имя*, в котором необходимо записать или выбрать из раскрывающегося списка имя ячейки.

Перемещаться по листу MS Excel можно с помощью полос прокрутки (скроллинга), которые на рисунке обозначены цифрами 11 (для вертикальной прокрутки) и 14 (для горизонтальной).

Панели инструментов – средство для быстрого доступа к часто используемым командам MS Excel – обозначены на рисунке цифрой 4. Команды отображаются в виде кнопок с пиктограммами или текстом. Две дополнительные области панелей инструментов имеют особое назначение. Область 2, предназначена для изменения положения панели в окне MS Excel, а область 3 используется в случаях, когда из-за размера окна невозможно отобразить в нем все команды панели инструментов.

Количество ячеек рабочего листа, которые одновременно отображаются в окне MS Excel, определяется масштабом отображения. Масштаб отображения можно изменить (увеличив или уменьшив тем самым количество отображаемой информации) с помощью кнопки *Масштаб* панели инструментов *Стандартная*.

3.3.3. Настройки и параметры MS Excel

Настройка параметров MS Excel производится с помощью специального диалогового окна *Параметры* (рис. 3.2), которое можно открыть командой меню *Сервис / Параметры*. Для чего необходимо кликнуть мышью на строке меню *Сервис* и затем выбрать команду *Параметры*.

На вкладке *Вид* представлен ряд параметров, позволяющих определить, какие из интерфейсных элементов (строка формул, строка состояния и др.) будут отображаться, а какие нет, и какие дополнительные параметры окна будут использованы.

Вкладка *Вычисления* содержит такие параметры, как *точность*

вычислений, максимальное количество выполняемых итераций и т. п. Изменение этих опций позволяет принудительно завершить работу некоторых функций MS Excel, если произошло заикливание, то есть вычисления повторяются снова и снова, так как MS Excel не может найти решения с помощью используемого алгоритма.

Всплывающие подсказки, например, можно вклю-

чить/выключить на вкладке *Общие*. Однако наибольший интерес в настройках MS Excel представляет параметр, позволяющий обезопасить пользователя от случайных сбоев сети и при возникновении критических ошибок операционной системы и различных приложений. Для этого на

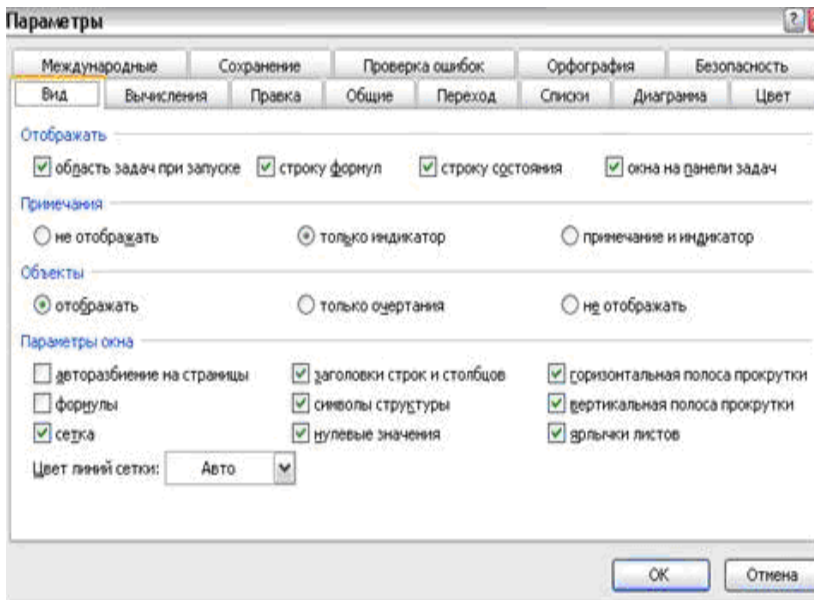


Рис. 3.2. Параметры MS Excel

вкладке *Сохранение* имеется опция *Сохранять данные для автовосстановления каждые:*, где указываются интервалы (в минутах), через которые MS Excel будет выполнять сохранение книги.

Иногда вместо букв в обозначении столбцов используются цифры 1, 2, 3, ... В этом случае имена ячеек представляются в виде R1C1, R12C5 и так далее. Такое обозначение отображается в MS Excel, если установлен стиль ссылок **R1C1**. Чтобы перейти к стандартному рассмотренному ранее стилю ссылок, следует выбрать команду меню *Сервис / Параметры*, затем в появившемся диалоговом окне перейти на вкладку *Общие*, убрать флажок *Стиль ссылок R1C1* и нажать кнопку *OK*.

Еще одна из полезных возможностей MS Excel, известная как *автозавершение*, призвана облегчить ввод текстовых заголовков. В процессе ввода MS Excel сравнивает введенный фрагмент с содержимым нескольких ячеек, расположенных выше, и, если первые символы совпали, предполагает, что данные ячеек совпадают. После этого он автоматически дописывает остальные символы значения ранее введенной ячейки к вводимому фрагменту. Если

предлагаемый текст именно тот, что необходим, следует нажать клавишу **Enter** для его ввода. В противном случае можно продолжить ввод текста. Чтобы отключить автозавершение, следует выбрать команду меню *Сервис / Параметры*, в появившемся окне перейти на вкладку *Правка* и убрать флажок на параметре *Автозавершение значений ячеек*.

3.3.4. Общие принципы работы с книгами

Файл, созданный средствами MS Excel, принято называть рабочей книгой. Рабочая книга (*.xls) является эффективным способом поддержания порядка среди пользовательских проектов. После запуска MS Excel создает новую пустую книгу с именем Книга1.xls. Эта книга, в зависимости от настроек параметров, содержит несколько листов, имена которых Лист1, Лист2, Лист3 и т. д. Задать количество листов в новой книге можно с помощью диалогового окна *Параметры* на вкладке *Общие*.

3.3.4.1. Создание и сохранение книг

Создать новую книгу можно несколькими способами.

Во-первых, новая книга создается автоматически при запуске MS Excel. Однако, если необходимость создания книги возникает после того, как пользователь уже запустил MS Excel, то существуют и иные способы.

Во-вторых, создать новую книгу можно кликнув мышью на значке *Создать* панели инструментов *Стандартная*.

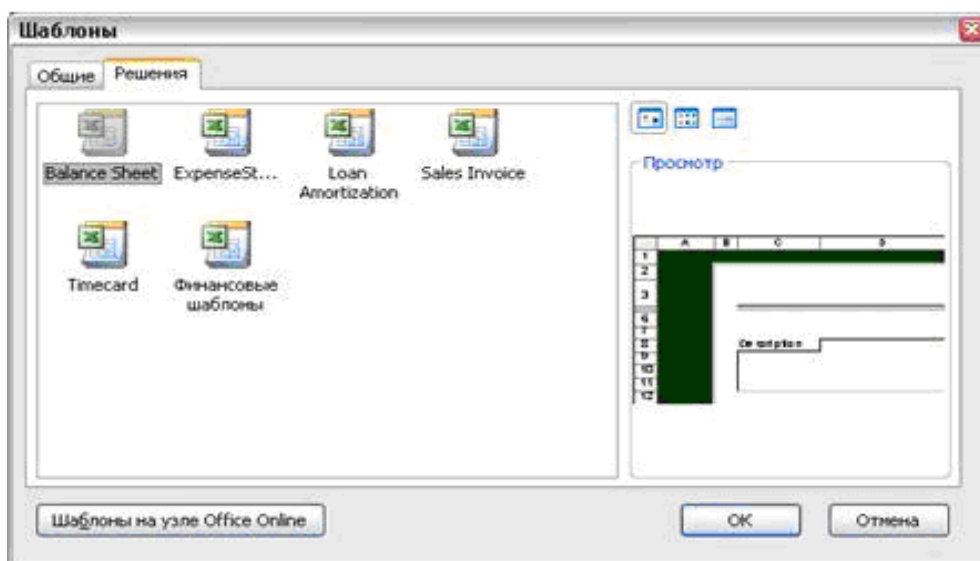


Рис. 3.3. Создание новой книги

В-третьих, это же действие можно выполнить с помощью команды меню *Файл / Создать...* . После этого на экране появится диалоговое окно *Шаблоны*, показанное на рис. 3.3.

В MS Excel 2003 вызов команды меню *Файл / Создать...* . приведет к появлению с правой стороны окна панели *Создание книги* (рис. 3.4).

Для открытия окна *Шаблоны* следует кликнуть в этой панели на команде *На моем компьютере...* . Вкладка *Общие* содержит, как правило, один единственный значок *Книга*, выбор которого и позволит создать новую книгу.

На вкладке *Решения* (рис. 3.3) размещены заготовки для создания различных офисных документов, таких как бухгалтерский баланс (Balance Sheet), расходная ведомость (Expense Statement) и др. Для использования названных шаблонов они должны быть установлены на ПК.

Для сохранения данных используется команда *Сохранить* (полностью аналогична команде меню *Файл / Сохранить*) панели инструментов

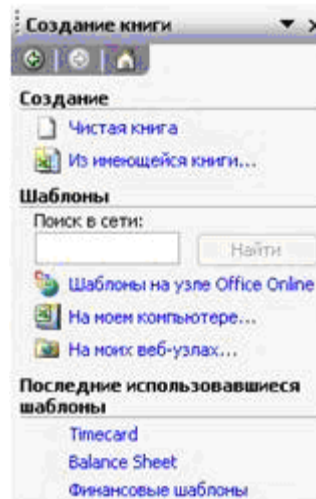


Рис. 3.4. Создание книги

инструментов

Стандартная,

которая позволяет сохранить книгу с тем же именем, под которым данная книга была сохранена ранее. Если книга ранее не сохранялась (работаем с новой книгой), то эта команда будет выполняться аналогично ко-

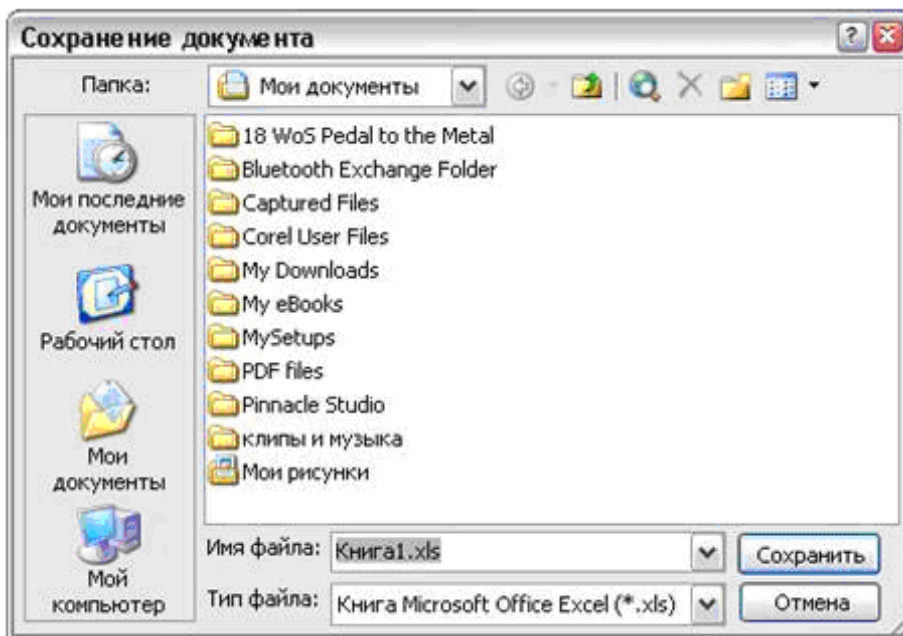


Рис. 3.5. Сохранение книги

манде меню *Файл / Сохранить как...*, при использовании которой на экране появляется диалоговое окно сохранения книги, показанное на рис. 3.5.

Как видно из этого рисунка, для сохранения книги следует указать ее положение на диске (выбрать папку для хранения) и ввести имя файла. По умолчанию файлу присваивается имя КнигаN.xls, где N – это номер книги. Следует отметить, что при сохранении книги сохраняются данные сразу всех листов.

Открыть уже существующую книгу можно с помощью команды меню *Файл / Открыть*. То же действие можно выполнить нажатием комбинации клавиш **Ctrl + O** или с помощью кнопки *Открыть* панели инструментов *Стандартная*. При этом на экране появляется диалоговое окно *Открытие документа*, в котором пользователь может выбрать нужный файл. Фактически диалоговое окно *Открытие документа* по своему внешнему виду и представлению данных полностью совпадает с окном *Сохранение документа*.

При работе с книгами MS Excel имеется одна особенность. Если открыто несколько книг, на панели задач для каждой из них создается соответствующий элемент (ярлык), однако на экране все книги отображаются в одном окне. Это может привести к неприятным последствиям, когда пользователь хочет закрыть одну из книг, но по привычке, считая, что каждая книга отображается в отдельном окне, нажимает системную кнопку *Заккрыть*. Данное действие приводит к закрытию сразу всех книг (всего приложения). Для того чтобы закрыть только одну книгу, следует использовать кнопку *Заккрыть окно*, расположенную в правом верхнем углу рабочего окна MS Excel, показанного на рис. 3.1. Использование этой кнопки приведет к закрытию только одной книги, и именно той, с которой пользователь работает в данный момент.

3.3.5. Работа с листами

По умолчанию при создании новой книги MS Excel автоматически создает в ней столько рабочих листов, сколько задано в настройках параметров. В зависимости от типа и количества вводимых данных, пользователю может понадобиться и другое количество листов.

Для удаления произвольного листа из книги следует кликнуть правой кнопкой мыши на имени этого листа и в появившемся контекстном меню (рис. 3.6) выбрать команду *Удалить*. После этого лист с выбранным именем будет удален из книги.

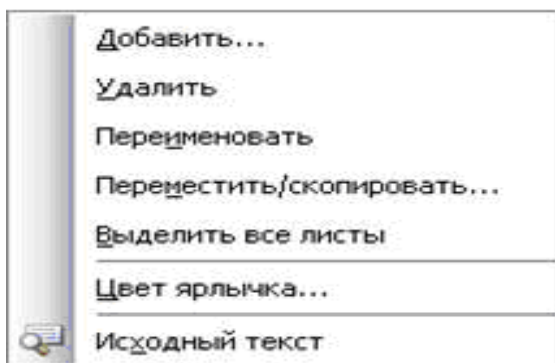


Рис. 3.6. Контекстное меню

При ее выборе на экране появляется диалоговое окно *Вставка*, показанное на рис. 3.7.

Как видно из рисунка, можно вставить как пустой лист (для этого следует выбрать изображение с именем *Лист*), так и лист с содержимым. Вставить чистый лист можно также с помощью команды меню *Вставка / Лист*. Например, можно вставить лист

с диаграммой, вставить окно диалога MS Excel, а на вкладке *Решения* выбираются листы, содержащие такие элементы, как бухгалтерский баланс (Balance Sheet), расходная ведомость (Expense Statement) и др. Вставляются новые листы слева от активного листа.

Если следование листов в книге имеет значение, то их очередность можно изменять следующим способом. Нужно зацепить мышью (нажать левую кнопку и не отпускать) за ярлык листа, перетащить его в новое положение и затем кнопку мыши отпустить. После этих действий положение листа в списке листов книги будет изменено.

Переименовать лист можно подобным же образом. На рис. 3.6 видно, что в контекстном меню имеется команда *Переименовать*, использование которой позволит задать произвольное имя листа.

Добавить новый лист в книгу можно также с помощью контекстного меню (рис. 3.6). Для этого существует команда *Добавить...*

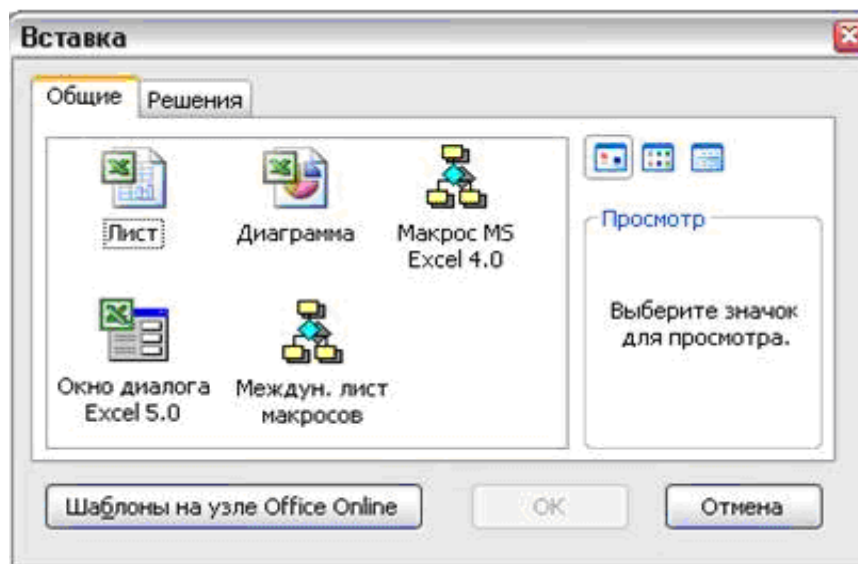


Рис. 3.7. Вставка листа

Часто возникает необходимость при изменении данных листа добавлять или удалять некоторые строки/столбцы. Эта возможность также реализована в MS Excel. Чтобы добавить/удалить какую-либо строку или столбец, достаточно кликнуть левой кнопкой мыши на области номеров строк (область 8 на рис. 3.1) или заголовков столбцов (область 7 на рис. 3.1) и в появившемся контекстном меню выбрать команду *Добавить ячейки* или *Удалить* соответственно для добавления или удаления. Следует отметить, что новые ячейки всегда вставляются перед той строкой, на которой было вызвано контекстное меню. Также вставить новую строку или столбец можно с помощью команд меню *Вставка / Строки* и *Вставка / Столбцы* соответственно.

Изменять ширину столбцов и высоту строк можно либо с помощью мыши, либо с помощью команд контекстного меню и меню *Формат*. Если подвести указатель мыши в строке заголовков столбцов к правой границе столбца (рис. 3.1, область 7), для которого требуется изменить ширину, то указатель мыши изменит свой вид. После этого необходимо нажать левую кнопку и плавно перемещать мышь вправо, не отпуская кнопку мыши. Аналогично меняется высота строки. Только указатель мыши подводится к нижней границе изменяемой строки в столбце номеров строк (рис. 3.1, область 8).

В MS Excel есть возможность одновременного ввода данных сразу на несколько листов. Для этого перед началом ввода нужно выделить интересующие листы, кликая мышью по их ярлыкам при нажатой клавише **Ctrl**. Ярлыки листов выделяются. После выделения можно приступить к вводу данных, которые автоматически будут копироваться на все выделенные листы книги.

3.3.6. Работа с фрагментами данных

Для переноса данных между приложениями, разными листами одной книги или изменения положения данных на листе разработан специальный механизм работы с фрагментами данных. Он сводится к выполнению следующих действий:

- 1) выделить ячейки, содержащие нужные данные;
- 2) скопировать (**Ctrl + Ins** или **Ctrl + C**) или переместить (**Shift + Del** или **Ctrl + X**) их в буфер обмена;
- 3) переместиться в новую позицию документа;

4) вставить данные (**Shift + Ins**).

Выделение может осуществляться несколькими способами. Если нужно выделить столбец или строку целиком, то достаточно кликнуть мышкой на номере строки или заголовке столбца. При этом при выделении столбца меняется изображение указателя мыши на «**↓**», а при выделении строки на «**→**». Выделенные данные будут отмечены цветом.

Для выделения всех данных таблицы нужно либо нажать сочетание клавиш **Ctrl + A**, либо кликнуть мышью в левом верхнем квадрате листа на пересечении строки заголовков столбцов и столбца номеров строк, как показано на рис. 3.8.

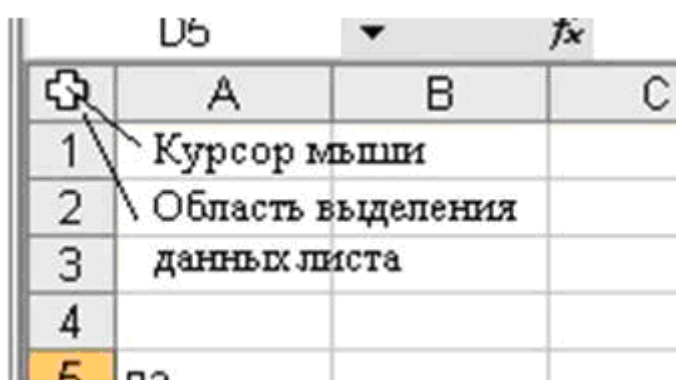


Рис. 3.8. Выделение данных листа

при нажатой клавише **Shift**, стрелками перемещения курсора вниз и вправо выделить прямоугольную область ячеек. Или выделить правую нижнюю ячейку и, при нажатой клавише **Shift**, выделить необходимую область стрелками перемещения курсора вверх и влево. То же действие можно выполнить и мышью. Для этого достаточно выделить левую верхнюю ячейку и, нажав клавишу **Shift**, кликнуть на правую нижнюю ячейку прямоугольной области или наоборот.

Пример выделения ячеек представлен на рис. 3.9. Как видно, на рисунке выделено три прямоугольных области ячеек, две строки, а также пять произвольных ячеек. То есть можно комбинировать различные способы выделения. Это осуществляется с помощью клавиши **Ctrl**. Для создания выделения, показанного на рис. 3.9, необходимо выполнить следующие действия:

- 1) Нажать левую кнопку мыши на ячейке B2 и, не отпуская ее, переместить указатель мыши в ячейку C4. Отпустить кнопку. При этом выделится первая область.

Можно также выделять как некоторую группу ячеек, так и одну произвольную ячейку. Для выделения группы расположенных рядом ячеек можно использовать как клавиатуру, так и мышь.

При выделении ячеек с помощью клавиатуры следует сделать активной крайнюю левую верхнюю ячейку и,

- 2) Нажать клавишу **Ctrl** и, не отпуская ее, выполнить оставшиеся шаги 3-9.
- 3) Кликнуть мышью по ячейкам F3, G2, H4.
- 4) Нажать левую кнопку мыши на ячейке C8 и, не отпуская ее, переместить указатель мыши в ячейку C13. Отпустить кнопку.
- 5) Кликнуть мышью на цифре 6 в колонке номеров строк.
- 6) Кликнуть мышью по ячейке E11.
- 7) Нажать левую кнопку мыши на ячейке G8 и, не отпуская ее, переместить указатель мыши в ячейку J13. Отпустить кнопку.
- 8) Кликнуть мышью на цифре 15 в колонке номеров строк.
- 9) Кликнуть мышью по ячейке B10.

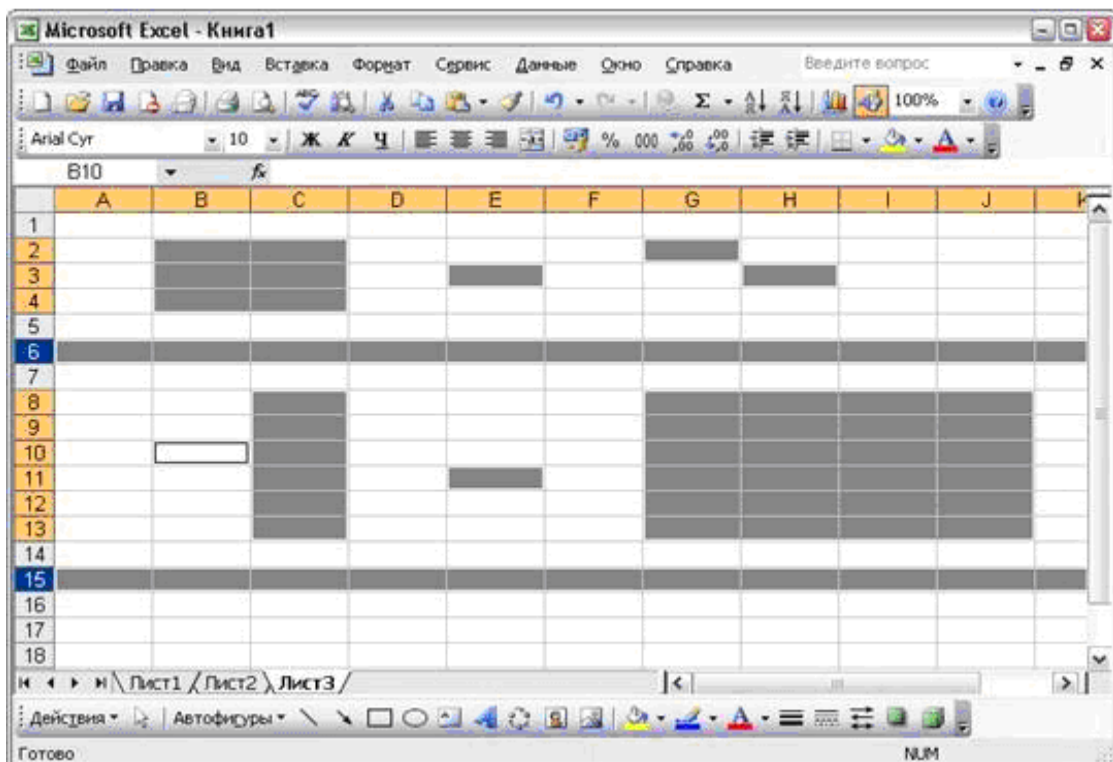


Рис. 3.9. Пример выделения ячеек в MS Excel

Следует отметить еще одну полезную функциональность. Если в рассмотренном выше случае использовать нажатие одной клавиши **Enter** вместо сочетания **Ctrl + Enter**, то можно последовательно вводить содержимое в каждую из выделенных ячеек в том порядке, в котором выполнялось их выделение. Переход на следующую ячейку будет выполняться автоматически, а для ввода следующего значения не требуется повторно переходить в строку формул. Дру-

гими словами, если нужно последовательно ввести значения в несколько отдельно стоящих ячеек, то их следует выделить, перейти в строку формул и вводить соответствующие значения, завершая ввод каждого из них нажатием клавиши **Enter**. Например, 1 **Enter**, 2 **Enter** и так далее

Заметим, что если при вводе данных выделено несколько листов, то все введенные значения будут скопированы в соответствующие ячейки каждого листа.

Бывает ситуация, когда размера ячейки (как правило, ее ширины) недостаточно для отображения, размещенного в ней значения. В такой ситуации MS Excel сообщает об этом пользователю, заполняя ячейку последовательностью символов «#####».

3.3.7. Ввод данных в ячейки

Все операции над данными осуществляются с использованием ячеек, в которых эти данные и хранятся. Содержимым ячейки может быть:

- 1) *число* – в качестве разделителя целой и вещественной частей выступает символ, указанный в региональных настройках MS Windows (по умолчанию это символ «,»);
- 2) *текст* – все, что не может быть интерпретировано как число, дата, время или формула;
- 3) *дата, время* или *дата и время*: в одном из возможных форматов;
- 4) *логическая константа* (значения ЛОЖЬ либо ИСТИНА);
- 5) *формула*, которая начинается с символа «=» и может включать любые операции, а также все доступные в MS Excel функции (формула – это, с точки зрения любой системы программирования, правильное выражение).

Для ввода данных в ячейку следует сделать ее активной и затем вводить информацию, которая будет отображаться как в самой ячейке, так и в строке формул (область б на рис. 3.1). При этом ранее введенные в ячейку данные будут заменены вводимыми значениями.

Если требуется не заменять, а только изменить ранее введенное содержимое ячейки, то можно воспользоваться одним из следующих способов:

- дважды кликнуть мышью на нужной ячейке и после этого

приступить непосредственно к редактированию ее содержимого на листе;

- сделать ячейку активной и изменять ее содержимое в строке формул (все изменения в этом случае будут отображаться в активной ячейке);
- сделать ячейку активной и нажать клавишу **F2**, что, в принципе, эквивалентно двойному клику мыши.

Последний способ хорош тем, что позволяет, не отрываясь от клавиатуры, выполнять ввод и изменение данных, в значительной степени повышая скорость набора.

Иногда бывает нужно ввести одинаковые значения в несколько разных ячеек одного листа. В MS Excel это можно сделать следующим образом:

- 1) выделить ячейки, в которые следует вводить данные (можно использовать любые способы выделения ячеек);
- 2) перейти в строку формул, кликнув на ней мышью;
- 3) ввести значение;
- 4) нажать сочетание клавиш **Ctrl+Enter**.

После этого введенные в строке формул данные будут автоматически скопированы во все выделенные ячейки.

3.3.8. Автозаполнение

MS Excel позволяет автоматически заполнять смежные ячейки числами, датами либо другими элементами.

3.3.8.1. Заполнение числами

Можно заполнять смежные ячейки числами в соответствии с арифметической или геометрической прогрессией. Чтобы задать арифметическую прогрессию, необходимо выполнить следующие действия:

- 1) ввести в смежные ячейки первые два числа прогрессии;
- 2) выделить эти ячейки;
- 3) перетащить маркер заполнения (на рис. 3.10 тоненький «плюс» в правом нижнем углу ячейки) по ячейкам, которые нужно заполнить, и отпустить кнопку мыши.

Нужно отметить, что предложенный порядок выделения отдельных ячеек и групп можно изменить. Для отмены выделения

следует кликнуть мышью в любом месте листа.

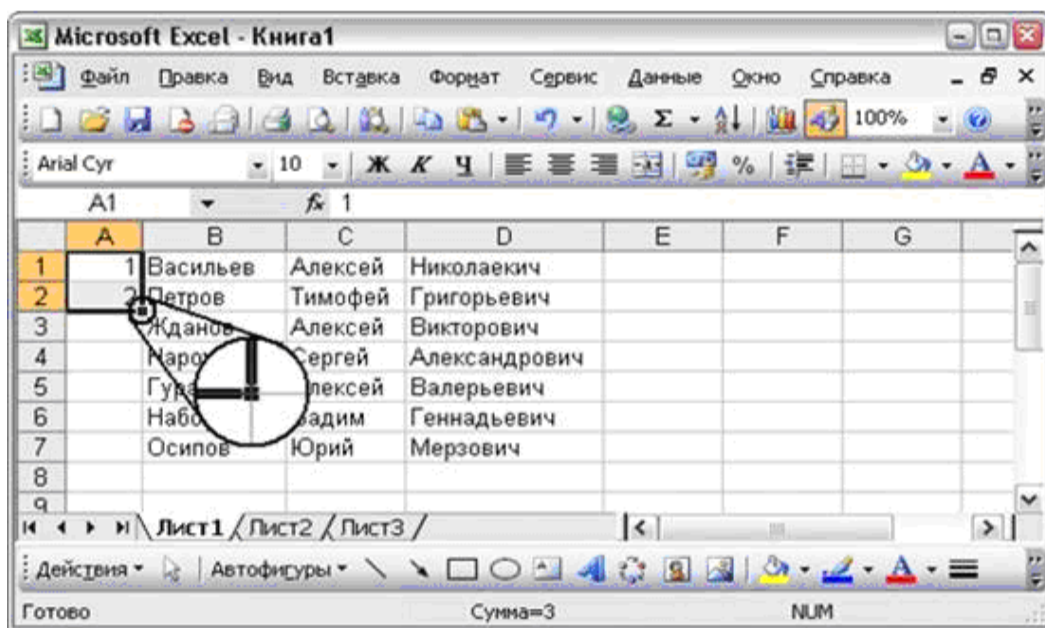


Рис. 3.10. Пример заполнения ячеек последовательностями

Копировать, перемещать содержимое выделенных ячеек в буфер обмена, а также вставлять затем его содержимое на лист можно с помощью команд контекстного меню *Копировать* или *Вырезать* и затем *Вставить*. Для вызова контекстного меню нужно кликнуть правой кнопкой мыши на любой выделенной ячейке рабочего листа.

В том случае, если необходимо заполнить ячейки числами, идущими по порядку (арифметическая прогрессия с шагом единица), достаточно заполнить только одну ячейку. Затем перетащить маркер заполнения с нажатой клавишей **Ctrl**.

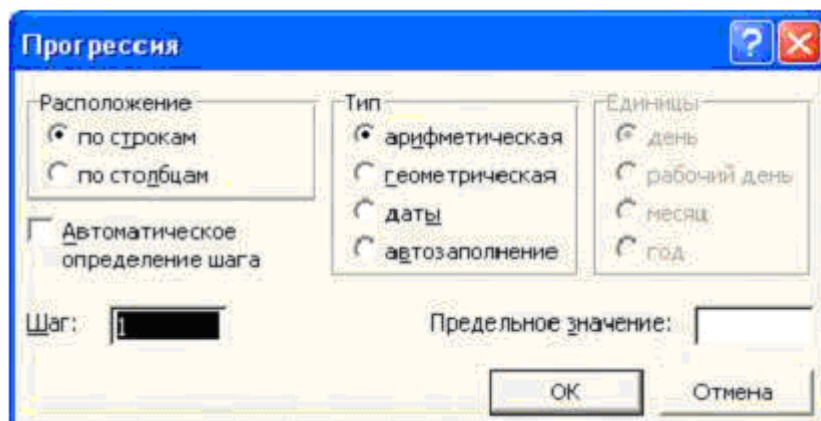


Рис. 3.11. Создание прогрессии

Второй способ создания прогрессий основан на использовании специального диалогового окна *Прогрессия* (рис. 3.11).

Для этого необходимо выполнить следующие действия:

- 1) ввести значение первого элемента прогрессии;

- 2) вызвать диалоговое окно *Прогрессия* с помощью команды меню *Правка / Заполнить / Прогрессия*;
- 3) задать направление заполнения (по строкам или столбцам);
- 4) ввести значение шага;
- 5) выбрать тип прогрессии (арифметическая, геометрическая, даты);
- 6) указать последнее значение в ряде и нажать кнопку *ОК*.

3.3.8.2. Заполнение датами

Заполнение смежных ячеек датами аналогично созданию арифметических прогрессий. Необходимо в двух смежных ячейках задать первые две даты. Затем выделить эти ячейки и с помощью маркера заполнения (маленький черный квадрат в нижнем правом углу выделенного диапазона, при наведении на него указатель мыши принимает вид черного креста) заполнить остальные.

3.3.8.3. Списки данных

Помимо числовых последовательностей и дат автозаполнение применимо к некоторым спискам данных. Существуют стандартные списки и пользовательские. Стандартных списков немного. Это названия месяцев в полной и сокращенной форме (“Январь”, “Февраль” и другие или “Янв”, “Фев” и другие), дни недели в полной и сокращенной форме (“Понедельник”, “Вторник” и другие или “Пн”, “Вт” и другие).

Кроме стандартных списков, можно создавать свои собственные. Для этого необходимо:

- 1) с помощью команды меню *Сервис / Параметры* открыть диалоговое окно *Параметры*;
- 2) перейти на вкладку *Списки* (рис. 3.2);
- 3) в окне *Элементы списка* ввести элементы нового списка, разделяя их нажатиями клавиши **Enter**;
- 4) нажать кнопку *Добавить* (при этом в левом окне *Списки*: появится новый элемент).

Можно также импортировать список из книги, для чего следует выполнить такие действия:

- 1) набрать в смежных ячейках любого листа книги список;
- 2) выделить его;

- 3) открыть окно *Параметры* и перейти на вкладку *Списки*;
- 4) нажать кнопку *Импорт*.

3.3.9. Работа с формулами

Как уже упоминалось, любая формула MS Excel начинается с символа «=». За знаком равенства следует выражение любого допустимого типа (арифметическое, текстовое, логическое, типа дата и время). При использовании на листе формул в ячейках отображается только результат вычисления. Для изменения самой формулы следует пользоваться строкой формул, в которой она и отображается. Формула (выражение) в качестве операндов (данных, над которыми производится действие) может включать в себя следующие элементы: константы, ссылки, функции. Операнды в выражении (формуле) отделяются друг от друга знаками *операций*.

Функция – специальная стандартная формула (алгоритм), которая возвращает результат выполнения определенных действий над значениями, выступающими в качестве аргументов (параметров). Функции позволяют упростить вычисления, особенно если они сложные или громоздкие.

3.3.9.1. Простые выражения

В MS Excel нет понятия «*переменная*». Все данные для расчета значения какого-либо выражения или функции помещаются в ячейки, а в выражениях используются ссылки на эти ячейки.

Константа – постоянное (не вычисляемое) значение. Например, число 23 и текст “Квартальная премия” являются константами. Любая ячейка может содержать либо константу, либо формулу.

Операция – элементарное действие над операндами – знак или символ, задающий тип вычисления в формуле. Существуют арифметические (числовые), текстовые, операции отношения (сравнения). В MS Excel некоторые операции определены и для дат.

Арифметические операции служат для выполнения арифметических действий, таких как возведение в степень, умножение, деление, сложение и вычитание. Эти операции выполняются над числами. В MS Excel используются арифметические операции, представленные в табл. 3.1 в порядке приоритета выполнения.

Таблица 3.1

| Арифметическая операция | Значение (пример) |
|-------------------------|---------------------------------|
| ^ (крышка) | Возведение в степень ($C2^2$) |
| * (звездочка) | Умножение ($B3 * B5$) |
| / (косая черта – слэш) | Деление ($A1 / A3$) |
| + (знак плюс) | Сложение ($A1 + B1$) |
| - (знак минус) | Вычитание ($A2 - A1$) |

Операции *отношения* используются для сравнения двух значений (табл. 3.2). Результатом сравнения является одно из двух возможных логических значений: либо ИСТИНА, либо ЛОЖЬ.

Таблица 3.2

| Операция отношения | Значение (пример) |
|----------------------------|---------------------------------|
| < (знак меньше) | Меньше ($A1 < B1$) |
| <= (знак меньше или равно) | Меньше или равно ($A1 <= B1$) |
| = (знак равенства) | Равно ($A1 = B1$) |
| <> (знак не равно) | Не равно ($A1 <> B1$) |
| > (знак больше) | Больше ($A1 > B1$) |
| >= (знак больше или равно) | Больше или равно ($A1 >= B1$) |

Единственная *текстовая* операция – *конкатенация* (*сцепление*) – обозначается символом амперсанд (&) и используется для объединения (сцепления) нескольких строк текста в одну. Например, =“Саша”&“Маша” даст в результате “СашаМаша”.

3.3.9.2. Копирование формул

Для получения нескольких значений необходимо под каждый аргумент и под каждое значение функции отвести отдельную ячейку. Вводить исходную формулу многократно достаточно утомительно. В таких случаях пользуются возможностями MS Excel и копируют формулу с помощью буфера обмена или с использованием маркера заполнения. При копировании формулы входящие

в ее состав ссылки на ячейки, если они записаны в относительной форме) изменяются по следующим правилам:

- если при копировании происходит смещение формулы на N столбцов вправо (влево), то индекс столбцов в относительных ссылках увеличивается (уменьшается) на N единиц;
- если при копировании происходит смещение формулы на N строк вниз (вверх), то индекс строк относительной ссылки на ячейку увеличивается (уменьшается) на N единиц.

Абсолютные же ссылки при копировании формул не изменяются.

3.3.10. Ссылки

Ссылка указывает на ячейку или диапазон ячеек листа (или нескольких листов – объемная ссылка) и передает в MS Excel сведения о расположении данных, которые требуется использовать в формуле. При помощи ссылок можно использовать в одной формуле данные, находящиеся в разных частях листа, а также использовать в нескольких формулах данные из одной ячейки. Кроме того, можно задавать ссылки на ячейки других листов той же книги и других рабочих книг (внешние ссылки). Ссылки на ячейки других книг называются *связями*.

По умолчанию MS Excel использует стиль ссылок **A1**, в котором столбцы именуются буквами (от A до IV, всего не более 256 столбцов), а строки нумеруются числами (от 1 до 65536).

В формулах возможно использование специальных функций, которые выполняют операции над несколькими данными, хранящимися в различных ячейках. Для обращения к группе (диапазону) ячеек используются интервальные ссылки, которые задаются адресом левого верхнего угла, отделенного от адреса правого нижнего угла двоеточием («:»). Если в формуле используется несколько ссылок (список), то элементы этого списка отделяются друг от друга точкой с запятой («;»).

Данные, используемые в формуле, могут находиться в одной строке, столбце или занимать некоторую прямоугольную область. В табл. 3.3 представлены возможные варианты расположения данных и ссылки на них.

Если данные располагаются не в смежных ячейках, то адреса этих ячеек просто перечисляются через точку с запятой.

Ссылки по форме записи подразделяются на *относительные*, *абсолютные* и *смешанные*.

Таблица 3.3

| Ячейка или диапазон | Ссылка |
|---|---------|
| Ячейки с данными располагаются в столбце С, с 10 по 20 строки | C10:C20 |
| Ячейки с данными располагаются в строке 15, со столбца В по Е | B15:E15 |
| Данные занимают все ячейки в строке 7 | 7:7 |
| Данные занимают все ячейки в строках с 7 по 10 | 7:10 |
| Данные занимают все ячейки в столбце N | N:N |
| Данные занимают все ячейки в столбцах с N по J | N:J |
| Ячейки занимают столбцы A÷E, строки 10÷20 (прямоугольная область, где ячейка A10 обозначает верхний левый угол, а ячейка E20 – нижний правый) | A10:E20 |

Относительная ссылка в формуле, например A1, основана на относительной позиции ячейки, содержащей формулу, и ячейки, на которую эта ссылка указывает. Как это было отмечено выше, при изменении позиции ячейки, содержащей формулу с относительными ссылками, изменяются и эти ссылки.

Абсолютная ссылка в формуле – это всегда ссылка на ячейку, расположенную в определенном месте. При изменении позиции ячейки, содержащей такую формулу, абсолютная ссылка не изменяется. Чтобы превратить относительную ссылку в абсолютную, необходимо перед индексами столбца и строки поставить знак «\$» (нажать клавишу **F4**). Например:

- 1) C1 – относительная ссылка;
- 2) \$C\$1 – абсолютная ссылка;
- 3) C\$1, \$C1 – смешанные ссылки.

Смешанная ссылка содержит либо абсолютный столбец и относительную строку, либо абсолютную строку и относительный столбец. Абсолютная ссылка по столбцу приобретает вид \$A1, \$B1 и т. д. Абсолютная ссылка по строке выглядит как A\$1, B\$1 и т. д. При изменении позиции ячейки, содержащей формулу, относитель-

ная ссылка изменяется, а абсолютная ссылка не изменяется. Для перехода от относительной ссылки к абсолютной или к смешанной ссылке можно использовать клавишу **F4**.

В формулах можно использовать данные, расположенные на разных листах. Для обращения к этим данным используются трехмерные (объемные) ссылки.

Трехмерная ссылка представляет собой ссылку на ячейку или диапазон ячеек нескольких листов рабочей книги, имена которых в ссылке отделяются друг от друга двоеточием. MS Excel использует все листы, указанные между начальным и конечным именами в ссылке. Например, с помощью формулы =СУММ(Лист2:Лист13!B5) суммируются все значения, содержащиеся в ячейке B5 на всех листах в диапазоне от Лист2 до Лист13 включительно. При использовании трехмерных ссылок необходимо придерживаться следующих правил:

- 1) трехмерные ссылки могут быть использованы для создания ссылок на ячейки других листов, определения имен и создания формул с использованием следующих функций: СУММ, СРЗНАЧ, СРЗНАЧА, СЧЕТ, СЧЕТЗ, МАКС, МАКСА, МИН, МИНА, ПРОИЗВЕД, СТАНДОТКЛОН, СТАНДОТКЛОНА, СТАНДОТКЛОНП, ДИСП, ДИСПА, СТАНДОТКЛОНПА, ДИСПР, ДИСПРА;
- 2) трехмерные ссылки нельзя использовать в формулах массива (формула, выполняющая несколько вычислений над одним или несколькими наборами значений, а затем возвращающая один или несколько результатов).

При перемещении, копировании, вставке и удалении листов, объемная ссылка изменяется. Предположим, что используется формула =СУММ(Лист2:Лист6!A2:A5), позволяющая просуммировать содержимое ячеек с A2 по A5 листов от Лист2 до Лист6 включительно. Тогда результат вычислений, в зависимости от выполненных действий, будет изменяться по следующим правилам.

- 1) *Вставка или копирование*. Если между Лист2 и Лист6 книги вставить новые листы, то MS Excel добавит в сумму содержимое ячеек с A2 по A5 на новых листах.
- 2) *Удаление*. Если между Лист2 и Лист6 книги удалить листы, то MS Excel исключит из суммы содержимое ячеек удаленных листов.

- 3) *Перемещение*. Если переместить листы, находящиеся между Лист2 и Лист6 книги, и разместить их таким образом, что они будут расположены перед Лист2 или после Лист6, то MS Excel исключит из суммы содержимое ячеек перемещенных листов.
- 4) *Перемещение граничного листа*. Если переместить Лист2 или Лист6 в новое место книги, то MS Excel включит в сумму содержимое ячеек листов, находящихся между Лист2 и Лист6 включительно.
- 5) *Удаление граничного листа*. Если удалить Лист2 или Лист6, то MS Excel включит в сумму содержимое ячеек листов, находившихся между ними.

3.3.11. Встроенные функции

Помимо простых арифметических операций, MS Excel имеет

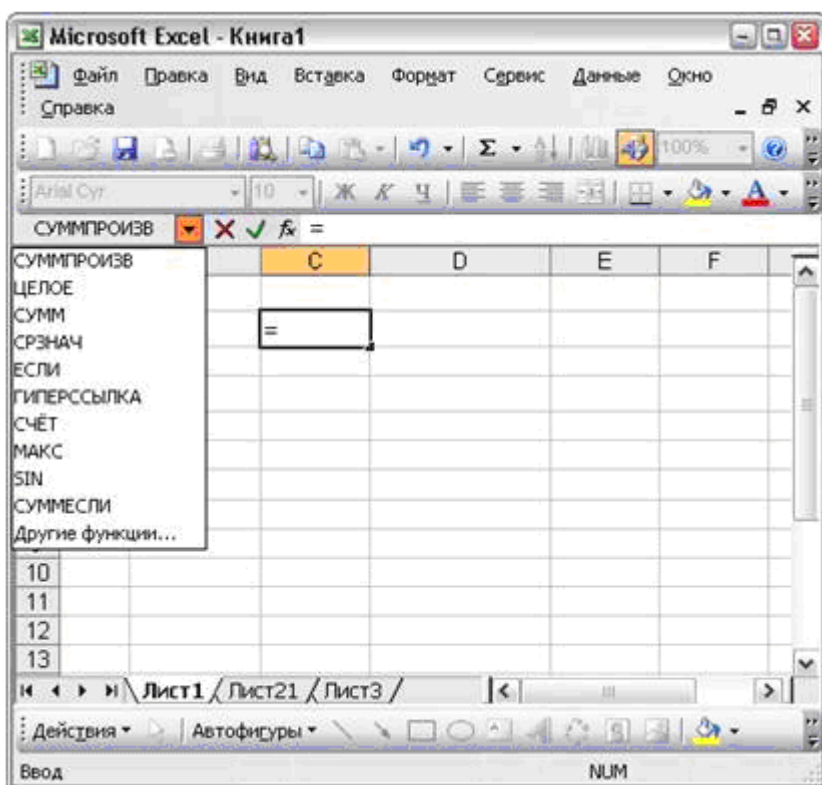


Рис. 3.12. Выбор функций

достаточно большой набор встроенных функций (функций рабочего листа), которые можно использовать в формулах для получения результата. Нужно отметить, что пользователь может создавать и собственные функции с помощью VBA и использовать эти функции для реализации вычислений.

При обращении к функции

нужно к функции

необходимо задать две части: ее имя (например, СРЗНАЧ, СУММ, ABS и др.) и набор входных параметров (аргументов), используе-

мых для вычисления результата. Имена функций могут быть набраны как строчными, так и заглавными буквами.

В зависимости от функции ее аргумент могут представлять собой число, текст, логическое значение, ссылку и т. д. В качестве аргументов функции можно также использовать другие функции и формулы (в общем случае, это выражение соответствующего типа).

После начала создания формулы доступ к функциям MS Excel осуществляется через специальные области, появляющиеся рядом со строкой формул (рис. 3.12), которые содержат кнопки *Ввод* и *Отмена*, а также выпадающий список из имен десяти последних использованных функций. Последний элемент этого выпадающего списка – команда *Другие функции...*, выбор которой приводит к появлению на экране окна мастера функций. Мастер функций можно вызвать и с помощью кнопки *Вставка функции*.

Чтобы вставить функцию в формулу, следует выбрать ее имя из выпадающего списка и указать аргументы. Имя функции можно вводить вручную, но в этом случае также вручную нужно будет указывать и аргументы функции. Ввод аргументов осуществляется в специальном окне, показанном на рис. 3.13.

Задать значение аргумента можно и с помощью клавиатуры, для чего следует просто записать необходимые ссылки. Возможно также использование мыши. Чтобы ввести ссылку на ячейку или диапазон ячеек с помощью мыши, необходимо:

- нажать на кнопку, расположенную справа от строки ввода (при этом окно *Аргументы функции* сворачивается);
- с помощью мыши выделить необходимые ячейки или диапазон ячеек (в свернутом окне *Аргументы функции* автоматически записывается ссылка);
- нажать на кнопку, чтобы развернуть окно *Аргументы функции*.

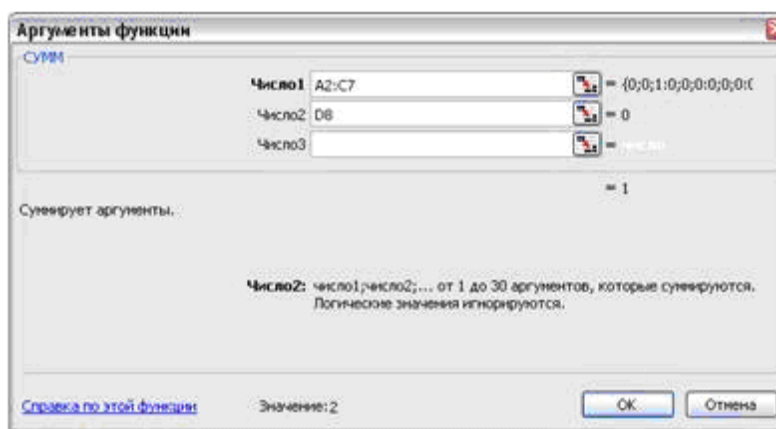


Рис. 3.13. Ввод аргументов функции

О том, для чего используется функция и каково назначение ее параметров, можно прочесть в соответствующем разделе справочной системы MS Excel, доступной после нажатия ссылки *Справка по этой функции* в окне *Аргументы функции*.

Выбрать одну из полного набора функций MS Excel можно после того, как в выпадающем списке имен функций (рис. 3.12) будет выбран элемент *Другие функции...*. Выбор этого элемента приводит к появлению мастера функций, окно которого показано на рис. 3.14.

В окне мастера функций (рис. 3.14) можно выделить четыре области. Область, отмеченная цифрой 1, предназначена для поиска необходимой функции по ее описанию. Область 2 позволяет включить сортировку функций по категориям (арифметические, логические и т. д.). В области 3 отображаются имена функций, соответствующие выбранной категории. В представленном на рисунке примере выбрана категория *Полный алфавитный перечень*, поэтому область 3 заполнена именами абсолютно всех доступных в MS Excel функций рабочего листа. Краткое описание функции выводится в области, отмеченной цифрой 4. Все эти действия по выбору функции выполняются на первом шаге мастера функций. На втором шаге пользователю предлагается ввести список аргументов. Технология ввода аргументов рассмотрена ранее.

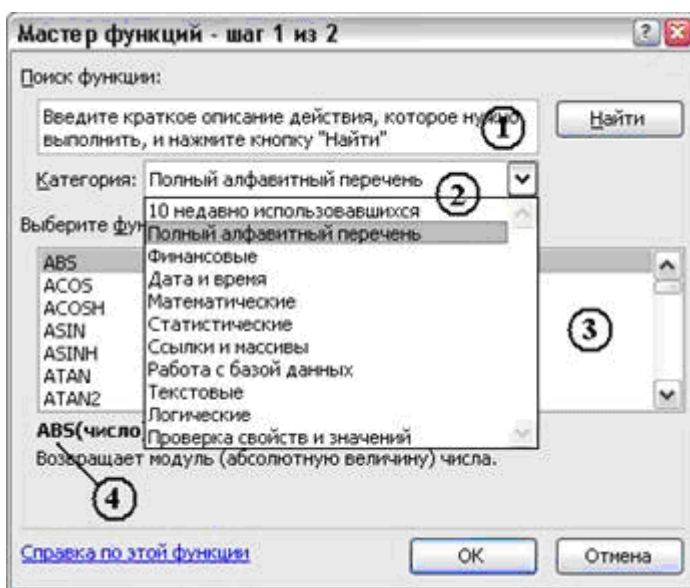


Рис. 3.14. Окно мастера функций

3.3.12. Форматирование рабочих листов

Все рассмотренные выше приемы относились к изучению способов ввода данных. Но когда речь заходит о представлении данных пользователю, недостаточно отображать «голые цифры». Требуется, по крайней мере, пояснять выводимые значения и указывать единицы их измерения. Использование же элементов оформления

позволит сделать отображаемые на листе данные максимально читаемыми.

Для выполнения тривиальных задач (изменение типа, размера или цвета шрифта, выравнивание и т. д.) можно воспользоваться панелью инструментов *Форматирование*.

Получить доступ ко всем возможностям форматирования ячеек можно с помощью команды меню *Формат / Ячейки* или команды контекстного меню *Формат ячеек*. В результате появится диалоговое окно *Формат ячеек* с шестью вкладками (рис. 3.15).

Назначение каждой из вкладок приведено в табл. 3.4.

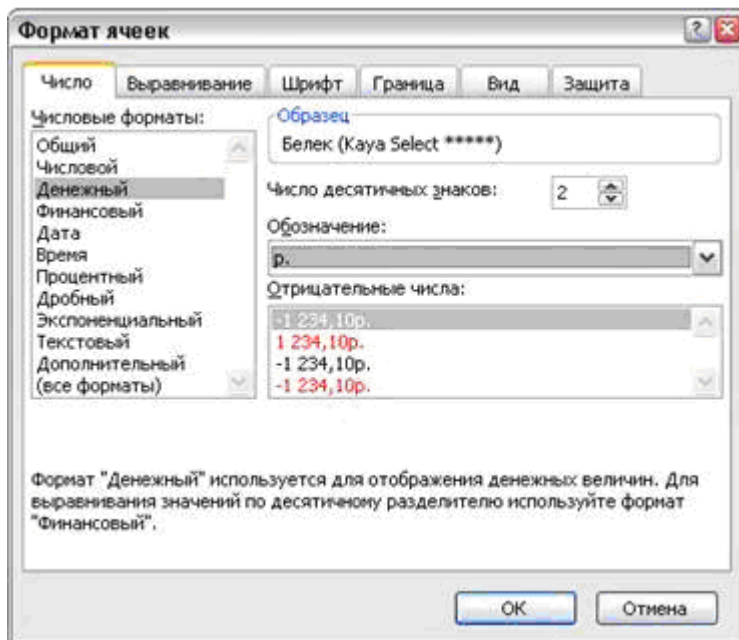


Рис. 3.15. Изменение формата ячеек

Таблица 3.4

| Наименование | Назначение |
|--------------|---|
| Число | Задается формат отображения данных в ячейке (числовой, текстовый, дата, время, и т. д.) |
| Выравнивание | Позволяет установить выравнивание текста по горизонтали и вертикали, определить угол и направление, перенос текста, объединить ячейки |
| Шрифт | Устанавливаются параметры шрифта |
| Граница | Позволяет задать параметры границ (тип, толщина, цвет) |
| Вид | Изменение цвета фона в ячейке |
| Защита | Защита листа от изменения позволяет скрыть формулу |

3.3.12.1. Форматирование числовых значений

MS Excel предоставляет к услугам пользователя более десяти видов различных форматов, описание которых представлено в табл. 3.5.

Таблица 3.5

| Формат | Описание |
|------------------|--|
| Общий | Применяется для отображения как текстовых, так и числовых значений. Числа отображаются так, как они введены. Используется по умолчанию |
| Числовой | Можно задать количество знаков после запятой, разделитель групп разрядов, вариант отображения отрицательных чисел |
| Денежный | Значения отображаются с использованием выбранного символа валюты |
| Финансовый | То же, что и формат <i>Денежный</i> , но нельзя задать вариант для отрицательных чисел |
| Дата | Предлагается 17 вариантов, определяющих отображение даты (число, месяц и год) |
| Время | Можно использовать один из 9 форматов отображения часов, минут и секунд |
| Процентный | Значение в активной ячейке умножается на сто и добавляется символ процентов |
| Дробный | Предлагается выбрать один из девяти способов отображения дробей |
| Экспоненциальный | Отображение вещественного числа в экспоненциальной форме |
| Текстовый | Отображает содержимое ячейки в том виде, в котором оно было введено, даже если содержимое представляет собой формулу |
| Дополнительный | Отображает содержимое ячейки в виде номера телефона, почтового индекса и т. д. |

Для установки формата числа необходимо выполнить следующие действия:

- выделить форматлируемую ячейку;
- с помощью команды **Формат ячеек...** из контекстного меню открыть окно **Формат ячеек** (рис. 3.16);
- из списка **Числовые форматы**: выбрать формат;
- в правой части окна выбрать тип формата (если требуется) (например, для формата *Число* можно определить количество символов в дробной части, установить (или убрать) разделитель групп разрядов);
- в поле **Образец** сравнить ожидаемый вид числа с желаемым;
- нажать кнопку **ОК**.

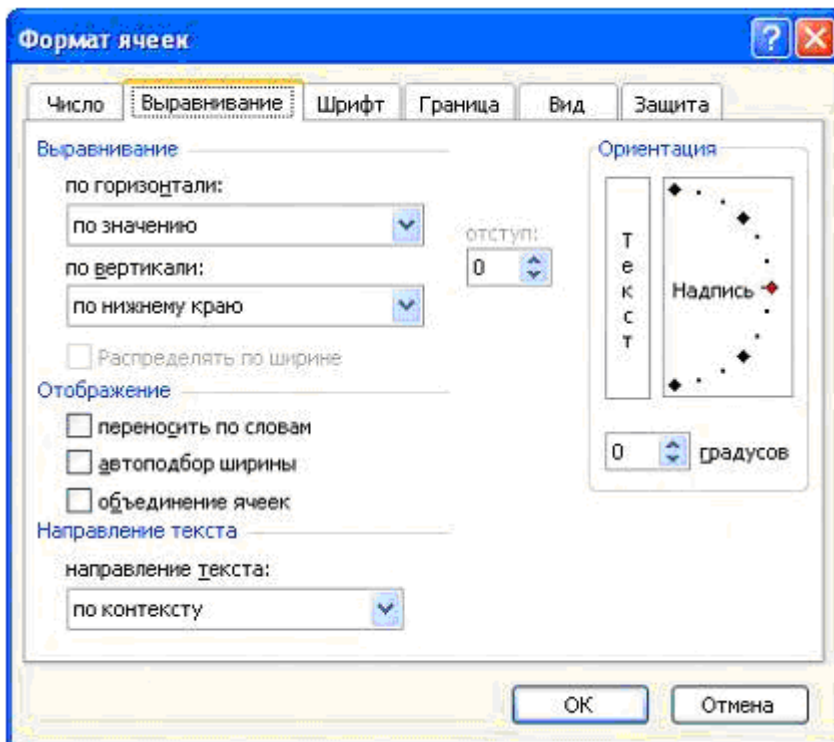


Рис. 3.16. Выравнивание в ячейке

- На панели инструментов **Форматирование** есть пять кнопок, которые упрощают применение соответствующего форматирования к выделенным ячейкам:
 - 1) *Денежный формат*: применяет разновидность формата *Денежный*, используемую по умолчанию;
 - 2) *Процентный формат*: отображает содержимое выделенных ячеек со знаками процента и без десятичной точки;
 - 3) *Формат с разделителями*: добавляет разделитель (пробел) и отображает значение с двумя дробными десятичными разрядами;
 - 4) *Увеличить разрядность*: каждый щелчок добавляет один разряд в дробной части;
 - 5) *Уменьшить разрядность*: каждый щелчок удаляет один разряд в дробной части.

3.3.12.2. Форматирование символов в ячейках

Возможность изменения атрибутов шрифта позволяет создавать легко воспринимаемые таблицы. MS Excel позволяет применить соответствующее форматирование как к содержимому всей ячейки, так и к отдельным словам или символам.

Изменить параметры шрифта можно с помощью вкладки *Шрифт* окна *Форматирование ячеек* (рис. 3.16). Практически все команды этого окна продублированы на панели инструментов *Форматирование*. Исключения составляют атрибуты *Подчеркивание* (возможно двойное и одинарное подчеркивание) и *Видоизменение*. Установка атрибутов шрифта выполняется так же, как и в MS Word.

3.3.12.3. Перенос строк и ориентация текста

MS Excel позволяет переносить длинные последовательности символов на несколько строк (вторую, третью и т. д.). Для этого необходимо:

- выделить ячейки или группу ячеек, форматирование которых необходимо изменить;
- вызвать контекстное меню и выбрать команду *Формат ячеек...* (откроется диалоговое окно *Формат ячеек*);
- перейти на вкладку *Выравнивание* (рис. 3.16);
- в группе *Отображение* установить флажок *переносить по словам*;
- щелкнуть на кнопке *ОК* внизу окна, чтобы сделанные установки вступили в силу.

В группе *Отображение* есть еще два флажка: *автоподбор ширины* и *объединение ячеек*. Первый из них позволяет разместить весь текст по ширине ячейки. При этом изменяется масштаб отображения символов в ячейке, а размер самой ячейки не изменяется. Второй – превратить несколько смежных ячеек в одну. Объединение применимо только к смежным ячейкам, которые находятся в одной строке или столбце или занимают прямоугольную область. Объединить ячейки в одну или отменить объединение можно также с помощью команды панели инструментов *Форматирование*.

Команды группы *Выравнивание* позволяют установить для текста выравнивание внутри ячейки. Выравнивание устанавливается по горизонтали и вертикали. Для этого необходимо выбрать из

выпадающего списка команду, соответствующую требуемому способу выравнивания.

Помимо прочего, можно расположить текст в ячейке под любым углом. Для изменения ориентации нужно щелкнуть на слове *Надпись* в группе *Ориентация* и с помощью мыши переместить красный маркер по (или против) часовой стрелке. Для точного указания угла наклона можно воспользоваться также счетчиком, расположенным ниже (рис. 3.16).

3.3.12.4. Использование границ и рамок

Те границы между ячейками, что видны при работе в MS Excel, на печать не выводятся и служат лишь для ориентации на листе при вводе данных. Для более понятного и наглядного оформления данных в таблице можно активно использовать границы и заливку. Чтобы оформить границы ячейки, требуется выполнить следующие действия:

- выделить нужные ячейки;
- с помощью команды контекстного меню *Формат ячеек* открыть диалоговое окно *Формат ячеек*;
- перейти на вкладку *Граница* (рис. 3.17).

В группе *Линия* можно задать тип, толщину и цвет линий, ко-

торые используются для отображения границ. В группе *Все* задаются сразу все промежуточные линии и рамка. В этой же группе находится кнопка, убирающая все границы.

Иногда возникает ситуация, когда нет необходимости указывать все границы сразу, или необходимо указать все границы, но разного типа линиями.

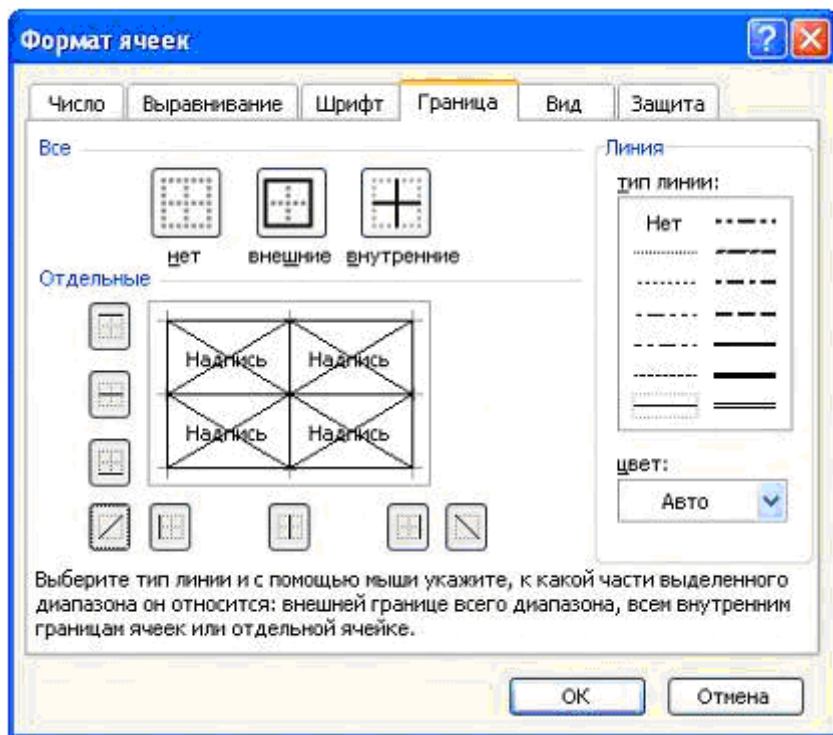


Рис. 3.17. Установка границ ячеек

В этом случае группа команд *Отдельные* поможет установить отдельно верхнюю, нижнюю, левую и правую границы, задать параметры для всех горизонтальных и вертикальных промежуточных границ и даже задать диагональные линии.

Заливка ячеек другим цветом производится с помощью палитры цветов на вкладке *Вид*.

3.3.13. Диаграммы

Для графического отображения представленных на листе данных в MS Excel используются *диаграммы*. При этом диаграммы связываются непосредственно с данными. Любое их изменение на рабочем листе сразу отображается в диаграмме. Вставить диаграмму можно с помощью кнопки *Мастер диаграмм* панели инструментов *Стандартная* или с помощью команды меню *Вставка / Диаграмма...*. В обоих случаях MS Excel запускает мастер диаграмм (рис. 3.18).

Построение диаграмм производится за четыре шага.

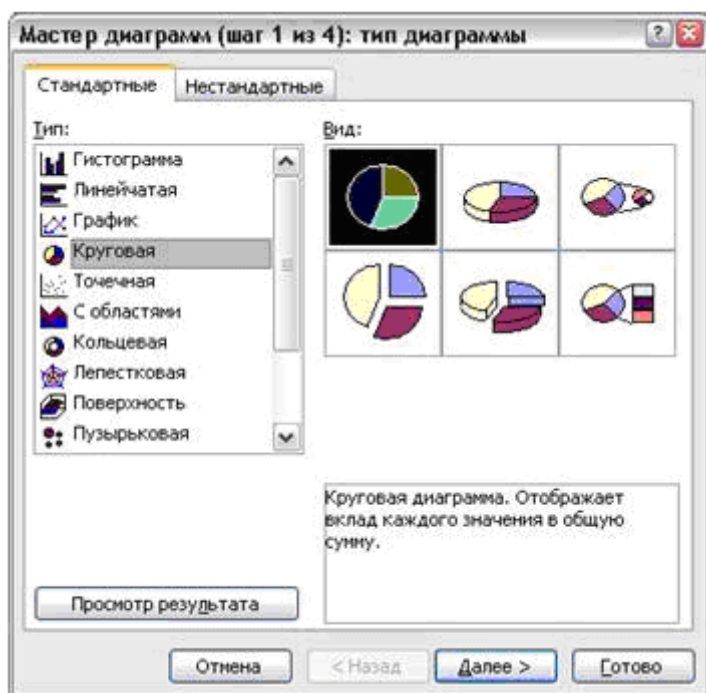


Рис. 3.18. Выбор типа диаграммы

точно большое количество диаграмм. Но наиболее привычными и используемыми типами диаграмм являются гистограмма, круговая диаграмма и график. При выборе типа диаграммы следует в первую

На первом шаге выбирается тип используемой диаграммы, как показано на рис. 3.18. В поле *Тип:* представлены различные типы диаграмм, каждая из которых содержит некоторое количество видов. Вид выбранного типа диаграммы можно определить в группе *Вид:*. Все диаграммы разбиты на две категории: стандартные и нестандартные. Для каждой из категорий определена своя вкладка. Как видно из этого рисунка, доступно доста-

очередь отдавать предпочтение тем типам, которые более наглядно представляют исходные данные.

3.3.13.1. Гистограммы

Гистограммы отображают данные в виде расположенных рядом вертикальных столбцов (или прямоугольников, или параллелепипедов). Высота их пропорциональна значениям, представляем диаграммой (рис. 3.19, а). Гистограммы целесообразно использовать в том случае, когда нужно отобразить изменение некоторого значения во времени или когда требуется представить несколько значений так, чтобы их было удобно сравнивать. Когда представляемые диаграммой значения очень отличаются (например, наименьшее равно 10, а наибольшее 10000) или когда одновременно требуется сравнить слишком много значений (т. е. диаграмма будет состоять из очень большого числа узких столбцов), гистограммы использовать не рекомендуется.

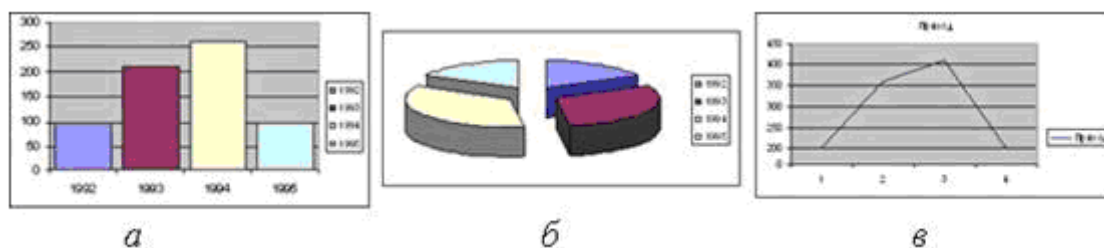


Рис. 3.19. Типы диаграмм:
а – гистограмма; б – круговая диаграмма; в – график

3.3.13.2. Круговые диаграммы

Это один из наиболее популярных типов диаграмм (рис. 3.19, б). Подобные диаграммы имеют вид круга, представляющего нечто целое, разделенного на секторы, представляющие части целого. На рисунке показан объемный вариант разрезанной круговой диаграммы. Круговая диаграмма будет полезна, когда имеется несколько значений, образующих нечто целое, и требуется показать, каков вклад каждого из них в общую сумму. Если имеется множество небольших значений, вклад каждого из которых в общую сумму достаточно мал, сектора круговой диаграммы будут чрезвычайно малы и диаграмма будет трудной для восприятия.

3.3.13.3. Графики

График показывает изменение значения (значений) во времени или от какого-то аргумента (рис. 3.19, в). Таким образом, можно показать тенденцию, которая иначе (при просмотре таблицы с числами) была бы неочевидна.

После того как тип диаграммы выбран, можно приступить к вводу отображаемых на ней данных, для чего следует нажать кнопку *Далее*>, после щелчка на которой появляется второе окно мастера диаграмм (рис. 3.20).

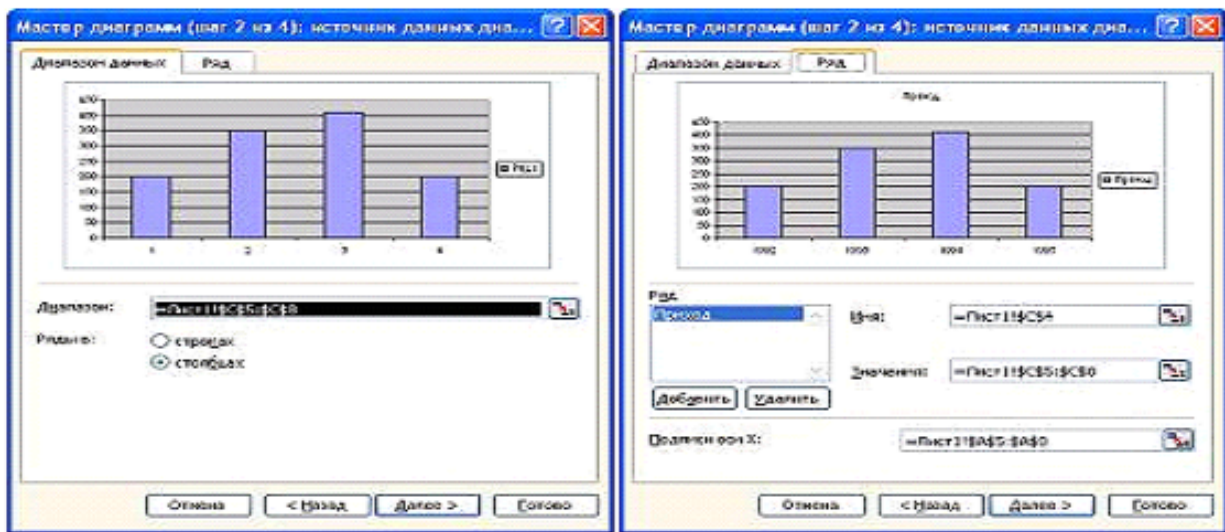


Рис. 3.20. Исходные данные диаграммы

В окне две вкладки: *Диапазон данных* и *Ряд*. Если перед построением диаграммы таблица со значениями была выделена, то в верхней части окна будет отображен примерный вид диаграммы. В поле *Диапазон:* будет записана ссылка на ячейки с данными. Если данные не известны, тогда необходимо:

- нажать на кнопку справа от этого поля (окно мастера диаграмм свернется);
- выделить мышью ячейки, содержащие данные для построения диаграммы;
- развернуть окно мастера диаграмм.

Выбранные для отображения, на диаграмме данные называются рядами. На вкладке *Ряд* можно посмотреть, сколько и с какими данными созданы ряды. Здесь есть возможность добавить новый ряд или удалить ненужный, а также задать имя для каждого ряда с помощью поля *Имя*. Имя ряда отображается в *легенде*. Легенда –

это пояснения к диаграмме, заключенные в рамку. Самое нижнее поле окна позволяет определить подписи оси X . Нумерация подписей оси X зависит от типа используемой диаграммы. Для гистограмм, например, MS Excel нумерует значения по порядку, начиная с 1. Пользователь имеет возможность заменить эти подписи как постоянными значениями, так и ссылками на ячейки какого-либо листа книги. На рис. 3.20 в качестве подписей оси X указаны годы с 1992 по 1995.

После того, как введены все исходные данные и заданы их параметры, нажатием кнопки *Далее* > следует перейти к третьему

шагу мастера диаграмм, связанному с вводом различного рода пояснений к диаграмме и отображаемым на ней данным (рис. 3.21), которые принято называть параметрами диаграммы.

Окно *Параметры диаграммы* может содержать до шести вкладок, количество которых зависит от

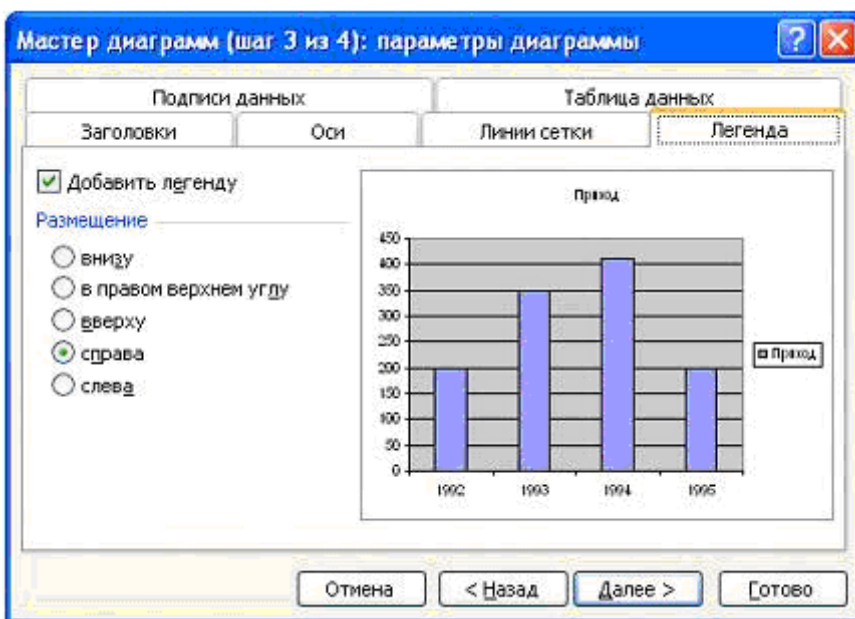


Рис. 3.21. Параметры диаграммы

выбранного типа диаграммы. В табл. 3.6 указано назначение всех этих вкладок.

На четвертом (последнем) шаге пользователю предлагается выбрать лист, где будет размещена создаваемая диаграмма. Этот шаг не является обязательным. Если сразу на третьем шаге нажать кнопку *Готово*, то диаграмма будет вставлена на текущий лист.

Каждый элемент диаграммы имеет набор собственных свойств и параметров. Так, если кликнуть правой кнопкой мыши на элементе *Легенда* (область, в которой отображаются подписи категорий), то появится другое контекстное меню, которое позволит изменять настройки этого элемента. Например, команда *Формат легенды...*

позволяет изменять размещение данного элемента, используемый для его отображения шрифт и цвета.

Таблица 3.6

| Вкладка | Назначение |
|----------------|--|
| Заголовки | Можно создать заголовки для обозначения каждой оси диапазона, а также всей диаграммы в целом |
| Оси | С помощью соответствующих флажков включаются и выключаются оси X и Y |
| Линии сетки | Можно задать отображение линий метки для осей X и Y независимо друг от друга |
| Легенда | Позволяет дополнить диаграмму легендой и задать место ее расположения |
| Подписи данных | Можно дополнить диаграмму подписями для лучшего понимания назначения элементов диаграммы. В подписи можно включить имя ряда, имя категории, значение, долю |
| Таблица данных | Имеется возможность добавить таблицу данных в диаграмму. Имеет смысл только в том случае, если диаграмма выполнена на отдельном листе |

При формировании нужного облика диаграммы пользователь может задействовать свойства каждого элемента, изменять путем перетаскивания его местоположение, размер и т. д., создавая диаграммы, разительно отличающиеся друг от друга.

3.3.14. Печать содержимого книги

По своей природе таблицы имеют тенденцию расти во все стороны, а данные вряд ли разместятся идеально для вывода их на печатных страницах без некоторого дополнительного форматирования. В связи с этим по завершении ввода данных следует обязательно войти в режим предварительного просмотра (нажать кнопку *Предварительный просмотр* панели инструментов *Стандартная*), чтобы увидеть, как введенная информация будет выглядеть на странице при печати. Для выхода из режима предварительного просмотра нужно нажать кнопку *Заккрыть*.

Подготавливая лист к предварительному просмотру, MS Excel

определяет *область печати*. Если эта область не была определена заранее, то предполагается, что нужно печатать все ячейки с адреса A1 и до самых отдаленных ячеек, в которых содержатся данные или было изменено форматирование. Если требуется вывести на печать не все данные листа, а только какую-то их часть, причем делать это приходится регулярно, то можно один раз установить на листе область печати и в дальнейшем использовать ее. Каждый лист может иметь свою область печати. Для определения области печати вначале необходимо выделить нужный диапазон, а затем выбрать команду меню *Файл / Область печати / Задать*. После этого данный диапазон будет использоваться по умолчанию как область печати каждый раз, когда инициализируется печать или осуществляется предварительный просмотр. Отменить область печати можно командой меню *Файл / Область печати / Убрать*.

Изменить параметры страницы, используемой для печати, можно с помощью команды меню *Файл / Параметры страницы....* При этом на экране появится окно, показанное на рис. 3.22.

Наиболее часто используемыми параметрами здесь являются поля – отступы от краев бумажного листа до области, в которой будет печататься выводимый текст. Изменить поля можно на одноименной вкладке показанного на рис. 3.22 диалогового окна. Еще

одним полезным параметром является ориентация страницы: книжная (когда высота больше ширины) или альбомная (ширина листа больше высоты). Изменить ориентацию страницы можно на вкладке *Страница*.

После того, как данные подготовлены и своим внешним представлением удовлетворяют требованиям пользователя, можно

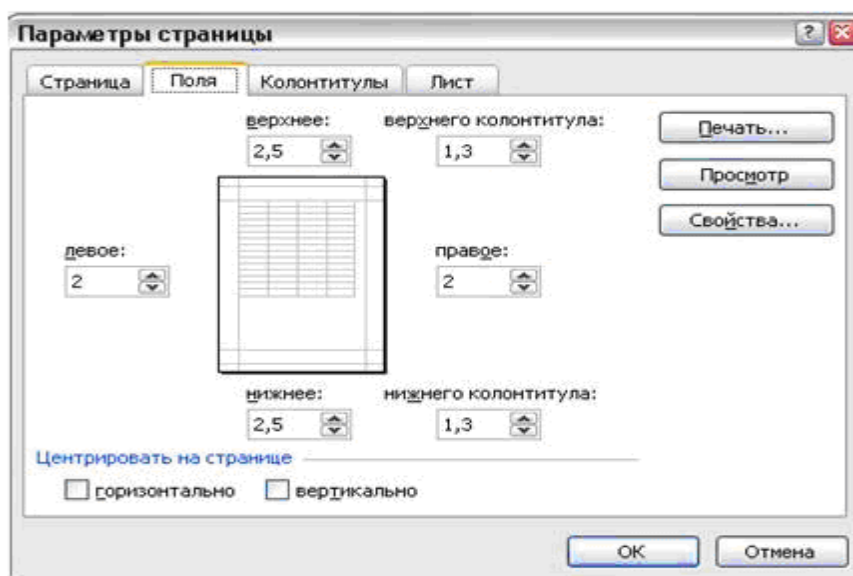


Рис. 3.22. Изменение параметров страницы

непосредственно отправлять документ на печать. Это делается либо нажатием кнопки *Печать* на панели инструментов *Стандартная*, либо вызовом команды меню *Файл / Печать...*. В первом случае все страницы будут сразу отправлены на печать, а во втором – появится диалоговое окно *Печать*, в котором можно будет уточнить (настроить) параметры печати.

В окне *Печать* можно определить принтер для печати (в случае, когда в операционной системе установлено несколько принтеров), число копий и диапазон печатаемых страниц. Кроме этого, в качестве параметра печати можно указать параметр *выделенный диапазон*. В этом случае, даже если на листе задана область печати, на принтер будут выводиться выделенные ячейки листа.

Кнопка *Просмотр* позволяет перейти в режим предварительного просмотра выводимого на печать рабочего листа. После выхода из режима предварительного просмотра в данном случае происходит возврат к окну *Печать*. Для печати данных следует нажать кнопку *ОК*.

3.3.15. Коды возможных ошибок при работе с формулами в MS Excel

Таблица 3.7

| Код ошибки | Причина возникновения | Предположения и рекомендации |
|-------------------|--|--|
| ##### | Столбец недостаточно широк или дата и время являются отрицательными числами | Расширить столбец, проверить значения даты и времени |
| #ДЕЛ/0! | Попытка деления на ноль или на значение из пустой ячейки | Проверить делитель в формуле, что он не равен нулю и не ссылается на пустую ячейку |
| #Н/Д | Значение недопустимо для использования формулой в качестве аргумента | Проверить совместимость типов аргументов функции и указанных значений |
| #ИМЯ? | Формула содержит текст, не являющийся ни именем функции, ни именем диапазона | Вероятно, допущена опечатка в имени функции или имени диапазона |

| Код ошибки | Причина возникновения | Предположения и рекомендации |
|------------|--|--|
| #ПУСТО! | Задано пересечение двух областей, не имеющих общих ячеек | Изменить ссылки таким образом, чтобы диапазоны пересекались |
| #ЧИСЛО! | Значение слишком велико/мало, мнимое или не обнаружено | Как правило, возникает при некорректном использовании функций |
| #ССЫЛКА! | Формула содержит неверную ссылку | Возможно, данные, на которые имеется ссылка в формуле, были удалены |
| #ЗНАЧ! | Формула содержит аргумент недопустимого типа | Возможно, формула пытается выполнить арифметическое или логическое действие, например, над текстом |

3.3.16. Примеры

Пример 1. Найти сумму любых двух чисел ($c = a + b$).

Так как в формулах используют ссылки на ячейки, то для решения данной задачи понадобятся три ячейки. В первую ячейку (пусть это будет ячейка A1) поместим значение переменной a . Во вторую (B1) – значение переменной b , а в третью поместим результат (ячейка C1).

Для решения задачи предлагается следующий алгоритм:

- в ячейку A1 поместить значение переменной a (например, число 3), в ячейку B1 – значение переменной b (например 5);
- в ячейку C1 записать выражение (формулу) =A1+B1 и нажать клавишу **Enter**.

В результате в ячейке C1 появится число восемь. Если изменить значения в ячейках A1 и B1, то MS Excel автоматически вычислит новое значение и в ячейке C1.

Возможен второй способ записи выражения в ячейку C1 – с использованием мыши. В режиме ввода формул выделение ячейки

с помощью мыши автоматически добавляет в формулу ее адрес.

Пример 2. Найти сумму любых двух чисел ($c = a + b$).

В данном случае запись в ячейку C1 выражения =A1+B1 потребует выполнения следующих действий:

- выделяем ячейку C1;
- нажимаем клавишу с символом =;
- с помощью мыши выделяем ячейку A1 (при этом в ячейке C1 после знака равенства появится адрес A1);
- вводим символ +;
- с помощью мыши выделяем ячейку B1 (ее адрес также появляется в ячейке C1 после знака +);
- нажимаем клавишу **Enter**.

Возможно, этот способ определения ссылок на ячейки с данными покажется более трудоемким, но в случае работы с большими формулами и большим объемом данных это существенно упрощает ввод. По крайней мере, пользователю нет необходимости запоминать все адреса ячеек, можно просто на них «показать».

Пример 3. Требуется рассчитать значение функции $f(x) = x^3 + 5x^2 + 2$ для любого значения x .

Для решения данной задачи понадобятся две ячейки. Положим, что в ячейку A1 пользователь запишет значение переменной x , а в ячейке B1 должен отображаться результат. Тогда в ячейку B1 должна быть введена соответствующая формула. Для возведения в степень используется символ ^, а для умножения – *. В этом случае в ячейку B1 вводим следующую формулу: =A1^3-5*A1^2+2. При создании формулы была использована ссылка на ячейку A1, в которой помещено значение аргумента.

Пример 4. Найти значения функции $f(x) = x^c - x + 2$, где x на отрезке от 1 до 5 изменяется с шагом 0,5 и c – некоторая константа.

В первую очередь заполняем ячейки, в которых будут храниться значения аргумента. Для этого можно использовать автозаполнение:

- в ячейку A1 помещаем значение 1;
- так как переменная x изменяется с шагом 0,5, то в ячейку A2 поместим число 1,5 (в качестве разделителя целой и дробной части используется запятая);

- выделяем ячейки A1 и A2, и с помощью маркера заполнения в ячейки с A3 по A9 заносим остальные значения аргумента функции.

Далее в ячейку C1 помещаем значение константы c (например 2) и заполняем ячейки с B1 по B9. В первую из этих ячеек поместим выражение $=A1^{\wedge}C1-A1+2$, в ячейку B2 помещаем точно такое же выражение, но данные будем брать из ячейки A2: $=A2^{\wedge}C1-A2+2$. И так далее для каждой ячейки. В последнюю (девятую ячейку) введем $=A9^{\wedge}C1-A9+2$.

В ячейках с B1 по B9 будут отображаться значения функции $f(x) = x^2 - x + 2$ для разных значений аргументов. Если изменить значение ячейки C1 на 3, то будет использоваться формула $f(x) = x^3 - x + 2$, и т. д.

Пример 5. Найти значения функции $f(x) = x^c - x + 2$, где x на отрезке от 1 до 5 меняется с шагом 0,5 и c – некоторая константа.

Ячейки, содержащие значения аргумента, заполняем так же, как и раньше, с использованием автозаполнения:

- в ячейку A1 помещаем значение 1;
- так как переменная x изменяется с шагом 0,5, то в ячейку A2 поместим число 1,5;
- выделяем ячейки A1 и A2 и с помощью маркера заполнения в ячейки с A3 по A9 заносим остальные значения аргумента функции.

В ячейку C1 поместим значение константы c (например 2), ячейки с B1 по B9 заполняем следующим образом:

- в ячейку B1 поместим выражение $=A1^{\wedge}C1-A1+2$;
- помещаем курсор после ссылки C1 и нажимаем клавишу **F4**, при этом формула примет вид $=A1^{\wedge}\$C\$1-A1+2$;
- нажимаем клавишу **Enter** (в ячейке B1 появляется значение функции для $x = 1$);
- выделяем ячейку B1 и с помощью маркера заполнения копируем формулу в ячейки с B2 до B9.

3.3.17. Контрольные вопросы

- 1) Как называется приложение, предназначенное для обработки таблиц?
- 2) Перечислите приложения, входящие в пакет MS Office.
- 3) Назовите способы переименования рабочего листа.
- 4) Перечислите способы адресации ячеек.
- 5) Какие способы применения функций рабочего листа MS Excel вам известны?
- 6) Что можно записать (ввести) в отдельную ячейку рабочего листа?
- 7) Какому десятичному числу соответствует число $1,1E+07$, отображенное в ячейке таблицы?
- 8) Что собой представляет легенда диаграммы MS Excel?
- 9) Что получится в результате вычисления по формуле, содержащей ссылку на ячейку, содержимое которой очищено?
- 10) Какой тип адресации использован в выражении $=\sin(A1)$?
- 11) Какой тип адресации использован в выражении $=\$A\1 ?
- 12) Каким образом задается диапазон ячеек в MS Excel?
- 13) В какую категорию входит функция ЕСЛИ?
- 14) При какой ситуации в ячейке появляется константа ошибки #ЗНАЧ!?

3.4. Текстовый процессор MS Word

Текстовые документы, создаваемые с помощью персонального компьютера, условно можно разделить на две группы – простые и комплексные. Простые документы представляют собой форматированный текст. Комплексные документы кроме текста содержат объекты иной природы (чертежи, рисунки, формулы, таблицы, объекты мультимедиа и прочие). Эти документы могут создаваться с помощью специальных программных средств, которые называют текстовыми процессорами. В настоящее время в нашей стране широко применяются текстовые процессоры семейства MS Word из состава пакета MS Office. В перспективе по мере развития и совершенствования технического и программного обеспечения неизбежно появятся и новые версии текстовых процессоров.

3.4.1. Главное окно MS Word

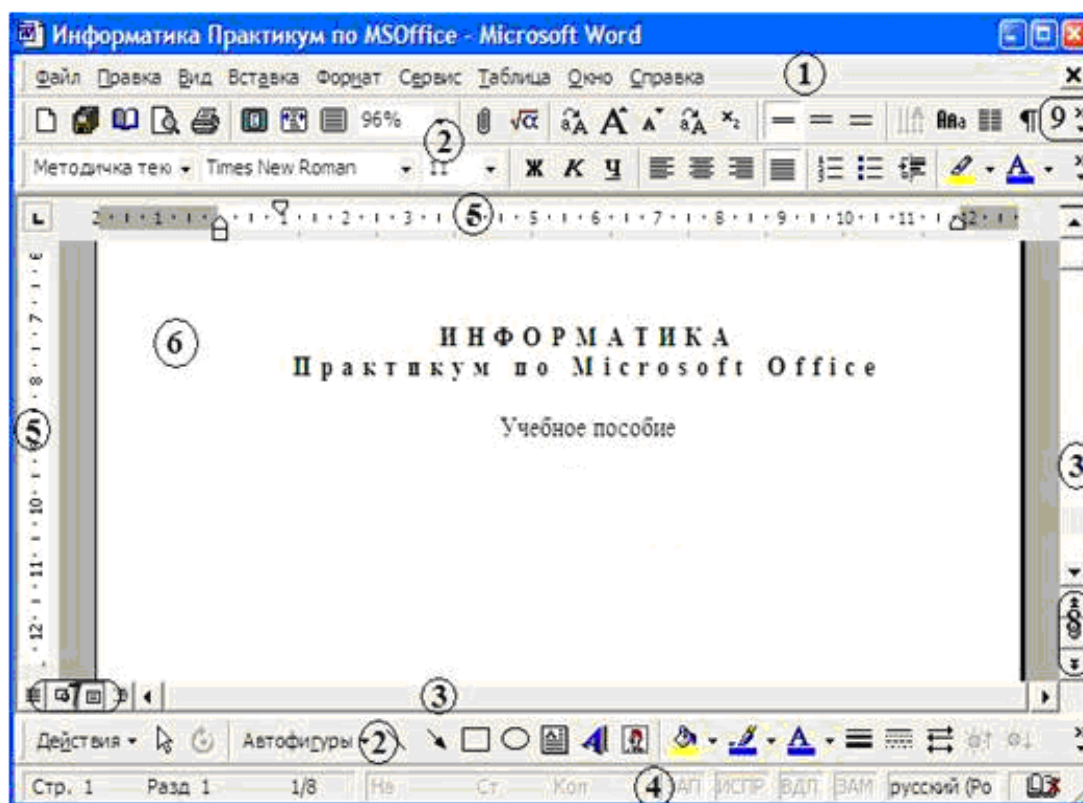


Рис. 3.23. Главное окно MS Word

После запуска MS Word на экране появляется окно, показанное на рис. 3.23, в котором можно выделить следующие элементы управления, обозначенные на рисунке соответствующими цифрами.

- 1) *Горизонтальное меню* предназначено для доступа ко всем функциям и командам.
- 2) *Панели инструментов* – средство для быстрого доступа к часто используемым командам MS Word. Команды на панелях инструментов отображаются в виде кнопок с пиктограммами (рисунками) или текстом.
- 3) *Полосы горизонтальной и вертикальной прокрутки (скроллинг)* – средства перемещения по документу.
- 4) *Строка состояния* служит для отображения дополнительной информации (текущая страница, раздел документа, общее количество страниц, режимы ввода текста и др.).
- 5) *Горизонтальная и вертикальная линейки*. С их помощью можно изменить отступы, поля, ширину колонок, высоту строк и ширину столбцов таблицы.
- 6) *Область отображения документа*. В этой части выводится содержание рабочего документа в выбранном режиме.

7) *Область переключения* между возможными режимами отображения документа MS Word:

- *Обычный* – отображается только содержательная часть документа без каких-либо элементов оформления; этот режим наиболее удобен при работе с большими документами;
- *Web-документ* – режим предназначен для создания и просмотра электронных публикаций в World Wide Web; в этом режиме понятие «печатная страница» не имеет смысла;
- *Разметка страницы* – отображаемые на экране страницы полностью соответствуют печатным страницам;
- *Структура* – позволяет видеть структуру документа, а также перемещать, копировать и реорганизовывать текст посредством перетаскивания заголовков.

8) *Кнопки* для постраничного перемещения по документу имеют следующее назначение:

- *Страница вверх*;
- *Выбор объекта перехода* позволяет перейти к произвольному объекту, размещенному в документе;
- *Страница вниз*.

9) *Значок*, появляющийся, когда размеры окна не позволяют отобразить все размещенные на панели инструментов кнопки. После нажатия мышью на данном значке открывается дополнительное меню, в котором отображены остальные кнопки данной панели инструментов.

В заголовке окна указывается имя открытого документа (файла). Например, имя документа, показанного на рис. 3.23, Информатика Практикум по MS Office.

3.4.2. Настройки и параметры MS Word

Для изменения настроек пользовательского интерфейса используются команды меню *Вид*. С помощью этого меню можно:

- изменять режимы отображения документа, о которых говорилось ранее;
- включать или отключать отображение линеек (строка *Линейка*);

- изменять масштаб отображения документа (строка *Масштаб...*);
- показать или убрать панели инструментов;
- отображать колонтитулы.

Масштаб отображения задается либо в процентах, либо может быть одним из следующих значений:

- *по ширине текста* – текст выводится на ширину окна;
- *по ширине страницы* – помимо текста в ширину окна размещается форматирование страницы (поля и т. д.);
- *целая страница* – страницы отображаются такого размера, чтобы полностью поместились в окне;
- *несколько страниц* – на экране помещаются несколько страниц целиком.

Пользователь может выбрать как одно из заданных значений масштаба, так и ввести собственный масштабный коэффициент в процентах.

Колонтитулы – области над и под текстом (соответственно верхний и нижний колонтитулы) на странице, в которых может отображаться произвольная информация: текст, рисунки и т. д. Это области ограниченного размера, но их содержимое отображается на каждой странице документа. Пример создания колонтитула представлен на рис. 3.24.

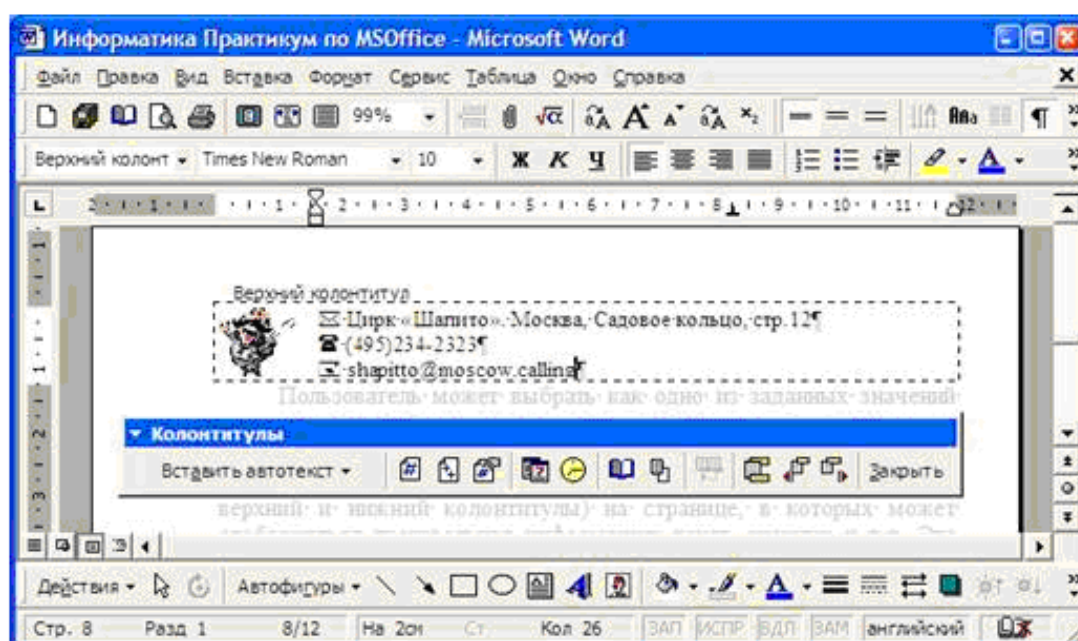


Рис. 3.24. Создание колонтитулов

Полезной функциональностью любого приложения является удобное размещение наиболее часто используемых команд. В пакете программ MS Office таким средством являются панели инструментов.

Текстовый процессор MS Word имеет набор панелей инструментов, в которых по типу сгруппированы наиболее часто используемые команды. Каждая панель имеет имя, отражающее, как правило, тип размещаемых на ней команд. Например, на панели инструментов *Рисование* размещены кнопки с командами, позволяющими создавать графические примитивы и выполнять некоторые действия над ними (задавать цвета линий, тип заливки и др.). На панели инструментов *Форматирование* собраны кнопки с командами изменения свойств текста, абзаца, списков и т. д.

Отобразить ту или иную панель можно следующим образом: выбрать пункт меню *Вид* и в нем – подменю *Панели инструментов*, в котором будут отображаться названия доступных панелей. Чтобы показать/скрыть панель, нужно кликнуть мышью на ее имени. Это же действие можно выполнить и другим способом – достаточно кликнуть правой кнопкой мыши в области панелей (область 2 на рис. 3.23) и затем в контекстном меню выбрать имя необходимой панели.

Для более «тонкой» настройки команд панелей инструментов в MS Word можно воспользоваться

меню *Сервис / Настройка...* При этом на экране появляется диалоговое окно, изображенное на рис. 3.25.

Вкладка *Панели инструментов* позволяет показать (скрыть) те или иные панели, т. е. ее назначение аналогично рассмотренным

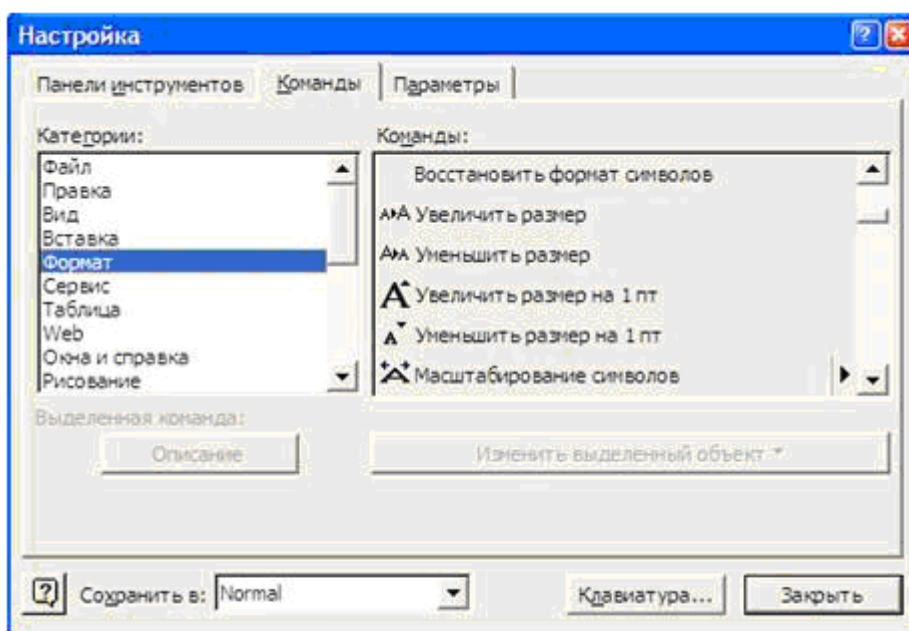


Рис. 3.25. Настройка панелей инструментов

ранее способам отображения панелей инструментов в окне MS Word. Исключение составляет возможность создания собственной панели инструментов, на которую пользователь может поместить команды, используемые именно им.

Вкладка *Команды* предназначена для добавления команд на панели инструментов. В поле *Категории*: выберите категорию для команды, которую хотите вынести на панель инструментов. Затем в поле *Команды*: найдите команду и перетащите ее с помощью мыши в нужное место на панель инструментов.

Вкладка *Параметры* позволяет настраивать некоторые свойства панелей инструментов, такие как использование всплывающих подсказок, отображение крупных значков и др.

Если внимательно посмотреть на рис. 3.25, то в правом нижнем углу можно заметить кнопку *Клавиатура...*, назначение которой – позволить пользователю сопоставить некоторой команде MS Word сочетание клавиш. Последующие вызовы такой команды будут осуществляться при нажатии на заданные клавиши, независимо от того, вынесена ли команда на панель инструментов или нет.

На рис. 3.26 представлено окно настройки сочетаний клавиш

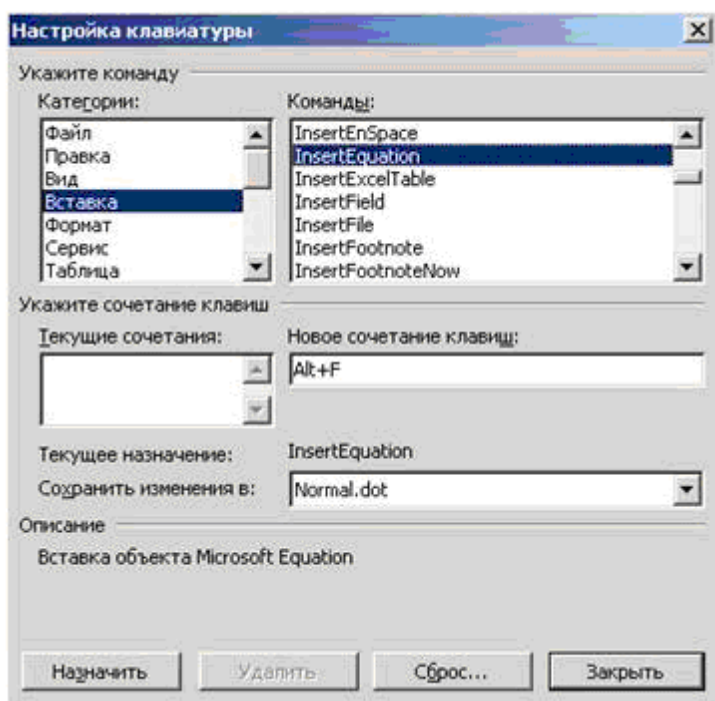


Рис. 3.26. Настройка клавиатуры

автоматически отобразится в строке ввода. На каждую команду может быть настроено несколько сочетаний клавиш. Все они

MS Word. В примере команде *InsertEquation* (вставка формулы) категории *Вставка* назначается сочетание клавиш **Alt + F**. Если теперь нажать кнопку *Назначить*, то в дальнейшем при работе в MS Word сочетание **Alt + F** будет открывать редактор формул.

Для того чтобы выполнить такую настройку клавиатуры, достаточно установить курсор в поле ввода *Новое сочетание клавиш*: и нажать нужное сочетание, после чего оно

отображаются в поле *Текущие сочетания*:. Описание выделенной команды показано в соответствующей области данного окна.

Еще одно средство управления интерфейсными параметрами MS Word доступно с помощью меню *Сервис / Параметры...* На вкладке *Вид* представлены все возможные настройки (рис. 3.27):

- интерфейсные элементы (закладки, строка состояния, всплывающие подсказки, полосы прокрутки, анимация текста и др.);
- знаки форматирования (пробелы, табуляции, поля и т. п.)
- параметры отображения в режиме разметки документа (отображать рисунки, поля, привязку объектов и т. д.).

На вкладке *Общие* наиболее важным является параметр *Единицы измерения*:, который позволяет указать, в каких единицах будут измеряться расстояния на странице документа в сантиметрах или в дюймах.

Чтобы обезопасить себя от потери введенных в документ данных при неожиданных сбоях сети электропитания или при возникновении внештатных ситуаций в операционной системе, в MS Word добавлено средство, позволяющее автоматически

сохранять сделанные изменения через заданные промежутки времени. Изменить интервалы автосохранения документа можно на вкладке *Сохранение*.

Как видно из рис. 3.27, MS Word обладает достаточно большим набором разнообразных параметров, основные из которых были рассмотрены выше. Для изучения всех параметров можно воспользоваться дополнительной литературой. В случае, если при работе с MS Word у пользователя возникают те или иные вопросы,

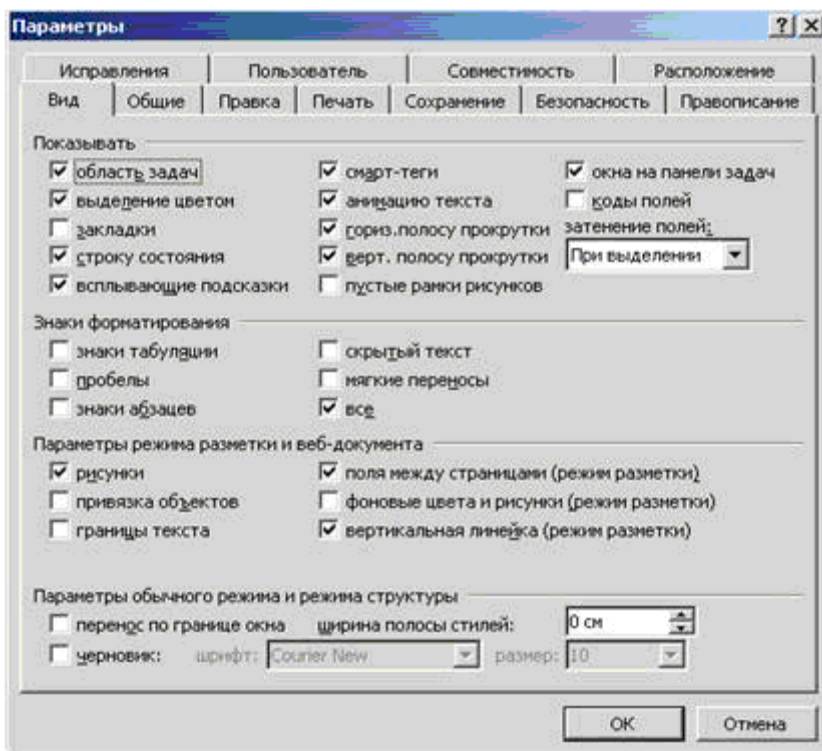


Рис. 3.27. Параметры MS Word

то ответы можно найти в справочной системе текстового процессора, доступной после нажатия клавиши **F1**.

3.4.3. Создание и сохранение документов

Создать документ MS Word можно несколькими способами.

- 1) Открыть нужную папку в проводнике и, кликнув правой кнопкой мыши, в контекстном меню выбрать команду *Создать / Документ Microsoft Word*. Затем ввести имя файла.
- 2) Нажать на кнопку *Создать* панели инструментов *Стандартная*.
- 3) Использовать команду меню MS Word *Файл / Создать...*

Если первые два способа выполняют только непосредственно создание документа, то последний способ является самым гибким инструментом, так как позволяет использовать различные дополнительные параметры.

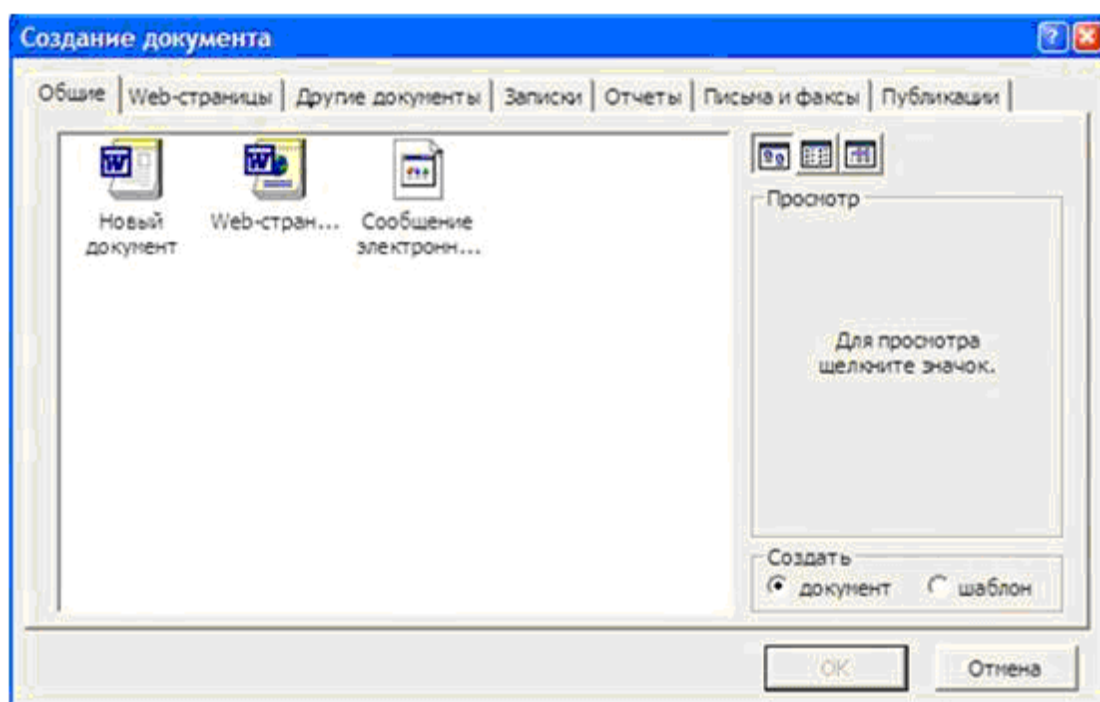


Рис. 3.28. Диалоговое окно создания документов MS Word

После выбора команды меню *Файл / Создать...* на экране появляется диалоговое окно, изображенное на рис. 3.28. В MS Word 2003 при выборе этой команды справа открывается дополнительная панель (рис. 3.29), с помощью которой можно вызвать показанное на рис. 3.29 окно, выбрав команду *На моем компьютере...* из разде-

ла *Шаблоны*. В правом нижнем углу окна *Создание документа* указывается тип создаваемого документа: документ или шаблон.

Как уже упоминалось ранее, шаблон – специальный тип файлов MS Word, имеющий в имени расширение *.dot*. Данный тип

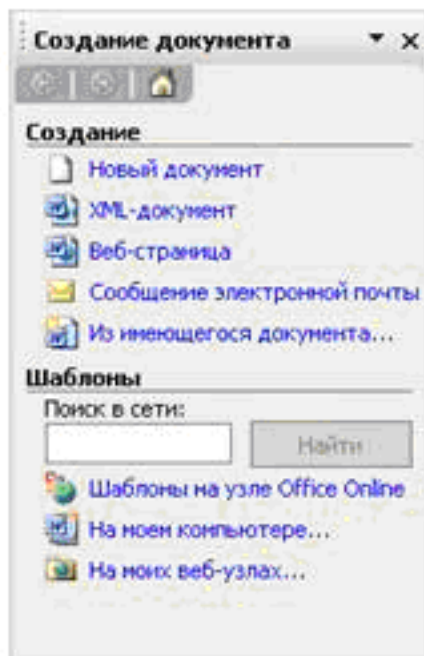


Рис. 3.29. Создание документа

файлов предназначен для хранения как различного рода пользовательских настроек, так и, при необходимости, некоторого отформатированного текста. Любой документ MS Word создается на базе того или иного шаблона (по умолчанию используется шаблон *Normal.dot*, который соответствует элементу *Новый документ* в представленном на рис. 3.29 окне), при этом все настройки и текст шаблона переносятся в документ с точно такими же параметрами и форматированием, с какими он введен в шаблоне.

На вкладках диалогового окна (рис. 3.28) располагаются имена шаблонов, сгруппированные по типу создаваемого с его помощью документа. Так, на вкладке *Записки* расположены различные варианты оформления служебной записки, на вкладке *Отчеты* – варианты оформления отчетов и т. д.

При выборе того или иного шаблона его оформление загружается в область просмотра. Для создания нового документа на основе выбранного шаблона достаточно нажать кнопку *ОК*. Далее можно вносить в документ любые необходимые изменения.

Для сохранения документа или шаблона на диске в MS Word существуют две команды меню:

- 1) *Файл / Сохранить (Ctrl + S)* сохраняет документ;
- 2) *Файл / Сохранить как...* перед сохранением позволяет присвоить или изменить файлу имя, для чего используется диалоговое окно, показанное на рис. 3.30.

При использовании команды *Файл / Сохранить как...* можно указать не только новое имя файла, но также его новое местораспо-

ложение на диске и формат. Ранее говорилось, что файлы, содержащие документы MS Word, имеют в имени расширение .doc. Однако в некоторых случаях (например когда требуется перенести файл на другой ПК, но нет уверенности, что на нем установлен MS Word) удобно использовать другой формат документа. Например, формат RTF, который поддерживается многими текстовыми процессорами, в том числе работающими под управлением отличных от MS Windows операционных систем. Для сохранения документа в ином формате следует указать этот формат в области *Тип файла*: (рис. 3.30), выбрать папку для сохранения, ввести имя файла и нажать кнопку *Сохранить*. Если для сохранения документа требуется создать новую папку, то это можно легко выполнить командой *Создать папку* рассматриваемого диалогового окна.

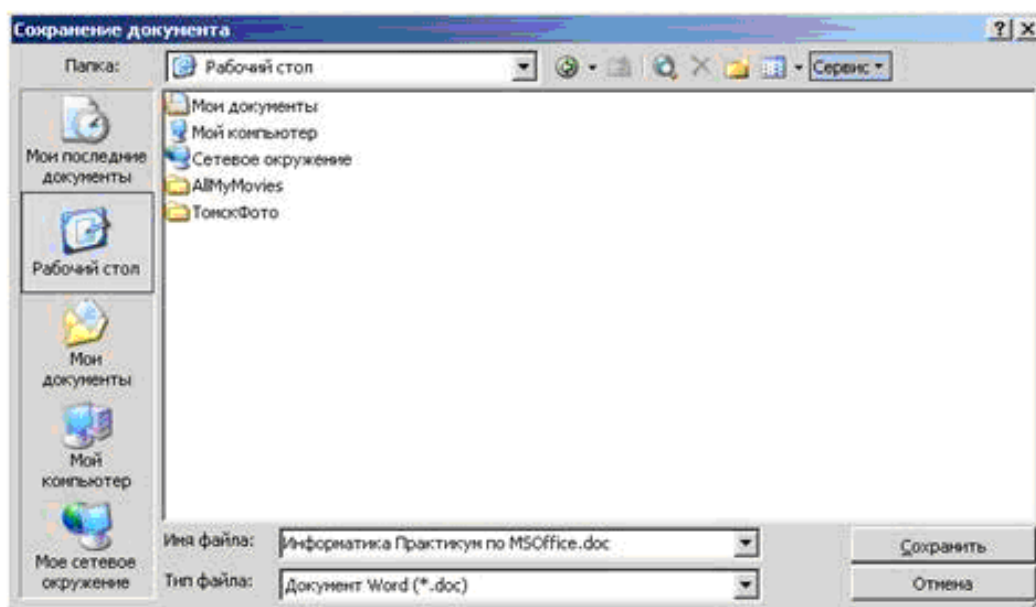


Рис. 3.30. Сохранение документов в MS Word

Для того чтобы открыть документ MS Word, достаточно дважды кликнуть мышью на имени файла документа. Однако если MS Word уже запущен, новый документ можно открыть вызвав команду меню *Файл / Открыть...* или нажав сочетание клавиш **Ctrl + O**. При этом на экране появится диалоговое окно *Открытие документа*, которое функционально и внешне очень похоже на окно *Сохранение документа*, представленное на рис. 3.30. Далее следует найти нужный каталог, выбрать интересующий файл и нажать кнопку *Открыть*, после чего указанный файл будет загружен в MS Word.

3.4.4. Параметры страницы

Страницы документа MS Word имеют такие параметры, как поля, ориентация и т. п. Изменение этих параметров возможно через команду меню *Файл / Параметры страницы....* При этом на экране появляется диалоговое окно, изображенное на рис. 3.31.

Наиболее важными параметрами здесь являются:

- поля, определяющие расстояния от края страницы до границ текста сверху, снизу, слева и справа;
- ориентация страницы: книжная или альбомная;
- размер бумаги, который задает в сантиметрах высоту и ширину страницы; здесь можно выбрать либо один из стандартных размеров, либо ввести высоту и ширину самостоятельно.

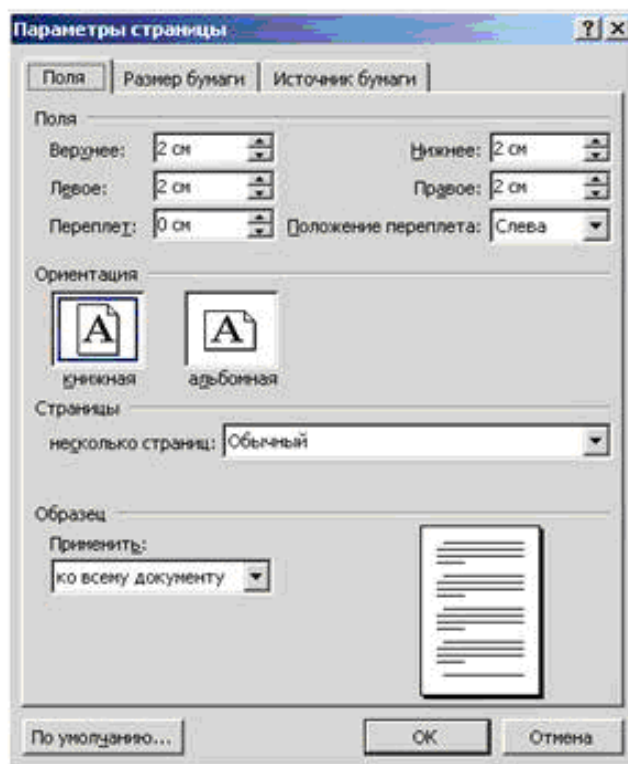


Рис. 3.31. Параметры страницы

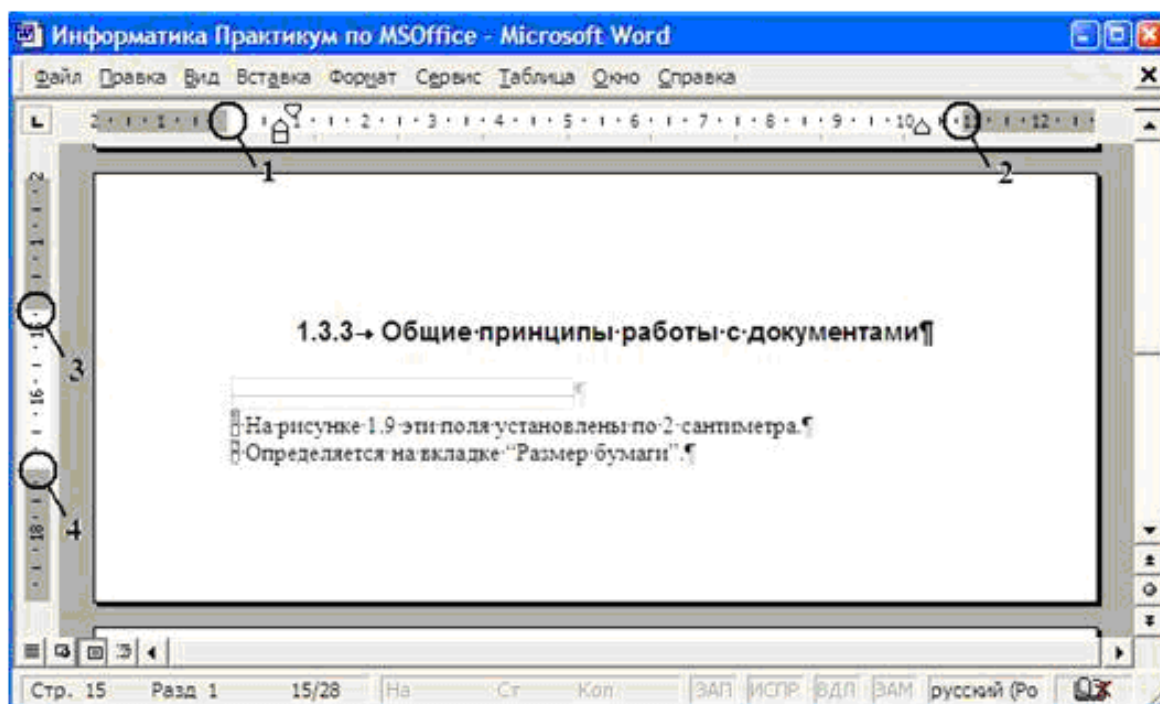


Рис. 3.32. Изменение полей с помощью линеек

По умолчанию в MS Word создается документ с форматом страниц А4 (210 × 297 мм), который используется наиболее часто для оформления текстовой документации.

Параметры страницы можно изменять не только с помощью диалогового окна *Параметры страницы*, но и с помощью горизонтальной и вертикальной линеек. На рис. 3.32 показаны области линеек, соответствующие полям: 1 – левому, 2 – правому, 3 – верхнему, 4 – нижнему. Для изменения этих значений достаточно зацепить курсором мыши области, обведенные кругом, и перемещать мышью. Размеры соответствующих полей будут меняться.

Недостаток данного способа в том, что установить точное значение поля достаточно проблематично. Можно, конечно, использовать перемещение курсора мыши при нажатой клавише **Alt**, что позволит задать значения соответствующего поля с точностью до сотых долей миллиметра, однако намного проще вызвать диалоговое окно *Параметры страницы*.

3.4.5. Общие принципы работы с документами

Минимальной структурной единицей текста документа MS Word является *абзац*. Фактически весь текст MS Word представляет собой некоторое количество абзацев. Друг от друга абзацы отделяются нажатием клавиши **Enter**. Иногда требуется начать предложение с новой строки, не разрывая абзац. Для этого следует использовать сочетание клавиш **Shift + Enter**.

Мощным средством работы с документами является возможность манипулирования отдельными его частями (блоками). Их можно копировать, перемещать, удалять, изменять параметры шрифта, абзаца и т.д. Первым шагом для работы с отдельным блоком является его выделение (рис. 3.33).

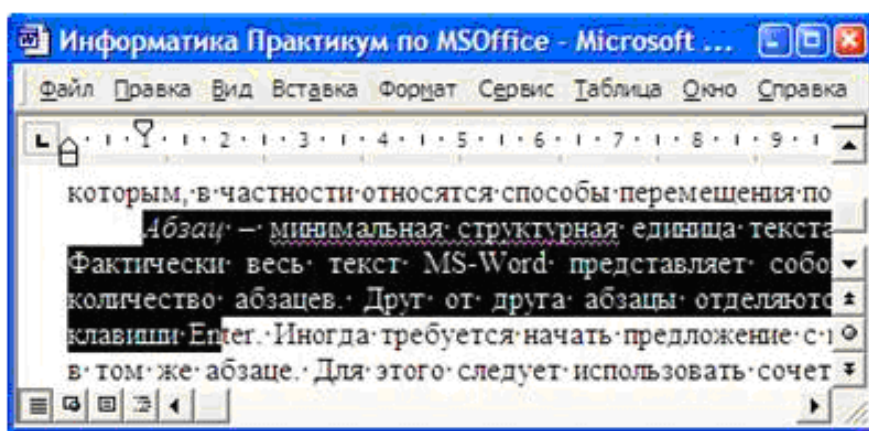


Рис. 3.33. Выделение фрагмента

Выделить часть документа можно либо с помощью мыши, либо клавиатуры.

Для выделения с помощью мыши необходимо:

- поместить указатель в начало фрагмента;
- нажать левую кнопку и переместить указатель мыши в конец фрагмента, не отпуская кнопку мыши.

Если требуется выделить строку целиком, то можно поместить указатель мыши в левое поле напротив этой строки и нажать левую кнопку.

Для выделения абзаца нужно поместить указатель мыши в левом поле напротив абзаца и дважды кликнуть левой кнопкой. Тройной клик левой кнопки мыши в левом поле страницы позволит выделить весь текст документа.

Выделение фрагментов документа с помощью клавиатуры происходит при одновременном нажатии клавиши **Shift** и клавиш управления курсором. Нажатие комбинации клавиш **Ctrl + A** позволяет выделить весь текст документа.

Рассмотрим следующий пример. Пользователь набрал достаточно большой фрагмент текста, но в ходе редактирования оказалось, что этот фрагмент должен размещаться двумя страницами раньше (выше). Естественно, повторный набор здесь не уместен. Для перемещения (или копирования) этого фрагмента необходимо выполнить следующие действия:

- выделить фрагмент с помощью мыши или клавиатуры;
- выделенный фрагмент скопировать (**Ctrl + Ins** или **Ctrl + C**), либо переместить (**Shift + Del** или **Ctrl + X**) в буфер обмена (в последнем случае выделенный фрагмент удаляется из исходного места в документе);
- установить курсор в новое положение;
- вставить содержимое буфера обмена в документ (**Shift + Ins** или **Ctrl + V**).

Так как к буферу обмена MS Windows имеют доступ все приложения, то его содержимое может быть перенесено в любую другую Windows-программу, и, наоборот, данные любой Windows-программы могут быть использованы в MS Word. Единственное ограничение здесь – приложение должно “уметь” работать с получаемыми из буфера обмена данными.

3.4.6. Средства поиска и замены

Очень часто возникает необходимость найти какой-либо фрагмент текста или отдельное слово в документе. Для этих целей используется специальное средство MS Word, доступ к которому можно получить одним из следующих способов:

- вызвать команду меню Правка / Найти...;
- нажать сочетание клавиш **Ctrl + F**

В обоих случаях на экране будет отображено окно, показанное на рис. 3.34.

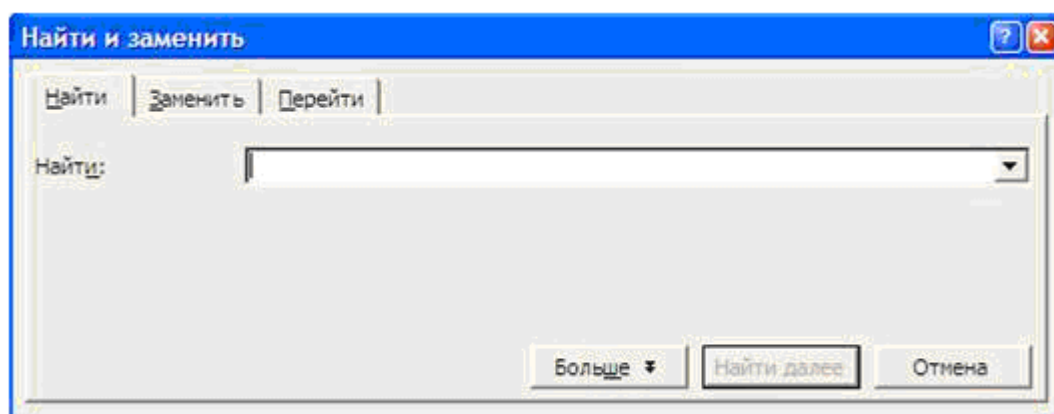


Рис. 3.34. Поиск в MS Word

Для того чтобы выполнить поиск, достаточно ввести искомый фрагмент в область ввода Найти: и нажать клавишу **Enter** (либо нажать кнопку *Найти далее*). MS Word просмотрит документ и перейдет на ту его страницу, на которой искомый фрагмент был найден. Повторное нажатие кнопки *Найти далее* приведет к следующей итерации поиска в документе, и опять будет выполнен автоматический переход на страницу, в тексте которой был найден введенный фрагмент.

Значительно повысить эффективность поиска помогают дополнительные параметры, доступные после нажатия кнопки Больше. Здесь можно уточнить такие опции, как:

- Учитывать регистр – по умолчанию при поиске нет разницы между прописными и строчными буквами;
- Только слово целиком – введенный фрагмент не может являться составляющей частью слова;
- Направление: – определяет направление поиска (вперед, назад или везде по тексту).

- *Специальный* – кнопка с выпадающим меню, позволяющая ввести в строку поиска или замены код специального символа (знак абзаца, знак разрыва строки, неразрывный пробел и т. д.).

Замена в документе осуществляется аналогичным образом. На рис. 3.35 представлено окно для осуществления процедуры замены одного фрагмента текста документа на другой, которое можно вызвать командой Правка / Заменить... или сочетанием клавиш **Ctrl + H**.

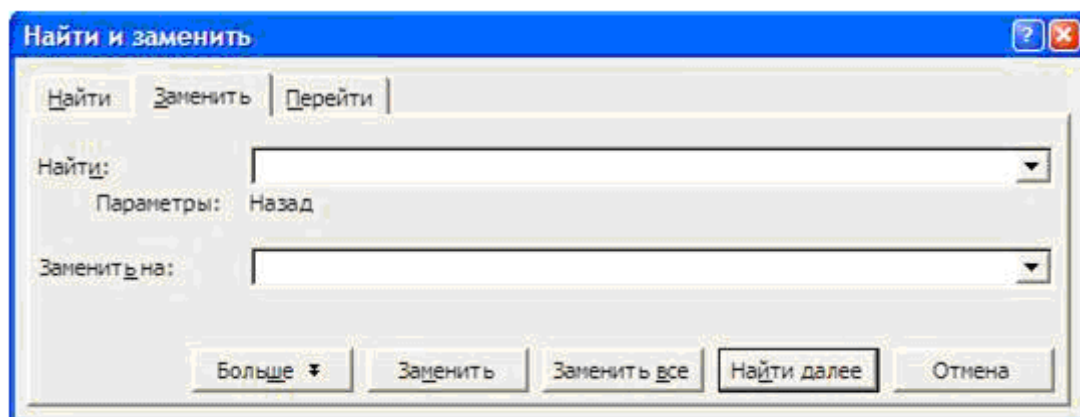


Рис. 3.35. Замена в MS Word

Как видно на рис. 3.35, вводится текст для поиска и для замены. Если пользователь уверен, что требуется заменить все вхождения, то можно сразу нажать кнопку *Заменить все*. В этом случае MS Word просмотрит весь текст и все вхождения, соответствующие фрагменту, указанному в поле *Найти:*, заменит на фрагмент, указанный в поле *Заменить на:*.

Однако, если полной замены не требуется, то можно выполнить ее по шагам. Для этого ищем фрагмент, нажав на кнопку *Найти далее*, и только в нужных местах заменяем, нажимая на кнопку *Заменить*.

Следует отметить, что процедуры поиска и замены могут выполняться не только во всем документе, но и в отдельной его области. Для этого нужно предварительно выделить эту область, а затем запустить процедуру поиска или замены (**Ctrl + F** или **Ctrl + H** соответственно).

В MS Word имеется возможность быстрого перемещения между фрагментами документа. Для этого необходимо вызвать специальное окно *Перейти* (рис. 3.36).

Вызвать его можно несколькими способами:

- дважды кликнуть мышкой в левой части строки состояния (в области отображения информации о страницах документа);
- нажать сочетание клавиш **Ctrl + G**;
- воспользоваться командой меню *Правка / Перейти...*

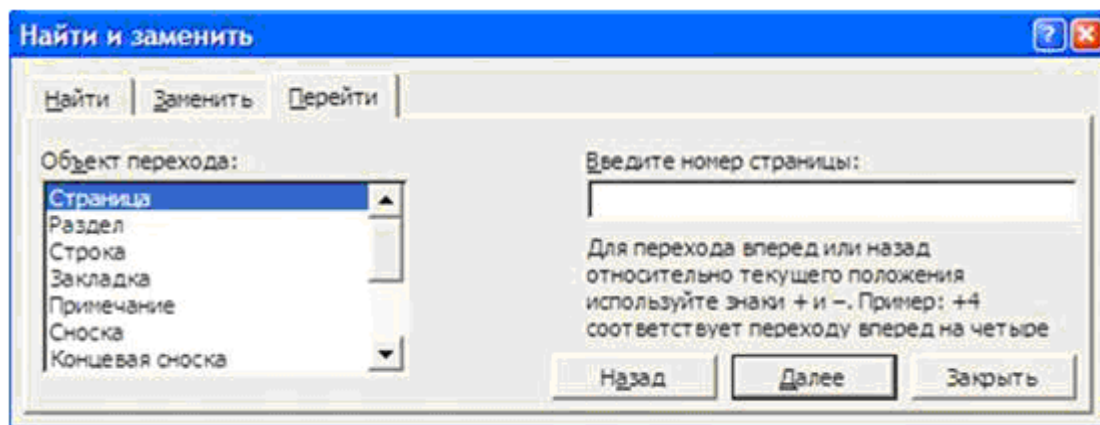


Рис. 3.36. Переход по документу MS Word

Переход выполняется по некоторому объекту. Тип объекта выбирается в левом окне *Объект перехода*: Так, в представленном на рис. 3.35 окне предполагается выполнить переход по номеру страницы. В поле *Введите номер страницы*: вводим номер (при этом кнопка *Далее* меняется на кнопку *Перейти*) и жмем кнопку *Перейти*. Если не ввести номер страницы, то нажатие кнопок *Назад* и *Далее* приведет к поочередному переходу по выбранным объектам. В представленном случае будет выполняться постраничный переход.

3.4.7. Изменение параметров шрифта

Шрифт играет одну из важных ролей в форматировании текста, т. к. именно он

Использование шрифтов для форматирования
 Использование шрифтов для форматирования
Использование шрифтов для форматирования
 Использование шрифтов для форматирования
 Использование шрифтов для форматирования

определяет начертание символов. На рис. 3.37 изображен один и тот же текст, но набранный разными шрифтами.

Рис. 3.37. Примеры различных шрифтов

Один шрифт от другого отличается составом и способом начертания символов. Каждый шрифт имеет свое уникальное имя. В примере на рис. 3.37 использовались следующие шрифты (сверху вниз, соответственно): **Times**

New Roman, Arial, Comic Sans MS, Franklin Gothic Medium и Palatino Linotype.

Изменить тип шрифта можно с помощью раскрывающегося списка (кнопки) *Шрифт* панели инструментов *Форматирование*. В поле списка отображается имя выбранного шрифта. На рис. 3.23 и 3.24 выбран шрифт **Times New Roman**.

На панели инструментов *Форматирование* размещены еще четыре элемента, позволяющие задать представление символов на странице документа MS Word:

- *Размер* – позволяет выбрать значение из выпадающего списка, либо задать его произвольно с клавиатуры, введя значение с точностью до 0,5 (например, 13,5);
- **Ж** (*Полужирный*) – текст **полужирный** выделяется толщиной линий символов;
- *К* (*Курсив*) – текст *курсив* выделяется наклоном букв;
- Ч (*Подчеркнутый*) – текст подчеркнутый выделяется подчеркиванием символов.

Следует сказать, что применять данное форматирование можно как к отдельным символам, так и к фрагментам текста любого размера. Причем данные форматы можно использовать в комбинациях друг с другом. Например, можно создать **полужирный курсив**, а можно полужирный курсив с подчеркиванием, изменяя при этом размер шрифта, и т. д. Доступ ко всем параметрам шрифта можно получить из меню *Формат / Шрифт...*. При этом на экране появляется диалоговое окно *Шрифт*, изображенное на рис. 3.38.

На вкладке *Шрифт* можно задать все основные параметры шрифта: изменить имя, начертание, размер, цвет и многое другое.

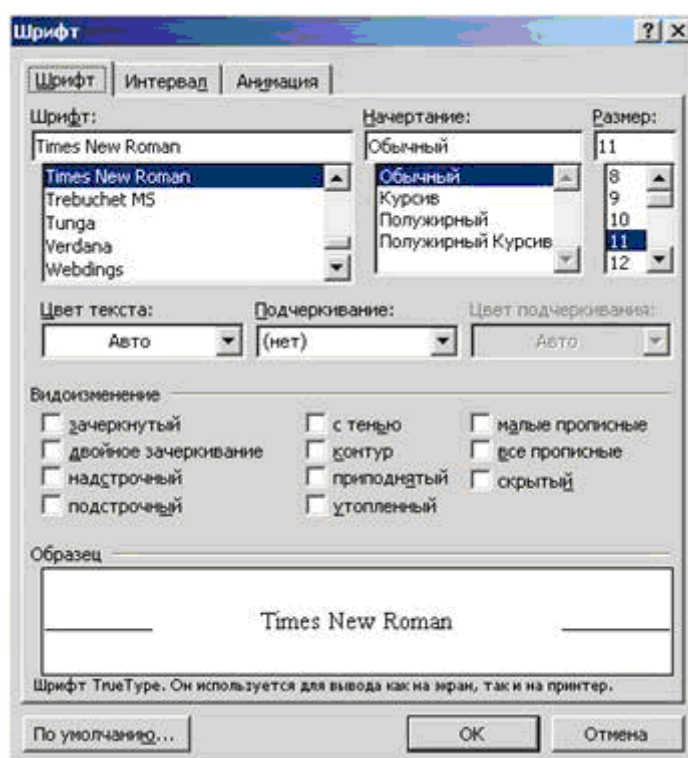


Рис. 3.38. Свойства шрифта

В частности, можно создавать нижние и верхние индексы.

Вкладка *Интервал* позволяет:

- изменить высоту букв относительно исходной (поле *Масштаб*);
- изменить величину межзнакового интервала (поле *Интервал*);
- сместить выделенный текст вниз или вверх относительно опорной линии (поле *Смещение*).

Выбор эффекта анимации, который следует применить к выделенному тексту, осуществляется на третьей вкладке – *Анимация*. Чтобы отменить эффект анимации, следует выбрать в поле *Вид*: параметр (*нет*). Одновременно можно выбрать только один эффект анимации. Следует помнить, что на печать выводится только текст. Эффект анимации отображается только на экране и на печать не выводится.

На каждой вкладке внизу есть специальное поле, в котором можно посмотреть, как будет выглядеть текст с выбранными параметрами. Изменять тип используемого шрифта можно как для уже введенного фрагмента, так и непосредственно перед вводом текста.

3.4.8. Настройки абзаца

Данные настройки позволяют определить положение текста абзаца на странице относительно полей, колонок, других абзацев и т. д. Для изменения абзацных параметров существует команда меню *Формат / Абзац...*. Диалоговое окно изменения параметров абзаца представлено на рис. 3.39.

Выравнивание предполагает различные способы размещения абзаца относительно полей.

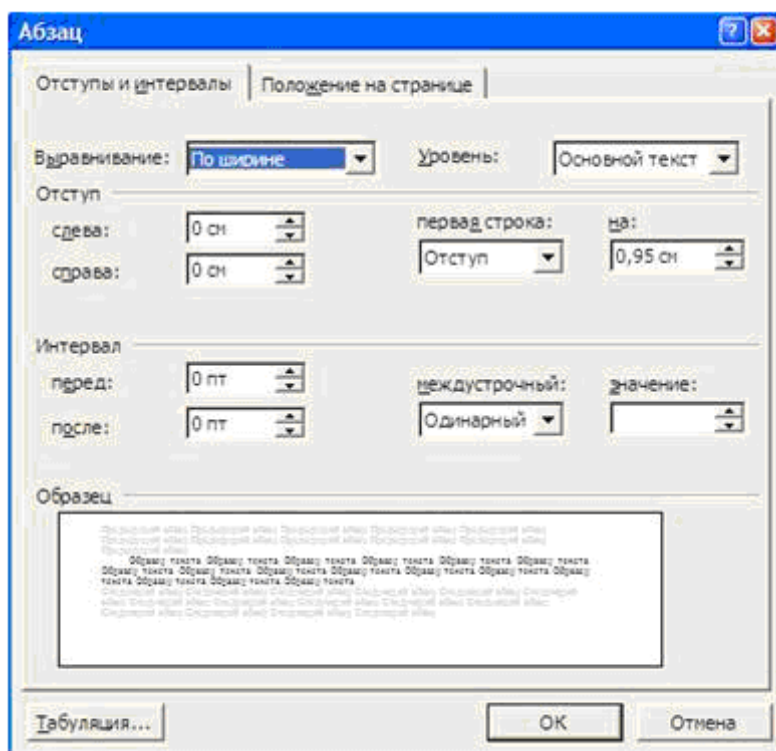






Рис. 3.39. Изменение параметров абзаца

Доступны четыре способа выравнивания текста: по левому краю, по правому краю, по ширине и по центру. Этим способам соответствуют кнопки панели инструментов *Форматирование*: , , , . Примеры разных способов выравнивания текста представлены на рис. 3.40.

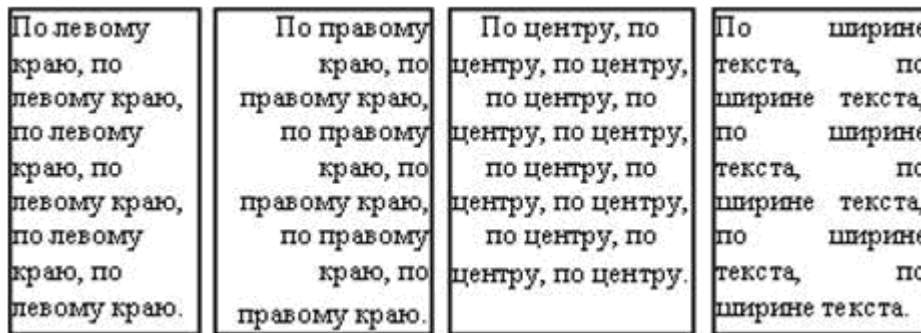


Рис. 3.40. Примеры выравнивания текста

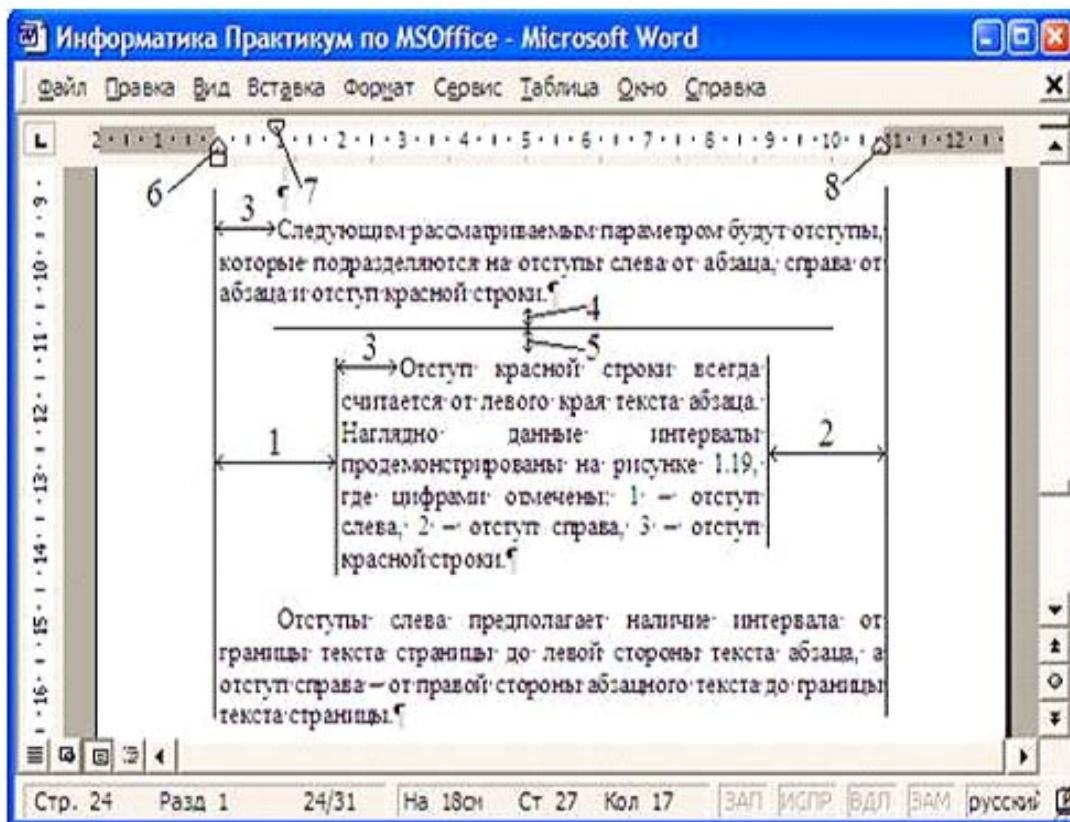


Рис. 3.41. Абзацные интервалы

Следующим рассматриваемым параметром будут отступы, которые подразделяются на отступы слева от абзаца, справа от абзаца и отступ красной строки. Отступ слева предполагает наличие интервала от левого поля страницы до текста абзаца, а отступ справа – от правого поля до абзаца. Отступ первой строки всегда считается

от левого края текста абзаца. Для создания выступа необходимо выбрать параметр *Выступ*. Величину отступа можно регулировать с помощью поля *на:*, которое располагается справа от поля *Отступ*. Наглядно данные интервалы продемонстрированы на рис. 3.41.

Цифрами отмечены: 1 – отступ слева, 2 – отступ справа, 3 – отступ красной строки. Самые крайние (левая и правая) вертикальные линии обозначают границы текста на странице и, как видно по линейке, расположены на расстоянии левого и правого полей.

Абзацные отступы слева, справа и отступ красной строки можно изменять также специальными бегунками, расположенными на горизонтальной линейке. Эти бегунки обозначены соответственно цифрами 6, 8 и 7 (рис. 3.40). Достаточно зацепить бегунок мышью и перетащить в новое положение. Здесь следует обратить внимание на то, что бегунок 6 состоит из двух частей. Верхняя часть (в виде треугольника) позволяет изменять абзацный отступ слева, без изменения отступа красной строки, относительно границ страницы. При изменении же абзацного отступа с помощью нижней части (в виде прямоугольника) отступ красной строки изменяется относительно границ текста, но не изменяется относительно левой границы абзаца. Оставшиеся два отступа, обозначенные на рисунке цифрами 4 и 5, определяют группу параметров *Интервал*: 4 – отступ после абзаца, а 5 – отступ перед абзацем.

Последний из рассматриваемых абзацных параметров – межстрочный интервал (*междустрочный*), который определяет расстояние между строками текста абзаца. Он указывается в единицах высоты символов. Например, межстрочный интервал *Одинарный* говорит о том, что строки находятся друг от друга на расстоянии, равном высоте символов. Иногда используются полуторный и двойной интервалы, но пользователь может указать и собственный множитель. Например, в области *значение*: можно ввести 1,8, что приведет к расположению строк на расстоянии 1,8 высоты символов друг от друга.

3.4.9. Вставка символов

Очень часто бывает необходимо вставлять в документ различные символы. Для этих целей используется команда меню *Вставка / Символ...*. Ее исполнение приведет к появлению диалогового окна, представленного на рис. 3.42. Важным моментом здесь является

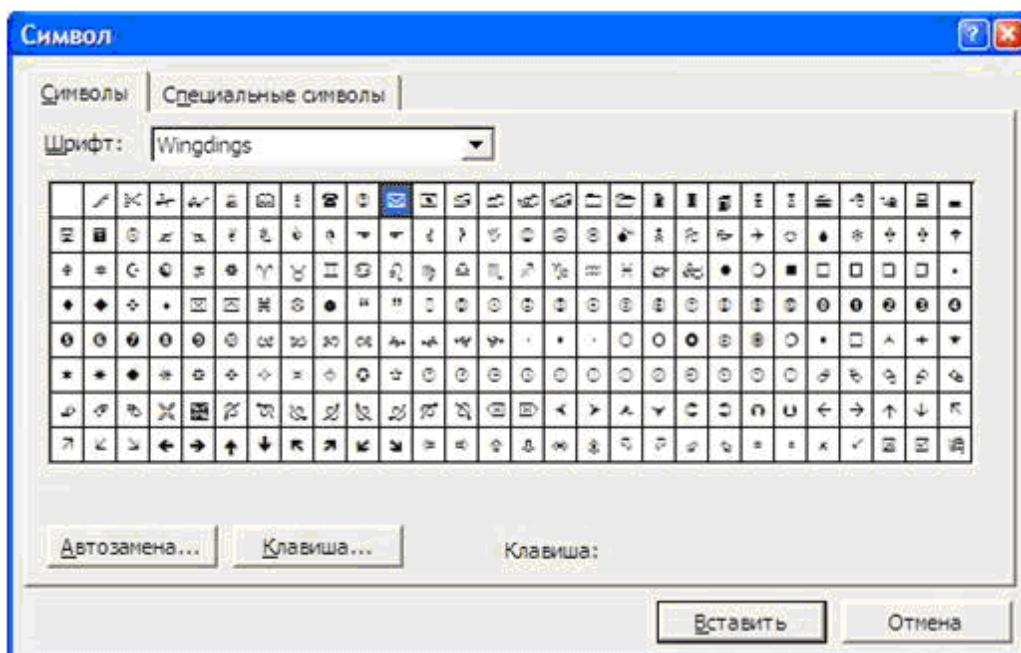


Рис. 3.42. Вставка символа

использование того или иного шрифта, который указывается в поле *Шрифт*: В Windows существуют шрифты, содержащие только специальные символы. К таким шрифтам относятся **Symbol**, **Wingdings** и др. Каждый из этих шрифтов содержит свой набор символов, поэтому для поиска нужного символа иногда требуется просмотреть несколько шрифтов, что, как правило, не представляет собой особого труда. Чтобы вставить символ в текст, нужно выбрать его, кликнув на изображении мышью, нажать кнопку *Вставить* и затем закрыть окно *Символ*.

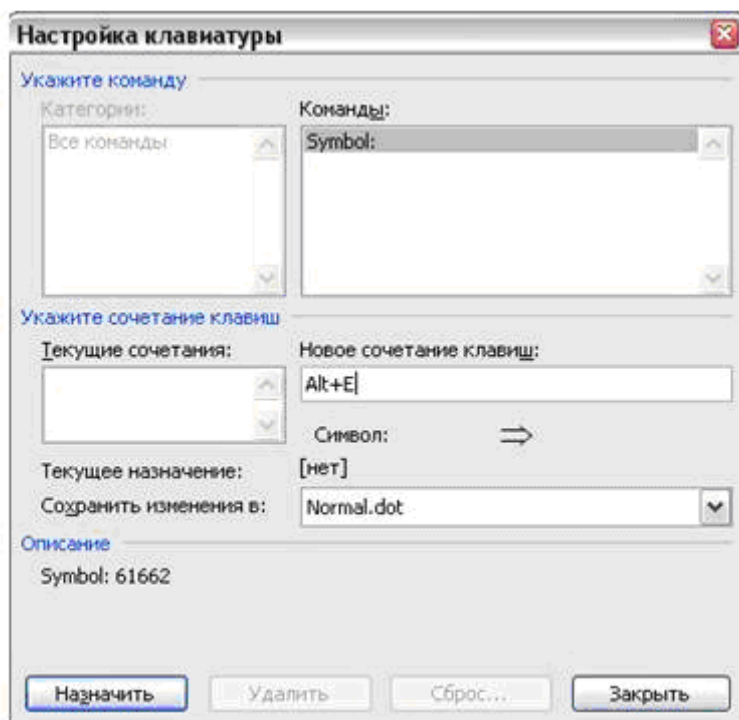


Рис. 3.43. Назначения сочетания клавиши

При наборе специализированных текстов вставка специальных символов (например, буквы греческого алфавита) происходит очень

часто. Постоянно открывать окно *Символ* не очень удобно, т.к. на это тратится много времени. Поэтому лучше всего назначить комбинацию клавиш для добавления в текст нужного символа. Осуществить это можно, используя кнопку *Клавиша...*, при нажатии на которую открывается окно *Настройка клавиатуры* (рис. 3.43). В поле *Новое сочетание клавиш*: можно задать сочетание клавиш для вставки символа. Сам символ отображается справа в рамке. Перед тем, как назначить новое сочетание клавиш, обратите внимание на поле *Текущие сочетания*:. Если некоторое сочетание уже назначено, тогда в новом назначении сочетания нет необходимости.

Кроме специальных символов, в текст документа можно вставлять даты, ссылки, поля, номера страниц и другие элементы. Все это возможно с помощью команд меню *Вставка*.

3.4.10. Колонки

В газетах и некоторых книгах (обычно словарях и энциклопедиях) информацию нередко представляют в форме столбцов. Это связано с тем, что короткие строчки легче читать.

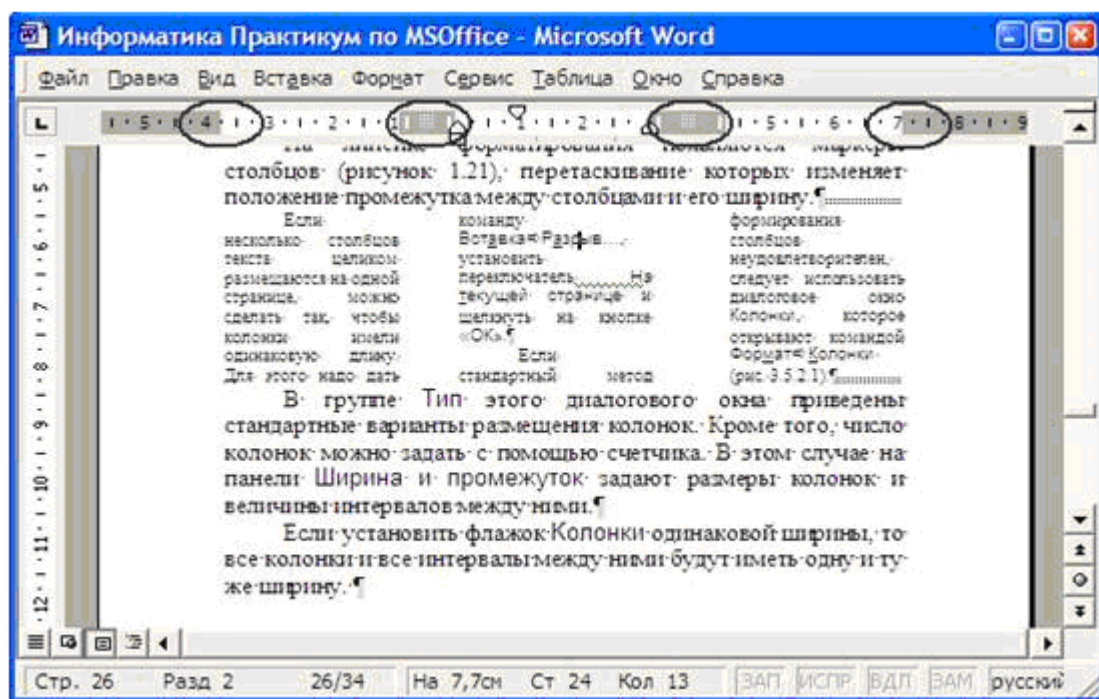


Рис. 3.44. Работа с колонками (столбцами) текста

Для разбиения текста на столбцы его вводят обычным образом, затем выделяют и щелкают на кнопке *Колонки* панели инструментов *Форматирование*. В открывшемся меню выбирают количество создаваемых колонок, причем по умолчанию все колон-

ки имеют одинаковую ширину. На горизонтальной линейке появляются маркеры колонок (обведенные овалами области на рис. 3.44), перетаскивание которых изменяет положение промежутка между столбцами и их ширину.

Размером содержимого колонок можно управлять, т. е. явно определить положение в тексте, откуда следует начинать новую колонку. Для этого надо вызвать команду меню *Вставка / Разрыв...*, в появившемся окне выбрать *новую колонку* и нажать кнопку *ОК*.

Если стандартный метод формирования столбцов не устраивает, следует использовать диалоговое окно *Колонки* (рис. 3.45), которое открывают командой меню *Формат / Колонки...*

В группе *Тип* этого диалогового окна приведены стандартные варианты размещения колонок. Число колонок можно задать с по-

мощью поля *Число колонок:*. В этом случае в области *Ширина и промежуток* задают размеры колонок и величины интервалов между ними.

Если установить флажок *колонки одинаковой ширины*, то все колонки и все интервалы между ними будут иметь одну и ту же величину, а если установить флажок *Разделитель*, то столбцы будут отделены друг от друга вертикальными линиями.

Набор текста производится по колонкам, то есть сначала заполняется первая (левая) колонка, затем вторая, третья и т. д. Разделитель между первой и второй колонками появится только после того, как будет набран текст во второй колонке.

Список *Применить:* указывает, какую часть документа следует разбить на столбцы. Доступные варианты зависят от того, был ли

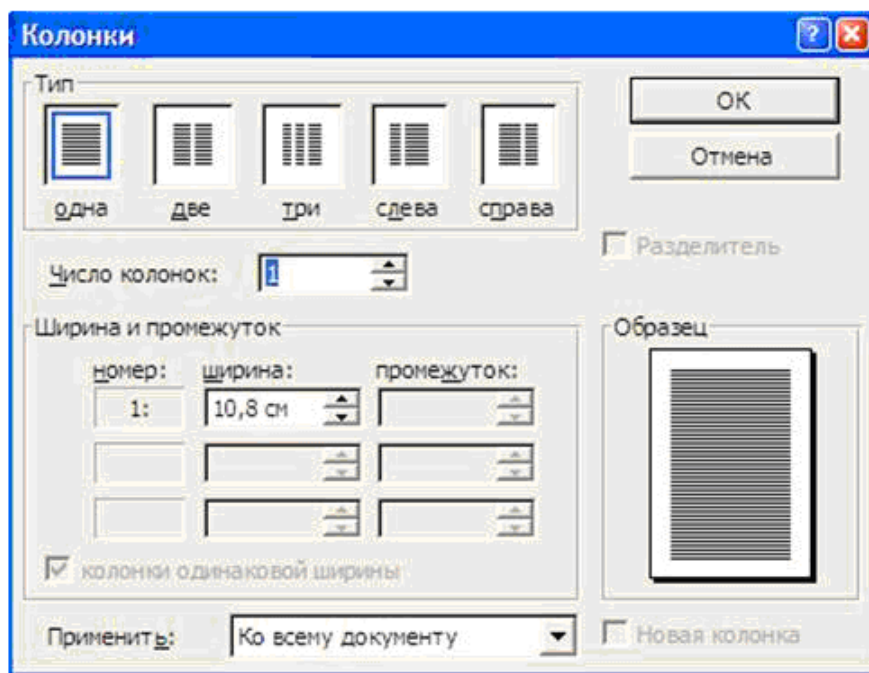


Рис. 3.45. Создание колонок

предварительно выделен фрагмент текста. Если установить флажок *Новая колонка*, то последующий текст документа располагается в начале нового столбца. То же можно сделать нажатием сочетания клавиш **Shift + Ctrl+ Enter**.

3.4.11. Списки

Упорядоченную тем или иным образом информацию (инструкции, перечисления и т. п.) удобно представлять специфической структурой, выделяющей каждый из ее элементов. Таким средством в MS Word являются списки. Различают маркированные, нумерованные и многоуровневые списки.

Для создания списков на панели инструментов *Форматирование* расположены две кнопки: *Нумерованный список по умолчанию* и *Маркированный список по умолчанию*. Нажатие на какую-либо из этих кнопок приводит к созданию нового списка либо, если до нажатия был выделен некоторый фрагмент текста, к преобразованию этого фрагмента в список. Изменять представление списка можно с помощью диалогового окна *Список* после выбора команды меню *Формат / Список...*. При этом курсор должен находиться на каком-либо элементе форматлируемого списка.

3.4.11.1. Маркированный список

Маркированный список представляет собой структуру, каждый элемент которой отмечен специальным *маркером*. В качестве маркера могут выступать символы, картинки и другие элементы оформления, которые позволяют выделить элементы списка без указания их последовательности. Изменение отображения маркированных списков доступно на вкладке *Маркированный* диалогового окна *Список* (рис. 3.46, а). Изменить маркер списка можно, нажав кнопку *Изменить...* и в появившемся окне выбрав нужный маркер.

3.4.11.2. Нумерованный список

В отличие от маркированного, нумерованный список определяет порядковые позиции элементов, например **1, 2, 3,...** или **a, b, c, d,...** и т. п., как показано на рис. 3.46, б. Изменить способ нумерации элементов списка данного типа можно на вкладке *Нумерованный* диалогового окна *Список*. Как и в случае с маркированным спи-

ском, формировать индивидуальное представление нумерованного списка можно после нажатия кнопки *Изменить...* .

3.4.11.3. Многоуровневый список

Многоуровневый список представляет собой произвольные комбинации маркированных и нумерованных списков.

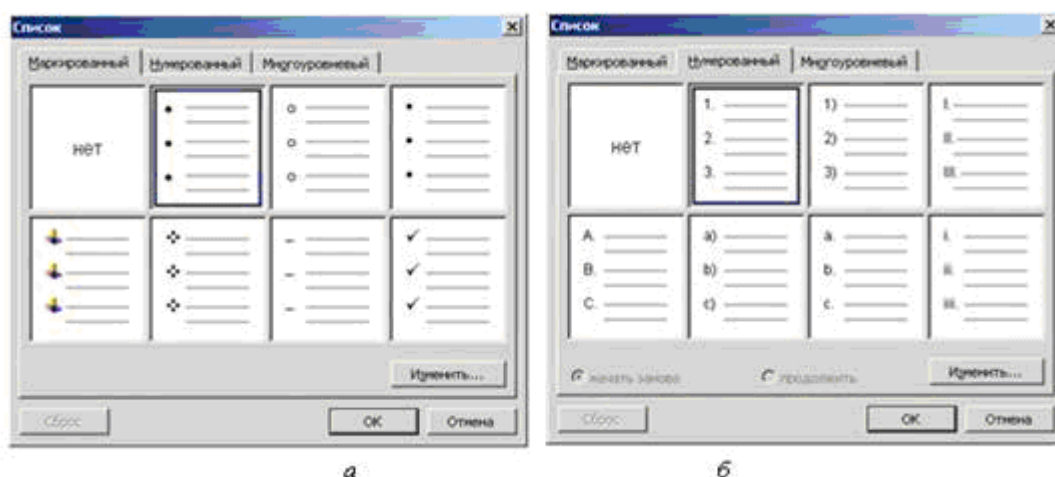


Рис. 3.46. Списки:

а – маркированный; *б* – нумерованный

Создать многоуровневый список можно из любого рассмотренного типа списков с помощью вкладки *Многоуровневый* диалогового окна *Список*. Для создания многоуровневого списка можно выполнить следующие шаги:

- 1) набрать элементы списка без номеров, маркеров, отступов и т. д. в столбик друг под другом;
- 2) выделить все элементы будущего списка;
- 3) открыть диалоговое окно *Список* с помощью команды меню *Формат / Список... / Многоуровневый*;
- 4) выбрать один из предложенных форматов и нажать кнопку *Изменить...* (при этом откроется диалоговое окно *Изменение многоуровневого списка*, показанное на рис. 3.47);
- 5) в левой части окна выбрать первый уровень и задать для него тип и формат номера, отступы для текста и номера (маркера), затем выбрать второй уровень и задать параметры для него и для всех используемых уровней, после чего нажать кнопку *ОК*;
- 6) в результате должен получиться список, все элементы которого относятся к первому уровню;

- 7) для перевода на другой уровень элемента списка необходимо выделить строку (можно просто поместить курсор в начало строки) и нажать клавишу **Tab** (переместить элемент списка на уровень вверх можно с помощью комбинации клавиш **Shift + Tab**).

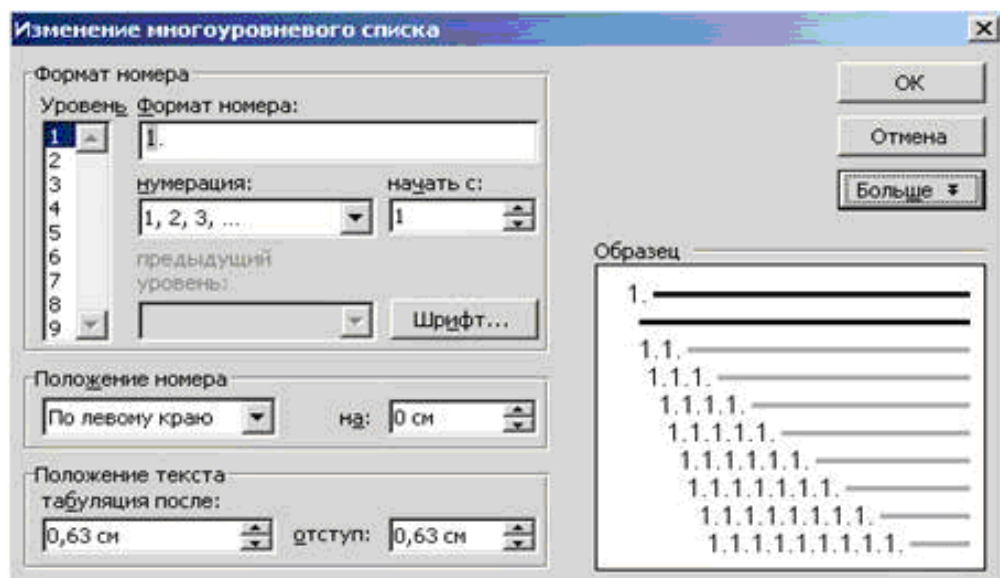


Рис. 3.47. Изменение формата многоуровневого списка

3.4.12. Вставка и редактирование формул

Фирма Microsoft значительно увеличила количество пользователей редактора MS Word, предоставив продукт под названием *Редактор формул*, который позволяет быстро и легко создавать (изображать) различные математические формулы.

Чтобы начать работу с редактором формул, нужно убедиться, что он инсталлирован (установлен) на ПК. Для этого следует вынести на панели инструментов команду *Добавить объект Microsoft Equation* из категории команд *Вставка*. Если значок не активен (картинка с изображением буквы альфа под знаком радикала серая), значит, редактор формул не установлен. После запуска редактора формул появляются панели формул (верхняя – панель символов и нижняя – панель шаблонов) и слот – прямоугольная область ввода формулы (рис. 3.48).

Свободные для ввода данных поля обведены пунктиром. Создать примитив (шаблон) или символ можно следующим образом: кликом мыши открыть соответствующую группу шаблонов (как это сделано на рис. 3.48) и выбрать нужный элемент.

Например, для создания показанной на рис. 3.48 формулы требуется выполнить следующие действия:

- вставить шаблон (дробь);
- установить курсор в верхнюю область ввода (числитель) и ввести число один;
- переместить курсор в нижнюю область ввода и вставить символ (из панели символов со строчными греческими буквами);
- поместить курсор в крайнее правое положение мышью или клавишами управления (перемещения) курсором;
- вставить знак суммы (обратите внимание, что знаков суммы несколько – пять).

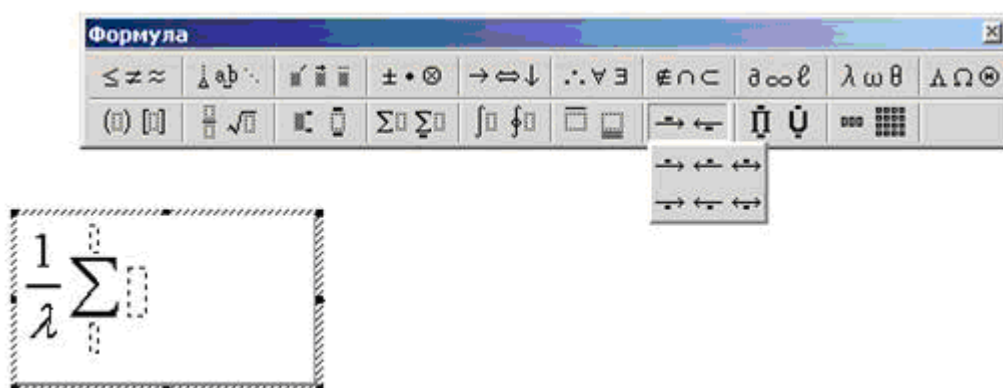


Рис. 3.48. Работа с редактором формул

В редакторе формул пробел ставится с помощью панели инструментов. Вторая слева кнопка верхней панели инструментов содержит шесть видов пробелов различной длины, пять видов многоточий. Пробел среднего размера можно поставить и с помощью клавиатуры. Для этого используется комбинация клавиш **Ctrl + Space**.

При вводе формул обратите внимание на два типа шаблонов, имеющих одинаковое назначение. Например, верхний правый индекс. Первый тип шаблонов представляет собой закрашенную область и черное поле, обозначающее месторасположение индекса. Ввод символа осуществляется сразу в положение, соответствующее закрашенной области, т. е. необходимо набрать символ (или строку символов) и уже затем выбрать эту команду. Второй шаблон изображается контуром и черным полем. При его использовании

появляются сразу два поля: первое – для ввода символа или формулы, а второе – для индекса.

Чтобы выйти из редактора формул, достаточно щелкнуть мышью вне редактируемого поля (слота) либо нажать клавишу **Esc**. Для повторного редактирования используется двойной клик мыши на формуле.

Если выйти из редактора, ничего не вводя, система создает невидимую область. Для устранения такого неприятного эффекта следует выделить данную область щелчком мыши (появляется маркированный квадрат) и нажать клавишу **Del**.

3.4.13. Создание и редактирование таблиц

Еще одним способом форматирования текста в MS Word являются таблицы, позволяющие наглядно представить различного рода информацию: результаты анализа, статистических прогнозов, финансовых отчетов и т. д.

Таблица – один или несколько рядов ячеек, обычно используемых для представления чисел и других элементов в виде, упрощающем их просмотр и анализ.

MS Word предлагает несколько способов создания таблицы. Выбор наилучшего из них зависит от сложности отображаемых данных и требований пользователя. Вначале следует выбрать место для создания таблицы, а затем использовать один из следующих способов:

- 1) если формат таблицы несложный, то необходимо на панели инструментов *Стандартная* нажать кнопку *Вставить таблицу*, выбрать нужное число строк и столбцов перемещением указателя мыши;
- 2) если формат таблицы должен отличаться от используемого по умолчанию, тогда надо выбрать команду меню *Таблица / Вставить / Таблица* и в появившемся окне (рис. 3.49) задать число строк, столбцов, параметры для настройки размера таблицы (группа *Автоподбор ширины столбцов*), один из встроенных форматов (кнопка *Автоформат...*).

Для создания более сложного представления таблиц в MS Word имеется целый набор средств.

Добавление строк/столбцов к уже существующей таблице возможно с помощью меню *Таблица / Вставить* и соответствующих команд добавления строк/столбцов. Удаление строк/столбцов осуществляется с помощью команд меню *Таблица / Удалить*. Аналогичные действия можно выполнить соответствующими ко-

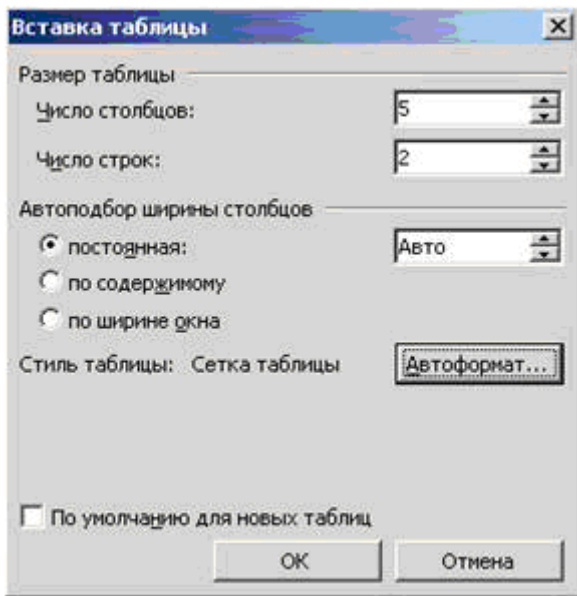


Рис. 3.49. Вставка таблицы

мандами контекстного меню, если вызвать его на этих элементах таблицы.

Ячейки таблицы можно также разбивать на несколько ячеек и выполнять обратную процедуру – объединять несколько ячеек в одну. В первом случае следует вызвать контекстное меню на ячейке, выбрать команду *Разбить ячейки...* и в появившемся окне указать нужное количество строк и столбцов. Во втором случае ячейки для объединения нужно выделить, а затем, вызвав контек-

стное меню, выбрать команду *Объединить ячейки*.

На рис. 3.50, а представлен пример таблицы с разбиением ячеек, а на рис. 3.50, б – с их объединением. Кроме того, на этом рисунке для оформления таблиц использовались различные приемы оформления, такие, как настройка оформления таблицы, изменение направления текста ячеек, выравнивание содержимого ячеек.

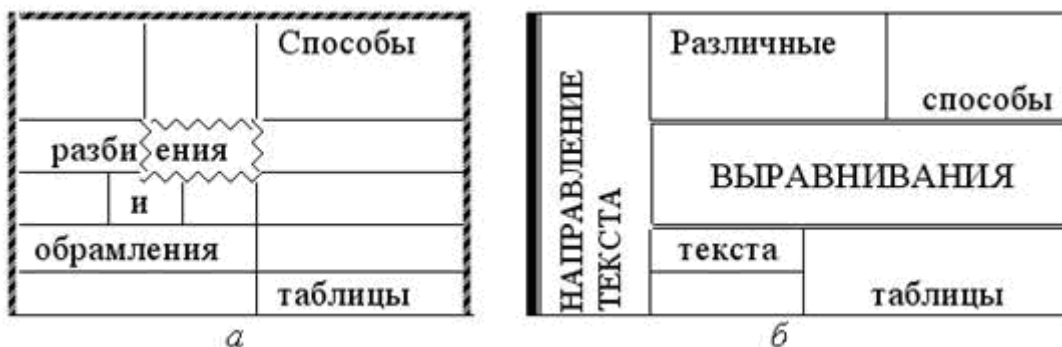


Рис. 3.50. Представление таблиц

Вид таблицы можно изменять с помощью команд панели инструментов *Таблицы и границы*, представленной на рис. 3.51, а.

Например, для изменения границ следует выбрать тип линий границы (на рисунке выбрана одинарная сплошная линия), их толщину (на рисунке установлено значение 0,5), а затем с помощью элемента *Внешние границы* определить границы таблицы, для которых следует использовать указанный тип форматирования.

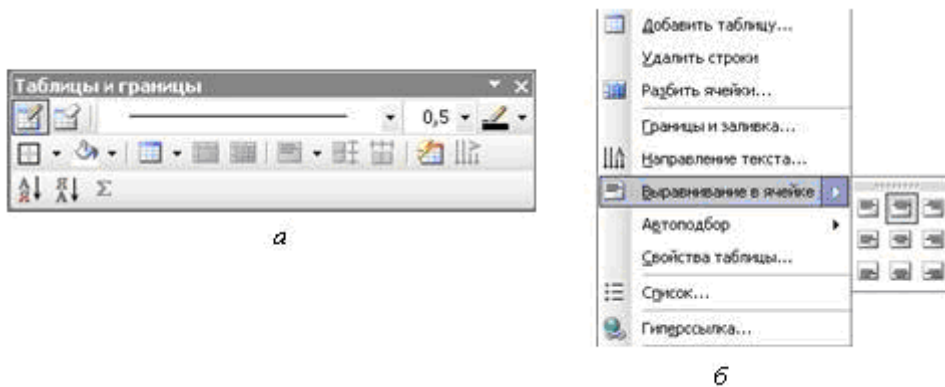


Рис. 3.51. Примеры работы с таблицами

Текст, расположенный в ячейках таблицы, может быть отображен с разным направлением. Для этого используется соответствующая команда *Направление текста* панели *Таблицы и границы*, исполнение которой приводит к смене направления вывода текста: горизонтально слева направо, горизонтально справа налево, вертикально снизу вверх, вертикально сверху вниз (рис. 3.51, б).

Содержимое ячеек также может быть выровнено внутри ячейки несколькими способами. Эти способы выравнивания доступны из контекстного меню. На рис. 3.51, б показано, как можно задать способ выравнивания информации в ячейке.

Для изменения практически всех параметров таблицы можно использовать команду меню *Таблица / Свойства таблицы...*. При выборе этой команды на экране появляется диалоговое окно, с помощью которого пользователь может изменять любые элементы таблицы и любые параметры этих элементов.

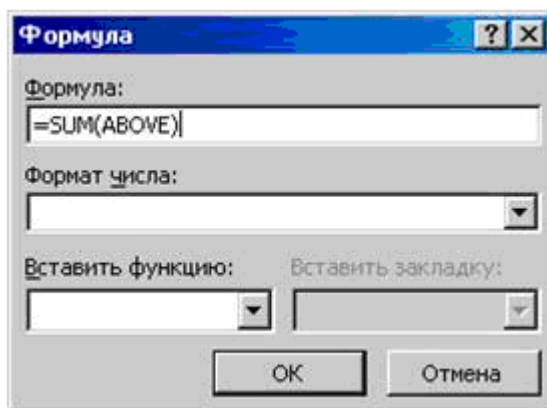


Рис. 3.52. Создание и редактирование формул

В таблицах MS Word предусмотрена возможность проведения элементарных математических расчетов. Для этого необходимо поместить курсор в ячейку и открыть диалоговое окно *Формула* с помощью команды *Таблица / Формула...* (рис. 3.52).

В первое поле вводится формула (выражение) с использованием простых арифметических операций и некоторых стандартных функций, доступных из выпадающего списка *Вставить функцию*:. Во втором поле *Формат числа*: задается формат представления результата.

Практически в любой формуле используются значения, находящиеся в других ячейках. Чтобы получить эти значения, необходимо обратиться к ячейке по ее адресу. Адрес ячейки формируется, как и в таблицах MS Excel, из индекса столбца и номера строки. Столбцы обозначаются буквами латинского алфавита (А, В, С, ...). Нумерация строк начинается с единицы. Рядом показана таблица размера 3 × 4, в каждой ячейке которой помещен ее адрес.

| | | | |
|----|----|----|----|
| A1 | B1 | C1 | D1 |
| A2 | B2 | C2 | D2 |
| A3 | B3 | C3 | D3 |

Если в функции используются значения из нескольких ячеек, то их адреса перечисляются через точку с запятой (A2; B4; C3; C4). В том случае, когда ячейки располагаются рядом, можно указать только адрес первой и адрес последней ячеек, разделив их двоеточием (интервальная ссылка).

Существуют еще два зарезервированных слова, обозначающие группы ячеек:

- LEFT – указывает на все ячейки, стоящие в строке слева от ячейки с формулой;
- ABOVE – указывает на все ячейки, стоящие в столбце выше от ячейки с формулой.

В табл. 3.8 представлены некоторые стандартные функции, используемые в поле *Формула*:.

Таблица 3.8

| Функция | Результат |
|-------------|--|
| ABS (x) | Абсолютное значение числа или формулы |
| AVERAGE () | Среднее значений, включенных в список |
| MIN () | Наименьшее значение в списке |
| MAX () | Наибольшее значение в списке |
| PRODUCT () | Произведение значений, включенных в список |
| SUM () | Сумма значений или формул, включенных в список |

Для всех функций, кроме ABS (), допустимо любое число аргументов, отделенных друг от друга точкой с запятой (;). Аргументы могут быть числами, формулами или именами *закладок*, содержащих числа. Закладка – отметка или место в тексте, которому присвоено определенное имя, что позволяет переходить к нему в дальнейшем.

Для выполнения вычислений в таблице MS Word можно использовать следующий алгоритм:

- 1) выделить ячейку, в которую будет помещен результат;
- 2) в меню *Таблица* выбрать команду *Формула...*;
- 3) если MS Word предлагает неподходящую формулу, удалить ее из поля *Формула:*;
- 4) из списка *Вставить функцию:* выбрать необходимую;
- 5) для использования ссылок на ячейки таблицы ввести в формулу их адреса в скобках;
- 6) в поле *Формат числа:* задать формат представления результата.

MS Word вставляет результат вычисления в выбранную ячейку в виде поля. При изменении ссылок на ячейки результаты вычисления можно обновить, выделив поле и нажав клавишу **F9**.

3.4.14. Графика в документах MS Word

Двумя основными типами рисунков, используемых в документах MS Word, являются графические объекты и рисунки. Графические объекты включают автофигуры (набор фигур, состоящий из прямоугольников, окружностей, различных типов линий и т. п.), кривые, линии и объекты WordArt (художественно оформленный текст). Эти объекты являются частью документа MS Word. Для изменения этих объектов, а также их цвета, заливки, границ и других параметров используется панель инструментов *Рисование*.

Рисунки представляют собой файлы, созданные с помощью графических редакторов, отсканированные изображения и др. Рисунки также делятся на два типа: рисунки, которые нельзя разгруппировать (большинство импортированных рисунков), и рисунки, которые можно разгруппировать (метафайлы из коллекции клипов). После разгруппирования рисунок можно преобразовать в графический объект и изменить с помощью кнопок панели инструментов *Рисование*.

В коллекции клипов содержится набор картинок, которые можно использовать в документах MS Word. Доступ к ним можно получить из меню *Вставка / Рисунок / Картинки...*. Большинство картинок выполнено в формате метафайла. То есть для изменения картинки необходимо разгруппировать ее и преобразовать в графический объект. Коллекция клипов включает большой выбор рисунков, упрощающих оформление документа профессионально сделанными изображениями.

Помимо объектов из коллекции клипов, в документ MS Word можно вставлять уже готовые рисунки. Если рисунок представляет собой файл одного из графических форматов, то вставить его в документ можно командой *Вставка / Рисунок / Из файла...*. Следует, однако, отметить, что для выполнения каких-либо действий с таким объектом его нужно сначала преобразовать в графический объект MS Word. Это можно сделать, например, изменив положение объекта в тексте с помощью диалогового окна *Формат объекта*.

Еще одним элементом графики является надпись. Надписи можно использовать в качестве выносок, подписей и других типов текста, описывающего рисунки. После вставки надписи ее можно изменять кнопками панели инструментов *Рисование*, как и любой другой графический объект.

Все графические объекты обладают набором параметров, позволяющих определить формат рисунка и его расположение в тексте. Управление этими параметрами осуществляется с помощью диалогового окна *Формат объекта* (рис. 3.53), доступного по команде *Формат объекта...* контекстного меню.

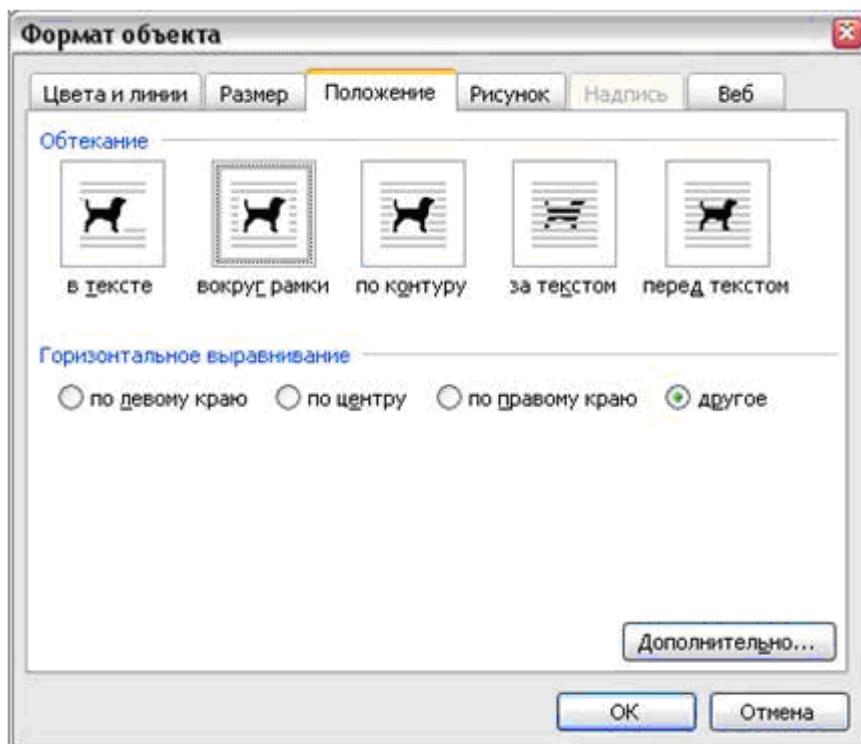


Рис. 3.53. Формат графического объекта


Название команды и окна может изменяться: при изменении формата надписи используется команда *Формат надписи...*, при изменении формата автофигуры – команда *Формат автофигуры...* и т. п.

Для изменения параметров рисунка доступно шесть вкладок, из которых наибольший интерес представляют следующие:

- *Цвета и линии* позволяют определить цвет и прозрачность заливки (заполнения графического объекта), толщину, тип и цвет линий, а также тип стрелок на концах линий;
- *Размер* позволяет определить размеры графического объекта (в процентах или сантиметрах) и угол наклона;
- *Положение* задает расположение рисунка в тексте и способы обтекания рисунка текстом;
- *Надпись* определяет расстояния от границ объекта до текста надписи (сверху, снизу, слева и справа).

При форматировании текста наибольший интерес представляет вкладка *Положение*. Как показано на рис. 3.53, выделяются пять способов обтекания рисунка текстом, при этом для большинства из них можно указать способ горизонтального выравнивания рисунка. Если представленных способов недостаточно, то можно нажать кнопку *Дополнительно...* и воспользоваться дополнительными параметрами.

Все графические объекты могут быть *сгруппированы* – специальным образом объединены в один объект, у которого можно изменять параметры: положение в тексте, положение на странице и т. д. Сгруппировать объекты можно следующим образом:

- выделить нужные объекты, для чего нажать кнопку , подвести указатель мыши к объектам и, зажав левую кнопку мыши, обозначить прямоугольную область; все объекты, попавшие внутрь области, будут выделены (повторное нажатие кнопки *Выбор объектов* приводит к отключению данного режима);
- можно выделять объекты и простым кликом мыши, но при этом необходимо нажать и удерживать клавишу **Shift**;
- выбрать команду *Группировать* панели инструментов *Рисование*;
- разгруппировать ранее сгруппированный рисунок можно командой *Разгруппировать* панели *Рисование*.

3.4.14.1. Создание рисунков

Для создания рисунков используется панель инструментов *Рисование* (нижняя область с номером 2, изображенная на рис. 3.23). По умолчанию на ней представлены следующие кнопки: *Линия*, *Стрелка*, *Прямоугольник*, *Надпись*, *Добавить объект WordArt*, *Цвет заливки*, *Цвет линий*, *Тип линий*, *Штрих* и другие.

Для создания какого-либо объекта нужно выбрать соответствующую команду, например линию, затем нажать левую кнопку мыши на странице в позиции начала линии и, не отпуская, переместить в точку, соответствующую концу линии, а затем отпустить левую кнопку. Линия будет нарисована. Для изменения ее свойств нужно сначала выделить линию (кликнув по ней мышью), а уже затем эти свойства изменять. Сделать сплошную линию пунктирной, например, можно, выбрав команду *Штрих*. Но можно вызвать диалоговое окно *Формат объекта* (рис. 3.53) и задать все параметры сразу. Кроме того, имеется достаточно большой выбор автофигур, доступ к которым осуществляется с помощью кнопки *Автофигуры* панели инструментов *Рисование*. Предположим, что требуется создать некоторую фигуру с надписью внутри (рис. 3.54). Для этого применим следующий алгоритм:



Рис. 3.54. Фигура с надписью

- с помощью кнопки *Автофигуры* выбрать необходимый объект и вставить его в документ;
- с помощью контекстного меню выбрать команду *Добавить текст*;
- добавить (или изменить) надпись.

Достаточно важным свойством любого объекта является порядок следования. Его можно определить с помощью команд меню *Действия / Порядок / На передний план* (*На задний план*, *Переместить вперед*, *Переместить назад*). На рис. 3.54 показан цилиндр с заштрихованной верхней гранью. Однако сделать верхнюю грань цилиндра штрихованной невозможно. На самом деле этот цилиндр состоит из двух фигур: цилиндра с надписью и расположенного поверх него заштрихованного овала.

3.4.14.2. Перемещение и копирование рисунков

Переместить любой рисунок можно либо мышью, либо клавишами управления курсором. Первый способ (с использованием мыши) предполагает выполнение следующих действий:

- подвести указатель мыши к рисунку;
- нажать левую кнопку мыши и, не отпуская, ее перетащить рисунок на новое место.

Для реализации второго способа (с использованием клавиатуры) надо выполнить следующее:

- выделить рисунок с помощью мыши;
- с помощью курсорных клавиш (вправо, влево, вниз, вверх) подвести объект к месту нового расположения;
- если не удастся добиться точного расположения рисунка, то перемещать его следует при нажатой клавише **Ctrl**;
- снять выделение.

Следует помнить, что во втором случае перемещение рисунка происходит по узлам сетки. По умолчанию сетка не видна, и ее шаг составляет 0,32 мм, как по вертикали, так и по горизонтали. Отобразить сетку и изменить ее параметры можно с помощью диалогового окна *Привязка к сетке*. Открывает это окно команда *Действия / Сетка...* на панели инструментов *Рисование*.

Не стоит забывать, что рисунок является объектом MS Word, и для его перемещения (и, конечно, для копирования) можно использовать буфер обмена. Копировать можно точно так же, как и любые другие объекты, с помощью буфера обмена. Но есть очень удобная комбинация клавиш – **Ctrl + D** (клонирование). Пусть необходимо создать несколько одинаковых прямоугольников на одинаковом



Рис. 3.55. Клонирование прямоугольников

Для решения этой задачи требуется выполнить следующие действия. Создать первый прямоугольник и изменить его параметры (размеры, цвет и т. д.). Выделить эту фигуру и нажать комбинацию клавиш **Ctrl + D**. Чуть ниже и правее появится точная копия первого прямоугольника. Помещаем второй прямоугольник справа от первого на нужном нам расстоянии и, не снимая выделения со второй фигуры, нажимаем еще раз сочетание

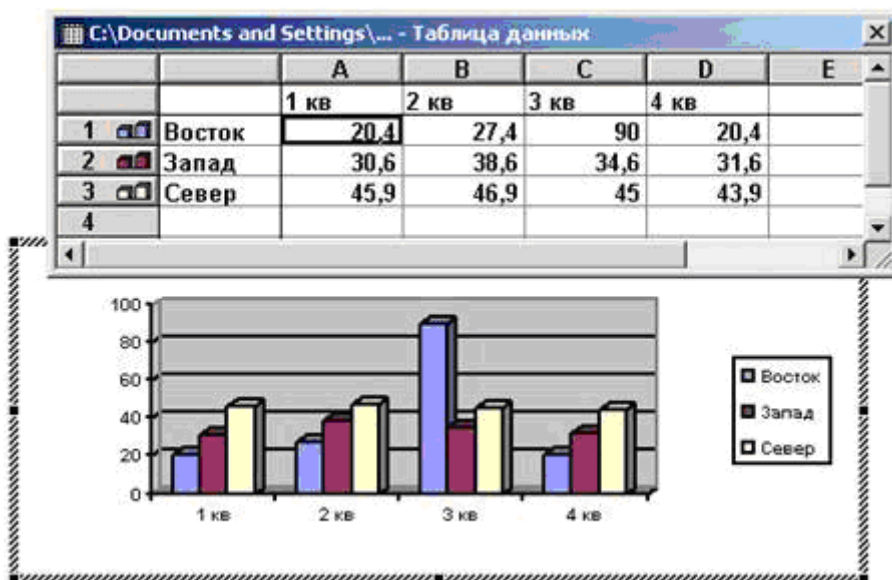
клавиш **Ctrl + D**. Справа точно на таком же расстоянии появляется третий прямоугольник. Нажмем эту комбинацию еще два раза и получаем пять одинаковых прямоугольников, расположенных в один ряд на равных расстояниях друг от друга.

3.4.15. Диаграммы

Инструменты рисования MS Word предоставляют достаточно широкие возможности для работы с графикой. Но когда требуется выводить какие-либо экспериментальные зависимости, то все инструменты рисования становятся малоэффективными.

Для графического представления экспериментальных зависимостей используется иное средство – диаграммы Microsoft Graph. Вставить данный объект можно с помощью команды меню *Вставка / Рисунок / Диаграмма*. При этом в документ будет вставлен объект, показанный на рис. 3.56. Таблица содержит данные для построения диаграммы. В ее самой верхней строке записаны названия данных (подписи на шкале абсцисс), а в самом левом столбце названия каждого набора данных. Наборы данных принято называть рядами.

Для того чтобы скрыть таблицу данных (рис. 3.56), достаточно



кликнуть мышью на кнопке в заголовке ее окна, а для отображения следует вызвать контекстное меню диаграммы и выбрать команду *Режим таблицы*. Для вызова этого меню нужно кликнуть на белой области диаграммы правой кнопкой мыши.

Рис. 3.56. Использование объекта MS Graph

Диаграммы имеют достаточно мощные инструменты для отображения данных, которые доступны из команд контекстного меню:

- *тип диаграммы* – задает вид графика отображаемых данных (рис. 3.56);

- *параметры диаграммы* – позволяет настраивать дополнительные опции диаграммы (название диаграммы, подписи осей, наличие линий сетки, отображение легенды и др.). Легенда – область с названиями рядов данных (рис. 3.56 содержит названия рядов “Восток”, “Запад”, “Север”).

Элементы диаграммы можно произвольно перемещать внутри области диаграммы. Каждый из элементов имеет собственные дополнительные параметры, доступ к которым осуществляется через контекстное меню. Вызвать его можно, нажав правую кнопку мыши на области нужного элемента диаграммы.

3.4.15.1. Построение диаграмм

Для построения диаграммы необходимо выполнить следующие действия:

- создать таблицу с данными (хотя и не обязательно);
- запустить мастер создания диаграмм с помощью команды меню *Вставка / Рисунок / Диаграмма*;
- ввести данные для построения диаграммы в таблицу (если таблица с данными была создана ранее, то сначала ее необходимо выделить и затем запустить программу Microsoft Graph);
- выбрать тип диаграммы, задать параметры для области построения диаграммы, осей и т. д.

Обычную таблицу с данными имеет смысл создавать в том случае, если на основании этих данных необходимо построить несколько диаграмм.

3.4.16. Форматирование текста с помощью стилей

Стилем называется набор параметров форматирования, который применяют к тексту документа, чтобы быстро изменить его внешний вид. Стили позволяют одним действием применить сразу всю группу атрибутов форматирования: шрифта, абзаца и т. д. Перед тем как набирать текст (или для выделенного фрагмента), необходимо установить несколько параметров шрифта и абзаца: размер, отступ первой строки, межстрочный интервал и т. д. Это легко сделать с помощью нескольких команд и диалоговых окон.

Однако то же самое можно выполнить одновременно, применив определенный стиль.

При запуске MS Word новый пустой документ создается на основе шаблона Normal.dot, а для оформления набираемого текста используется стиль *Обычный*. Таким образом, когда начинается ввод текста, автоматически используются шрифт, размер шрифта, междустрочный интервал, выравнивание текста и другие параметры форматирования, определенные в стиле *Обычный*. Этот стиль является базовым стилем шаблона Normal.dot, то есть служит основой других стилей этого шаблона. Помимо стиля *Обычный* шаблон Normal.dot содержит большое число других стилей. Некоторые основные стили представлены в раскрывающемся списке *Стиль* на панели инструментов *Форматирование*. Остальные стили можно просмотреть и выбрать в диалоговом окне *Стиль* (команда меню *Формат / Стили и форматирование...*).

Если ни один из стандартных стилей не устраивает, можно создать свой собственный. Для этого надо с помощью команды меню *Формат / Стили и форматирование...* открыть окно *Стили и форматирование*. После чего необходимо нажать кнопку *Создать стиль* и в окне *Создание стиля* (рис. 3.57) задать необходимые параметры.

В поле *Имя*: следует ввести имя стиля, из списка *Стиль*: выбрать параметр *Знака*, если хотите создать стиль знака, или *Абзаца* – для создания стиля абзаца.

Стиль абзаца полностью определяет внешний вид абзаца, то есть выравнивание текста, позиции табуляции, междустрочный интервал и границы, а также может включать форматирование символов.

Стиль знака задает форматирование выделенного фрагмента текста внутри абзаца, определяя такие параметры текста, как шрифт и размер, а также

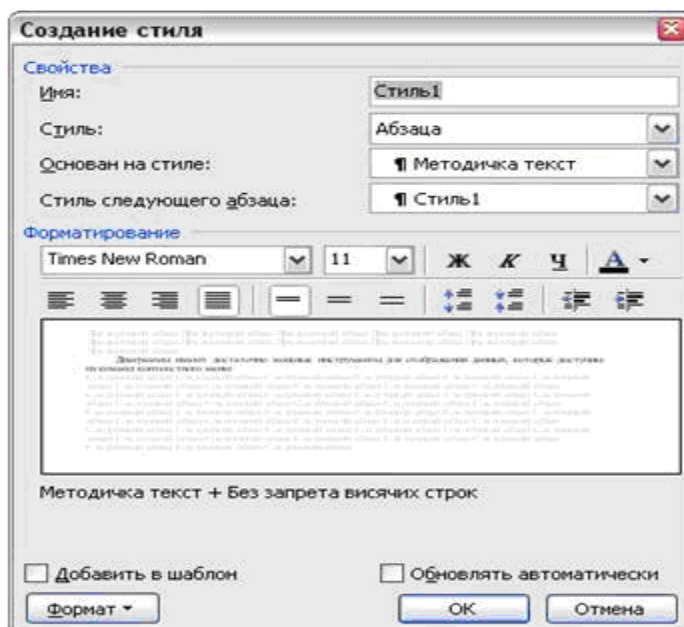


Рис. 3.57. Создание стиля

полужирное и курсивное начертание. Знаки текста в абзаце могут быть оформлены собственным стилем независимо от того, применен ли какой-либо стиль абзаца к этому абзацу в целом.

В списке *Основан на стиле*: выбирается уже существующий стиль, параметры которого наиболее близки к желаемым. Если стиль создается заново, то лучше выбрать слово (*нет*).

В выпадающем списке *Стиль следующего абзаца*: можно определить стиль, который будет применен к тексту после нажатия клавиши **Enter** при вводе текста.

Параметры шрифта и абзаца задаются с помощью стандартных диалоговых окон *Шрифт*, *Список*, *Абзац* и т. д. Вызвать эти окна можно с помощью кнопки *Формат*. После того как все параметры определены, нужно нажать кнопку *ОК*.

Быстро выбрать стиль можно с помощью выпадающего списка *Стиль* панели инструментов *Форматирование*.

3.4.17. Колонтитулы и нумерация страниц

Колонтитулы – специальные области над и под текстом документа (соответственно верхний и нижний колонтитулы), в которых может размещаться произвольная информация: текст, графика и т. д. Особенностью колонтитулов является то, что набранная в них информация копируется на каждую страницу документа.

Чтобы начать работу с колонтитулами, следует выбрать команду меню *Вид / Колонтитулы*. При этом откроется панель инструментов *Колонтитулы*, содержащая такие кнопки, как *Дата*, *Время*, *Формат номера страницы* и др. Во время ввода информации в колонтитулы редактировать текст документа нельзя. Завершение работы с колонтитулами осуществляется нажатием кнопки *Закр^ыть* на панели работы с колонтитулами.

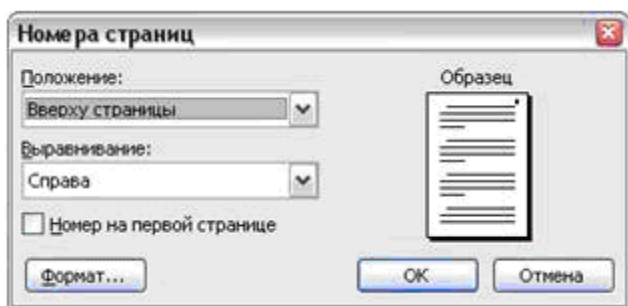


Рис. 3.58. Номера страниц

Для вставки номеров страниц в текст документа в качестве колонтитулов необходимо открыть меню *Вставка* и выбрать команду *Номера страниц...*, после чего откроется окно *Номера страниц* (рис. 3.58).

В списке *Положение*:

можно выбрать место размещения номеров страниц: вверху страницы (в верхнем колонтитуле) или внизу страницы (в нижнем колонтитуле). Если на первой странице номер не нужен, следует убрать флаг *Номер на первой странице*. В случае необходимости с помощью кнопки *Формат...* можно задать формат номера, определить, с какого значения следует начать нумерацию.

3.4.18. Оглавление

Многостраничные документы, содержащие несколько разделов, воспринимаются лучше при наличии оглавления. При машинописном оформлении работы составление оглавления занимает много времени. MS Word автоматизирует эту работу при соблюдении некоторых условий.

MS Word может создать оглавление только в том случае, если все заголовки набраны с использованием ограниченного набора стилей. Например, перед созданием документа следует для себя решить, что все заголовки будут создаваться с использованием стилей *Заголовок1*, *Заголовок2* и *Заголовок3* (трехуровневое оглавление).

Для создания оглавления необходимо выполнить следующие действия:

- 1) установить курсор в то место, где должно быть оглавление;
- 2) с помощью меню *Вставка* открыть диалоговое окно *Оглавление* (рис. 3.59);
- 3) выбрать вкладку *Оглавление*;
- 4) выбрать из предложенного списка формат оглавления;
- 5) установить количество уровней, которые должны быть включены в оглавление;
- 6) нажать кнопку *ОК*.

Однако, как правило, этих действий недостаточно. Чтобы сформировать оглавление, требуется дополнительно нажать кнопку *Параметры...* и в появившемся окне явно указать те стили, которые следует использовать в качестве заголовков, а также определить их уровни. Это легко сделать расстановкой номеров уровней (цифры 1, 2, 3, ...) в поле *Уровни:*, соответствующих каждому из стилей.

При необходимости оглавление легко обновить, установив курсор на любую его строку и нажав на клавиатуре функциональную клавишу **F9**. В готовом оглавлении можно обновить только номера страниц. Для этого следует вызвать контекстное меню ог-

лавления, выбрать команду *Обновить поле* и затем указать, что обновлять: номера страниц или оглавление целиком.

3.4.19. Печать документов

Все возможности MS Word не представляли бы такого интереса, если бы не был реализован механизм вывода документа на печать (принтер, плоттер или другое устройство).

Для вывода на печать используются две команды. Первая из них — *Печать* панели инструментов *Стандартная*

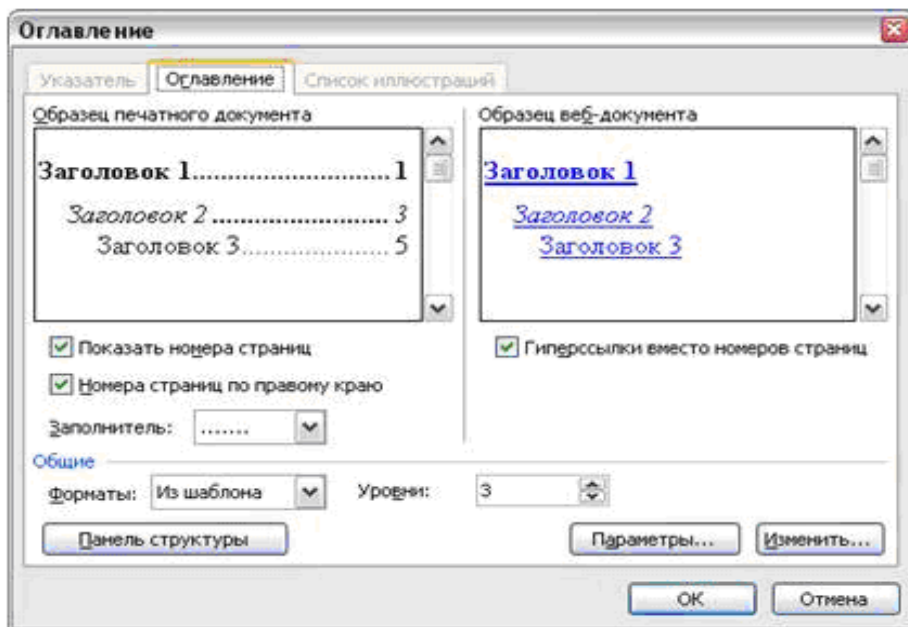


Рис. 3.59. Создание оглавления

позволяет организовать печать всего документа целиком. Если не-

обходимо вывести на печать только часть документа или организовать вывод специфически, то следует воспользоваться командой меню *Файл / Печать...* (эта команда также доступна сочетанием клавиш **Ctrl + P**). При этом на экран выводится окно, изображенное на рис. 3.60, в котором

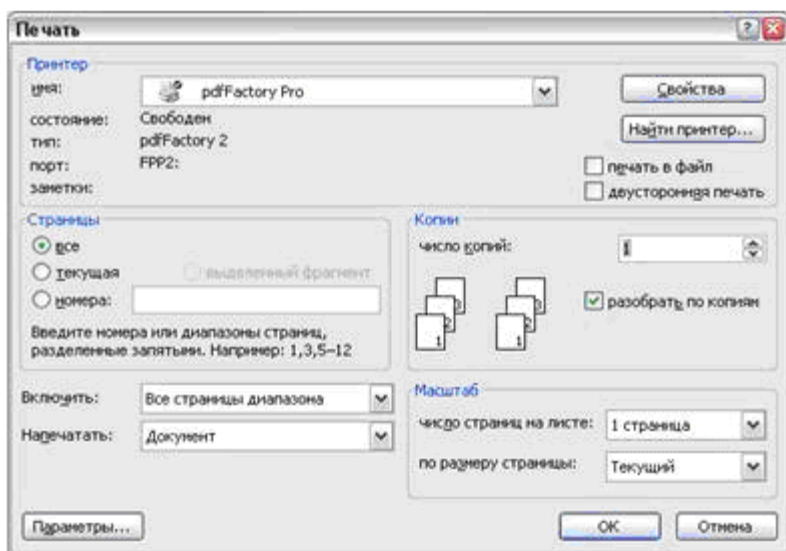


Рис. 3.60. Печать документа

можно уточнить параметры вывода документа на печать.

Верхняя часть окна содержит информацию о принтере (принтерах для случая, когда в системе установлено несколько печатающих устройств), который подключен к компьютеру. Если нажать на кнопку *Свойства*, то откроется окно со свойствами принтера, которые при необходимости можно изменить.

Нижняя часть диалогового окна *Печать* имеет непосредственное отношение к печатаемому документу. Слева располагается группа команд *Страницы*. По умолчанию MS Word распечатывает все страницы документа (выбран переключатель *все*). Но можно распечатать только текущую страницу, для чего требуется установить переключатель положение *текущая*. Существует возможность печати заданных страниц одного или нескольких разделов, а также диапазона страниц. Для этого в группе страницы надо установить переключатель в положение *номера* и ввести номера страниц, разделяя их запятыми. Если требуется ввести диапазон страниц, то следует разделить дефисом номер первой и последней страниц этого диапазона. Например, если требуется вывести на печать 2, 5, 7, 8, 9, 10, 16 страницы, то в поле номера можно ввести следующую строку “2,5,7-10,16”.

Группа команд *Копии* позволяет задать требуемое количество копий документа. Флаг *разобрать по копиям* устанавливается в том случае, когда нужно распечатать сначала целиком первую копию документа, затем вторую и т. д. В противном случае будут распечатываться сначала несколько копий первой страницы, затем несколько копий второй и т. д.

Прежде чем отправить документ на печать, нужно предварительно посмотреть, как будет выглядеть документ на бумаге. Просмотреть документ можно с помощью кнопки *Предварительный просмотр* панели инструментов *Стандартная*. Для выхода из режима предварительного просмотра следует нажать кнопку *Закрыть* появившейся панели инструментов *Предварительный просмотр*.

3.4.20. Контрольные вопросы

- 1) Перечислите элементы рабочего окна Microsoft Word.
- 2) Что позволяет установить задание стиля в текстовом редакторе MS Word?

- 3) Назовите типы документов, создаваемых современными текстовыми процессорами.
- 4) Как называется режим, не являющийся режимом представления документов?
- 5) Когда возможно изменение параметров страницы?
- 6) Какие параметры страниц может иметь каждый раздел документа MS Word?
- 7) Для чего используются шаблоны в MS Word?
- 8) Каким образом устанавливается ориентация листа бумаги документа MS Word?
- 9) Что нельзя переслать в буфер обмена?
- 10) К выделению чего приводит двойной щелчок левой кнопки мыши перед словом в MS Word?
- 11) Что будет выровнено при задании типа выравнивания «по правому краю» для выделенного слова?
- 12) Каким образом можно выделить группу объектов в документе MS Word?

3.5. Средства создания и использования презентаций MS PowerPoint

Приложение MS PowerPoint, входящее в состав офисного пакета MS Office, является универсальной системой подготовки презентаций и слайд-фильмов.

Презентация – это набор слайдов и спецэффектов, сопровождающих их показ на экране, раздаточный материал, конспект и план доклада, хранящиеся в одном файле, созданном средствами MS PowerPoint.

Слайд – это отдельная страница (кадр) презентации, которая может содержать заголовок, текст, графику, диаграмму, рисунок и другие элементы иллюстративной графики, аудио- и видео информацию.

Раздаточный материал – это распечатанные в компактном виде слайды (от 2 до 9 слайдов на одной странице).

Конспект доклада (заметки докладчика) – уменьшенное изображение слайда и текст, поясняющий его содержание.

План доклада (структура презентации) – документ, содержащий только заголовки слайдов и основной текст без графических изображений и специального оформления.

Открытие (запуск) приложения MS PowerPoint осуществляется аналогично другим приложениям MS Office.

3.5.1. Основные возможности приложения MS PowerPoint

3.5.1.1. Главное окно

Окно MS PowerPoint представлено на рис. 3.61, где цифрами обозначены следующие элементы:

- 1) меню программы MS PowerPoint;
- 2) строка состояния (статус-строка);
- 3) панели инструментов;
- 4) область структуры презентации;
- 5) текущий (или активный) слайд: рабочая область, в которой непосредственно создается оформление выбранного слайда презентации;
- 6) область задач MS PowerPoint (представляет собой область в окне приложения MS Office, содержащую часто применяемые команды, использовать которую можно, не прекращая работы с файлом), содержащая команды получения справки по MS PowerPoint, а также другие ссылки;
- 7) кнопки переключения режимов отображения слайдов.

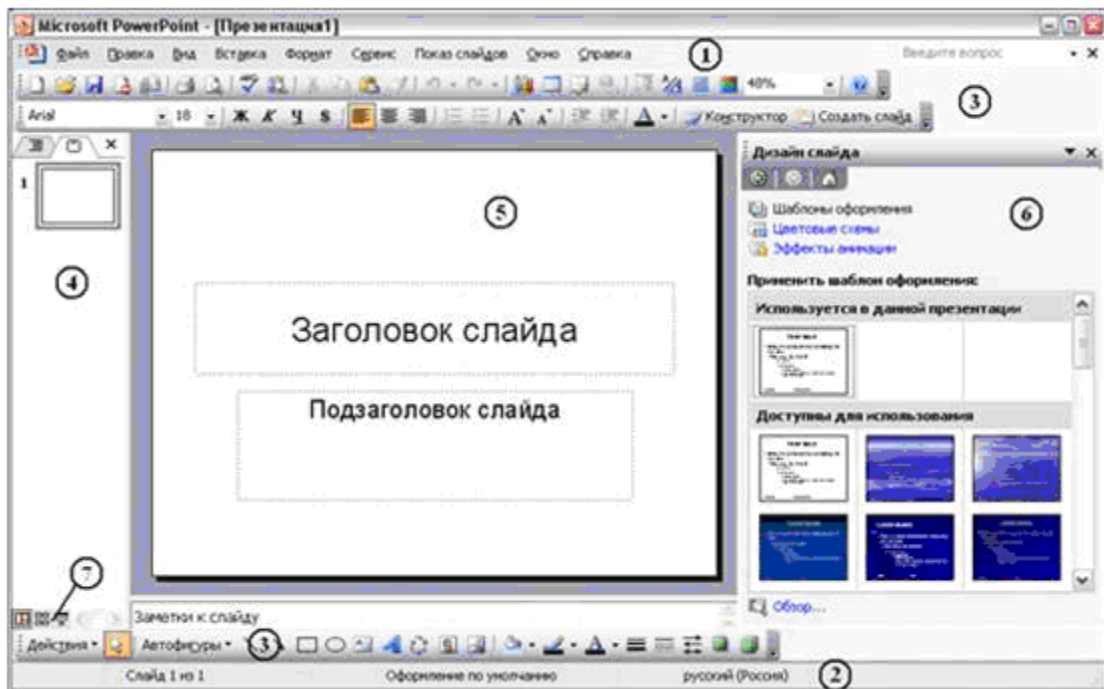


Рис. 3.61. Главное окно MS PowerPoint

Такие элементы, как меню, панели инструментов, строка состояния, частично были рассмотрены ранее в предыдущих главах данного пособия. Их использование в MS PowerPoint выполняется аналогично, поэтому останавливаться на изучении работы с ними здесь не будем. Большой интерес представляют другие возможности MS PowerPoint, о которых и поговорим подробнее.

Область 4 представляет собой мощный инструмент управления слайдами внутри презентации. Если она не отображена на экране, ее можно включить, выбрав команду меню *Вид / Обычный (восстановить области)*.

Область структуры презентации содержит две вкладки.

- 1) *Структура* – помогает осуществлять редактирование содержания презентации и перемещать маркированные пункты списка или слайды. Если сохранить презентацию как веб-страницу, то текст данной области становится оглавлением, что позволяет перемещаться от слайда к слайду.
- 2) *Слайды* – на этой вкладке последовательно отображаются все эскизы страниц (слайдов) презентации и соответствующие им номера, определяющие положение в презентации. С помощью нее пользователь может изменять порядок следования слайдов, устанавливать некоторые эффекты анимации.

С помощью кнопок области 7 можно изменять режимы отображения слайдов. Доступны три режима:

- 1) *Обычный режим*;
- 2) *Режим сортировщика слайдов*;
- 3) *Показ слайдов (с текущего слайда)*.

3.5.1.2. Обычный режим

Обычный режим – это основной режим редактирования, который используется для создания презентации. При работе в этом режиме доступны три рабочие области. Левая область содержит вкладки, позволяющие переходить от структуры презентации (вкладка *Структура*) к слайдам, отображаемым в виде эскизов (вкладка *Слайды*). В правой области (*область слайдов*) отображается доступный для редактирования текущий слайд. Нижняя область (*область заметок*) в обычном режиме используется для создания заметок к слайду, которые можно распечатать. Заметки отображаются на экране при сохранении презентации в формате веб-страницы.

3.5.1.3. Режим сортировщика слайдов

Режим сортировщика слайдов – это монопольное представление слайдов в форме эскиза. По окончании создания и редактирования презентации сортировщик слайдов дает общую картину презентации, облегчая изменение порядка следования слайдов, их добавление или удаление, а также просмотр эффектов перехода и анимации.

3.5.1.4. Показ слайдов

В режиме показа слайдов занят весь экран компьютера, как при реальной презентации. Презентация отображается во весь экран так, как она будет представлена аудитории. В этом режиме можно посмотреть, как будут выглядеть рисунки, временные интервалы, фильмы; анимированные элементы (добавление к тексту или объекту специального видео- или звукового эффекта, например, влетающие на страницу слева слова некоторого предложения и другое) и эффекты перехода, которые будут отображаться в реальном виде. Данный режим удобен при форматировании уникального образа презентации. Для его включения используется сочетание клавиш **Shift + F5**.

3.5.2. Настройки и параметры MS PowerPoint

Для изменения параметров MS PowerPoint, установленных по умолчанию, следует выбрать команду меню *Сервис / Параметры...* Окно параметров приложения, открывающееся при выборе этой команды, изображено на рис. 3.62.

Наибольший интерес в окне *Параметры* представляют две вкладки: *Вид* и *Сохранение*.

На последней из них расположено средство, позволяющее обезопасить документы в случаях проблем с сетью питания. Укажите нужный интервал (в минутах), через который MS PowerPoint будет выполнять автоматическое сохранение файла на диск, после включения опции (флага) *автосохранение каждые*.

С помощью вкладки *Вид* можно определить интерфейсные элементы, которые будут отображаться в главном окне MS PowerPoint при каждом запуске. Например, в представленном на рис. 3.62 окне с помощью флагов определено, что при запуске

MS PowerPoint будут отображаться такие элементы управления, как *область задач*, *область* “*Разметка слайда*”, *вертикальная линейка* и другое.

В MS PowerPoint присутствует средство проверки орфографии. По умолчанию оно включено, но в случае необходимости его можно и отключить (включить, если было отключено ранее). Для этого следует перейти на вкладку *Орфография и стиль* и изменить состояние флага *автоматически проверять орфографию*.

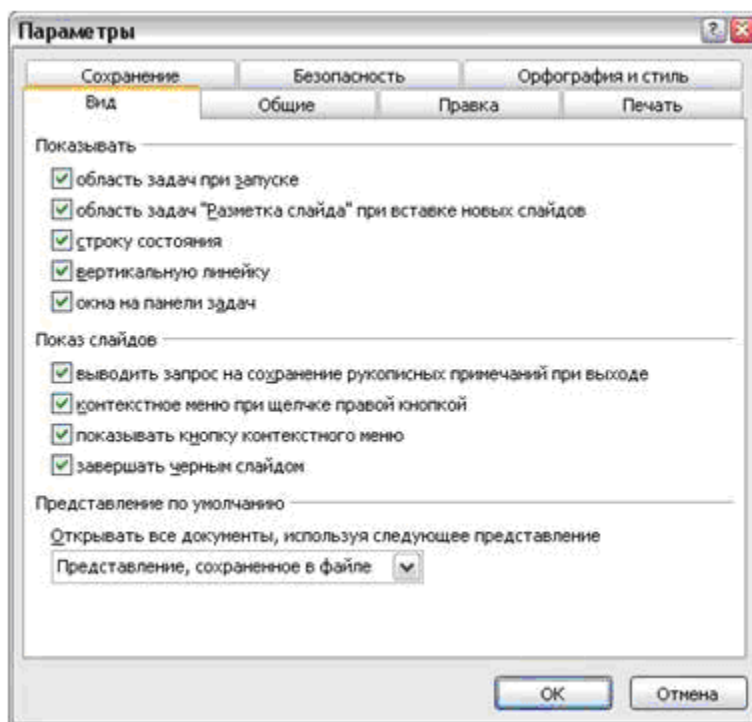


Рис. 3.62. Параметры MS PowerPoint

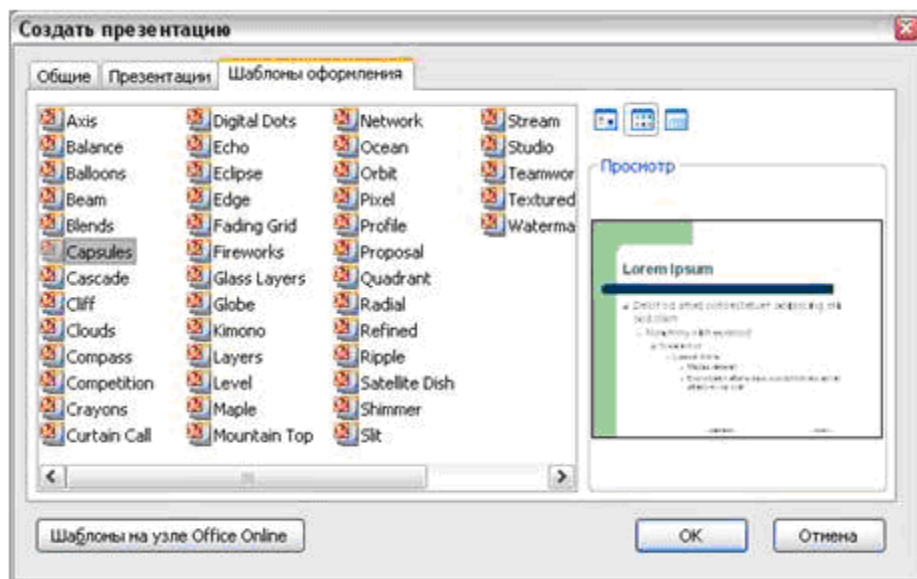
3.5.3. Общие принципы работы с презентациями

В процессе создания презентаций осуществляется работа с несколькими объектами, к числу которых относятся: презентация, слайды (или листы), объекты слайда (непосредственно элементы оформления). Рассмотрим основные принципы работы с этими объектами.

3.5.3.1. Создание и сохранение презентаций

Следует отметить, что создание, открытие и сохранение файлов MS PowerPoint осуществляется точно так же, как и файлов приложений MS Word или MS Excel. То есть для сохранения презентации используются команды меню *Файл / Сохранить*, *Файл / Сохранить как...* или сочетание клавиш **Ctrl + S**. Чтобы открыть презентацию, достаточно дважды кликнуть мышью на имени файла, а также можно использовать команду меню *Файл / Открыть...* или сочетание клавиш **Ctrl + O**.

Для создания новой презентации следует выполнить команду меню *Файл / Создать...*, исполнение которой приведет к открытию



области задач MS Office *Создание презентации*. Если в данной области презентации выбрать команду *На моем компьютере...*, то откроется диалоговое окно *Создать презентацию*, используя которое для вновь создаваемой презентации можно определить шаблон

Рис. 3.63. Выбор шаблона презентации

лон (рис. 3.63). Шаблон (образец слайда) представляет собой слайд, содержащий определенные элементы оформления, включая параметры шрифтов, размеры и размещение рамок, параметры фона и цветовые схемы.

3.5.4. Оформление слайдов

Оформление (дизайн) слайдов презентации выполняется с помощью *шаблонов*. Использовать шаблон можно как для оформления отдельного слайда, так и всей презентации. Для этого используется область задач *Дизайн слайда*. Чтобы применить шаблон, нужно навести курсор мыши на его изображение в области задач *Дизайн слайда*. При этом с правой стороны изображения появится область со стрелкой, нажатие на которую приведет к распаиванию контекстного меню, содержащего следующие команды:

- 1) *применить ко всем слайдам* задает шаблон оформления для всех созданных, а также для всех вновь создаваемых слайдов в текущей презентации;
- 2) *применить к выделенным слайдам* изменяет шаблон оформления только для выделенных слайдов;

3) *использовать для всех новых презентаций* назначает шаблон, как используемый по умолчанию для всех новых презентаций.

Для выделения слайдов используется область структуры презентации (область 4 на рис. 3.61). Выделение выполняется мышью, при нажатых клавишах **Ctrl** и/или **Shift**, аналогично выделению ячеек рабочего листа MS Excel или строк документа MS Word.

Помимо дизайна слайда, MS Power Point позволяет определить разметку слайда, для чего используется область задач *Разметка слайда*. Разметка слайда представляет собой оформление элементов, таких как заголовки, подзаголовки, списки, рисунки, таблицы, диаграммы, автофигуры и видеофрагменты на слайде. Механизм применения разметки к слайдам аналогичен использованию шаблонов. Существуют четыре группы разметки слайдов: макеты текста, макеты содержимого, макеты текста и содержимого и другие макеты.

Термин *макет* относится к расположению объектов на слайде. В макетах содержатся рамки (местозаполнители – это поля с пунктирными границами, являющиеся частью макетов большинства слайдов), которые включают текст, такой как заголовки слайдов или маркированные списки, и основное содержимое слайда: таблицы, диаграммы, рисунки, фигуры и картинки. Например, на рис. 3.61 созданы две рамки для размещения заголовка и подзаголовка слайда.

Основной макет с рамкой используется обычно для заголовка и маркированного списка. Макет с тремя рамками – для заголовка, маркированного списка и содержимого слайда (таблиц, схем, диаграмм и картинок).

Макет нового слайда можно выбрать в области задач *Разметка слайда* каждый раз при его добавлении. Расположение рамок внутри макета можно менять: изменять их размеры, добавлять заливку и границы. Такие изменения применяются в слайдах и страницах заметок (распечатанные страницы с авторскими заметками, расположенные под соответствующими им слайдами).

В выдачах (раздаточные материалы – распечатанная презентация, которая может содержать на одном листе несколько слайдов и поле для самостоятельной записи заметок) можно изменять только рамки колонтитулов. Можно также вносить подобные изменения в отдельные слайды и страницы заметок, а также в образец слайдов, образец заметок или образец выдач. Для согласованности оформления во всей презентации желательно вносить изменения в образцы.

После изменения рамок на страницах заметок или на слайде может понадобиться вернуться к исходному макету. Для этого следует повторно применить исходный макет. При удалении рамок из образца можно восстановить исходный образец с помощью команды меню *Формат / Разметка слайда...*

3.5.5. Работа со слайдами

К работе со слайдами обычно относят те приемы, которые позволяют изменять состав презентации и порядок следования слайдов во время показа. Рассмотрим возможные приемы работы со слайдами.

3.5.5.1. Основной показ

Основным показом называются все страницы (слайды) презентации MS PowerPoint. Со слайдами основного показа можно выполнить такие действия, как добавление нового слайда, его удаление, изменение порядка следования слайдов в презентации, скрытие слайдов и другое.

Создать новый слайд можно несколькими способами. Например, можно кликнуть правой кнопкой мыши в области 4 (рис. 3.61) и в контекстном меню выбрать команду *Создать слайд*. Либо можно нажать такую же кнопку (*Создать*) на панели инструментов MS PowerPoint.

Для удаления слайда следует выделить его эскиз (в области 4 на рис. 3.61) и нажать кнопку **Del** на клавиатуре. Также удалить слайд можно и с помощью мыши, для чего нужно кликнуть правой кнопкой на эскизе слайда и в контекстном меню выбрать команду *Удалить слайд*.

Изменение порядка следования слайдов в основном показе презентации осуществляется перетаскиванием нужного эскиза слайда в требуемое положение. Для перетаскивания достаточно нажать левой кнопкой мыши на эскизе и, не отпуская кнопку, перемещать указатель в области *Слайды*. Новое положение слайда в презентации будет отмечено горизонтальной линией. Чтобы завершить перемещение, следует отпустить кнопку мыши.

Если текст, размещаемый на слайде, достаточно большой, то можно воспользоваться свойством MS PowerPoint, позволяющим

разбить этот фрагмент на два слайда. Для этого, находясь в обычном режиме, следует поместить курсор в области *Структура* в положение, с которого требуется начать разбиение текста (например, в конце маркированного абзаца), и нажать клавишу **Enter**. Далее на панели инструментов *Форматирование* нужно нажимать кнопку *Уменьшить отступ* до тех пор, пока не появится значок и номер нового слайда (часть текста появится под значком в виде основного текста), а затем останется только ввести заголовок нового слайда.

Существует и другой способ разбиения текста на два слайда. Если при вводе текста в текстовую рамку на макете слайда с одним столбцом он превышает размеры рамки, слева от текста появляется кнопка *Параметры автоподбора ширины*. При ее нажатии на экране отображаются варианты автоподбора: разбиение текста на два слайда и создание второго пустого слайда с таким же заголовком, на котором можно ввести дополнительный текст.

Скрытый слайд – слайд, который присутствует в файле, но не отображается во время показа. Чтобы скрыть слайд или слайды в презентации, достаточно выделить их эскизы и выбрать команду меню *Показ слайдов / Скрыть слайд*. Повторное использование данной команды делает слайд отображаемым. Иногда требуется все же показать скрытый слайд во время показа презентации. Для этого следует во время показа кликнуть правой кнопкой мыши на любом слайде и выбрать команду *Переход*, а затем кликнуть мышью на номере нужного слайда (номера скрытых слайдов показаны в скобках).

3.5.5.2. Произвольные показы

Одну и ту же презентацию можно адаптировать для разных аудиторий, создав произвольные показы. *Произвольный показ* – это либо слайды, сгруппированные в презентацию, которые можно продемонстрировать независимо от всего показа, либо группы слайдов в презентации, на которые созданы гиперссылки.

Основной произвольный показ может потребоваться, например, для того чтобы продемонстрировать презентацию двум группам сотрудников организации, работающим в разных отделах. Показ слайдов включает слайды с 1 по 5. Для первой группы можно создать произвольный показ по имени «Отдел 1», включающий слайды 1, 3, и 5, а для второй – произвольный показ «Отдел 2», включаю-

щий слайды 1, 2, 4 и 5. Разумеется, в любой момент также можно запустить показ слайдов в их исходной последовательности.

Для создания произвольного показа следует выбрать команду меню *Показ слайдов / Произвольный показ...* и нажать кнопку *Создать...* После этого на экране будет открыто два окна, как показано на рис. 3.64. Далее следует выполнить следующие действия:

- 1) в области *Слайды презентации* (область 4 на рис. 3.61) выбрать слайды (выбор нескольких слайдов осуществляется при нажатой клавише **Ctrl**), которые следует включить в произвольный показ, и затем нажать кнопку *Добавить>>*;
- 2) для изменения порядка показа слайдов достаточно переместить нужный слайд в списке *Слайды произвольного показа*, нажав одну из стрелок (вверх или вниз);
- 3) ввести имя в поле *Имя произвольного показа* и нажать кнопку *OK*;
- 4) для создания других произвольных показов, содержащих любые слайды презентации, следует повторить шаги с 1 по 3.

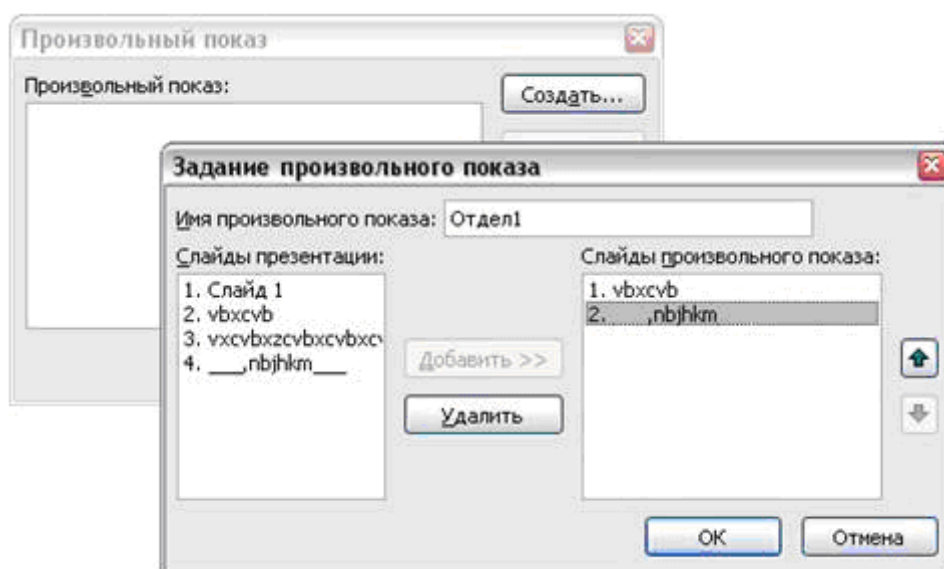


Рис. 3.64. Создание произвольного показа

Произвольные показы со ссылками используются для организации содержимого показа на основе гиперссылок. Например, можно создать основной произвольный показ о структуре всей организации целиком. Затем можно создать произвольные показы по структуре отдельных отделов организации (например, отделов А и Б) и организовать ссылки на эти показы из основного.

Чтобы создать произвольный показ со ссылками, нужно вначале создать один или несколько произвольных показов, а затем в основном показе определить гиперссылки на созданные произвольные показы, для чего требуется выполнить следующие действия:

- 1) выбрать (выделить) текст или объект, который должен представлять гиперссылку;
- 2) нажать кнопку *добавление гиперссылки*;
- 3) в области *Связать с*: выбрать кнопку *местом в документе*;
- 4) для связи с произвольным показом следует выбрать из списка произвольный показ, к которому требуется перейти, и установить флажок *Показать и вернуться*;
- 5) для связи со слайдом в текущей презентации нужно выбрать из списка слайд, к которому требуется перейти.

Выполнить предварительный просмотр произвольного показа можно с помощью команды *Показать* в диалоговом окне *Произвольный показ*.

3.5.5.3. Демонстрация произвольного показа

Демонстрация произвольного показа служит для отображения только слайдов произвольного показа, а не всей презентации. Для ее исполнения следует выбрать команду меню *Показ слайдов / Настройка презентации...*, затем в группе *Слайды* указать вариант *Произвольный показ*, выбрать имя этого показа из списка и нажать кнопку *ОК*. После этого можно запускать показ, например, нажав клавишу **F5**.

3.5.6. Добавление текста

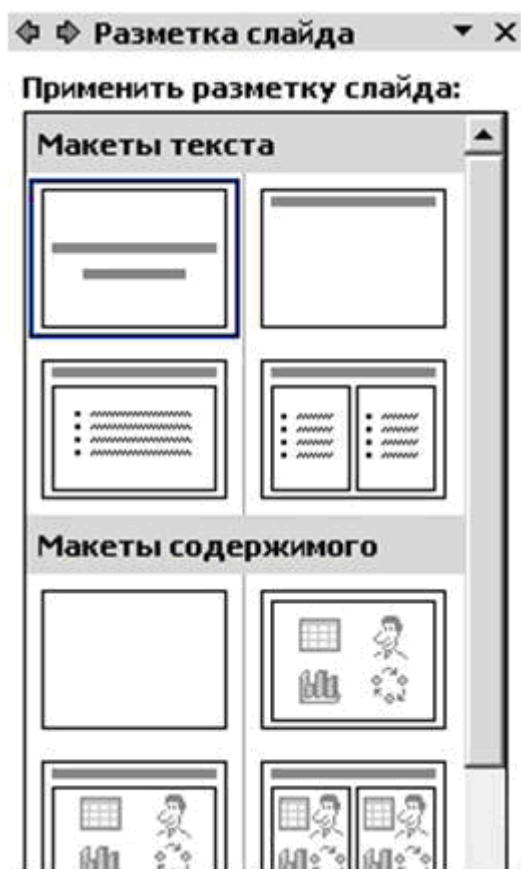
Возможно размещение на слайде четырех типов текста: текст в *рамке* (поля с границами в виде точек или штрихов, которые являются составной частью большинства макетов слайдов); текст в *Автофигурах*, текст в *Надписях* и текст *WordArt*.

Макет слайда может содержать рамки не только для текста, но и для других объектов в различных комбинациях. В соответствующие рамки вводятся заголовки, подзаголовки или основной текст слайда.

Выберите команду *Создать слайд* в меню *Вставка* или на панели инструментов *Форматирование*. В результате справа

откроется область задач *Разметка слайда* (рис. 3.65), в которой представлены макеты слайдов, разделенные на несколько групп.

Первой идет группа *Макеты текста*. Представленные здесь



макеты разбиты на участки с помощью рамок, в которых и должен размещаться текст. Текст, вводимый в рамках, таких как заголовки и маркированные списки, можно изменить на слайде или в области задач *Структура*. Его можно экспортировать и из документа MS Word.

Текст в объектах, таких как надписи или автофигуры, а также текст WordArt не отображается в области *Структура*, и его следует редактировать непосредственно на слайде.

Автофигуры представляют собой стрелки, блок-схемы, выноски и т. д. Если открыть панель инструментов *Рисование*, то можно увидеть несколько элементов меню, распо-

Рис. 3.65. Разметка слайда

ложенных на ней. Например, меню *Автофигуры* позволяет выбрать любой объект из достаточно большого списка. На выбранной автофигуре необходимо нажать правую кнопку мыши и из контекстного меню выбрать команду *Добавить текст*. Этот текст становится вложенным в фигуру и перемещается, а также поворачивается вместе с ней.

Другой элемент – *надписи*, которые используются для размещения текста в любом месте слайда, в том числе за пределами текстовых рамок. Например, можно добавить заголовков к рисунку, создав надпись и разместив ее около рисунка. Также надписями можно воспользоваться для добавления текста к автофигурам, когда его не требуется связывать с ней. Надпись может иметь границы, заливку, тень и трехмерные (объемные) эффекты. Также можно изменить и ее форму.

Текст WordArt позволяет создать яркие эффекты. Его можно растягивать, скашивать, размещать по изогнутой линии и вращать, а также делать объемным и вертикальным.

Кроме текста, можно вставить в слайд и другие объекты: таблицы, диаграммы и т.д. Для этого можно использовать команды меню *Вставка*. Некоторые команды (*Добавить таблицу*, *добавление диаграммы* и т. д.) вынесены на панель инструментов *Стандартная*. Можно воспользоваться готовыми макетами, представленными в области задач *Разметка слайда*. Группы *Макеты содержимого* и *Макеты содержимого и текста* имеют панель инструментов с шестью командами:

- 1) *Добавить таблицу*;
- 2) *Добавить диаграмму*;
- 3) *Добавить картинку*;
- 4) *Добавить рисунок*;
- 5) *Добавить диаграмму*;
- 6) *Вставить клип из коллекции*.

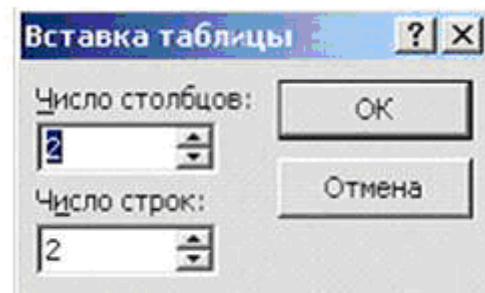


Рис. 3.66. Вставка таблицы

3.5.7. Вставка таблиц

Как уже упоминалось выше, существуют различные способы добавления таблиц в презентацию. Таблицу можно добавить в MS PowerPoint из другого приложения в виде *связанного объекта* – объекта, созданного в файле-источнике и вставленного в файл-приемник с поддержанием связи между этими двумя файлами. Связанный объект в файле назначения может быть обновлен при обновлении файла-источника. Также таблицу можно добавить в виде *внедренного объекта* – объекта, содержащегося в файле-источнике и вставленного в файл назначения без установления связи. После внедрения объект становится частью файла назначения. Изменения, вносимые во внедренный объект, отражаются в файле назначения.

В MS PowerPoint можно создать как простую таблицу с несложным форматированием, так и более сложную с использованием большего количества параметров.

Если нужно создать таблицу больших размеров или с более широкими возможностями форматирования, чем те, которые доступны в MS PowerPoint, можно создать ее в MS Word и затем внедрить. Мож-

но также внедрить электронную таблицу из MS Excel или из MS Access. Текст из таблицы не отображается в области задач *Структура*. Для создания таблицы нужно выделить соответствующий слайд и нажать кнопку *Добавить таблицу* на панели инструментов *Стандартная*, затем с помощью указателя мыши выбрать требуемое количество строк и столбцов, щелкнув на них.

Создать таблицу можно также с помощью соответствующей команды меню *Вставка / Таблица...*. В составе MS PowerPoint имеются макеты слайдов, содержащие пустые рамки для таблиц. Для создания нового слайда с помощью одного из этих макетов выберите в меню *Вставка* команду *Создать слайд*, а затем укажите один из макетов, содержащий рамку для таблицы. В обоих случаях откроется окно *Вставка таблицы* (рис. 3.66), в котором требуется указать число строк и столбцов. В результате этих манипуляций на слайде появляется заготовка таблицы (рис. 3.67) и панель инструментов *Таблицы и границы*. Работа с таблицами в MS PowerPoint ничем не отличается от работы с ними в MS Word.

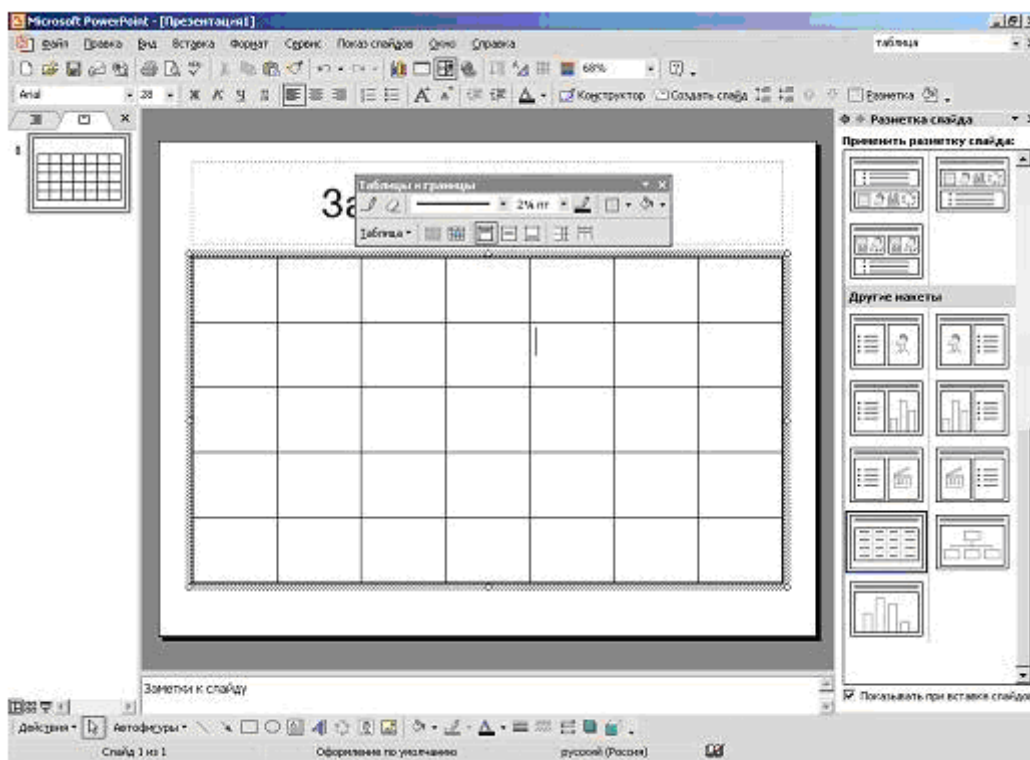


Рис. 3.67. Слайд с таблицей

3.5.8. Добавление клипов

Клипами принято называть рисунки, фотографии, звуки, видео и другие файлы мультимедиа, которые можно вставлять и использовать в презентациях, публикациях и других документах MS Office.

Для поиска, добавления и упорядочения мультимедиа-клипов в приложениях MS Office используется такое средство, как коллекция клипов (рис. 3.68). Открыть область задач *Коллекция клипов* можно командой меню *Вставка / Рисунок / Картинки...*

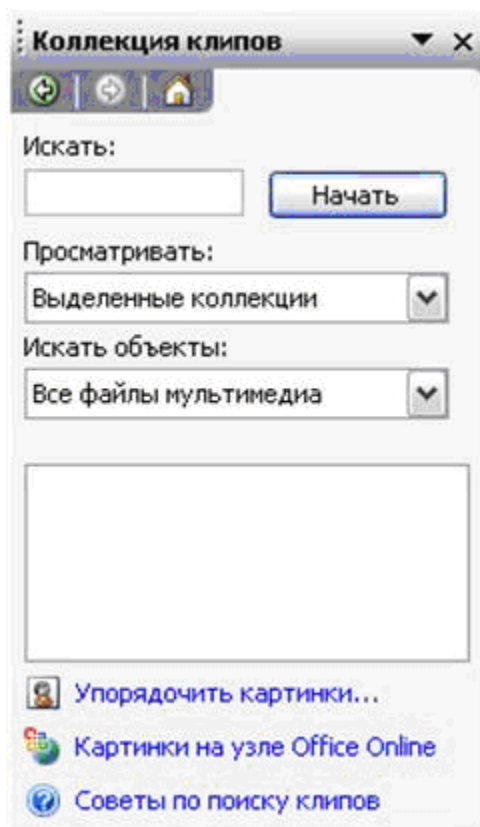


Рис. 3.68. Вставка картинок

Поиск клипов мультимедиа можно выполнять по описательным ключевым словам, имени файла, формату файла и коллекциям клипов.

Коллекцию клипов можно открыть с помощью команды *Упорядочить картинки...* главного окна *Коллекция клипов* (рис. 3.68). В появившемся окне (рис. 3.69) можно просматривать все доступные картинки, добавлять клипы и структурировать их удобным образом. Например, можно создать коллекцию, объединяющую наиболее часто используемые клипы, или

включить режим автоматического добавления и структурирования файлов мультимедиа на жестком диске.

Еще одним средством работы с рисунками является коллекция картинок в интернете. При наличии открытого подключения к сети в результате поиска клипов автоматически включается содержимое коллекции картинок в интернете. Этот веб-узел также можно посетить в любое время, щелкнув соответствующую ссылку в нижней части области задач. При наличии подключения к интернету можно использовать средства *Office в интернете* для заказа различных компонентов для работы с клипами.

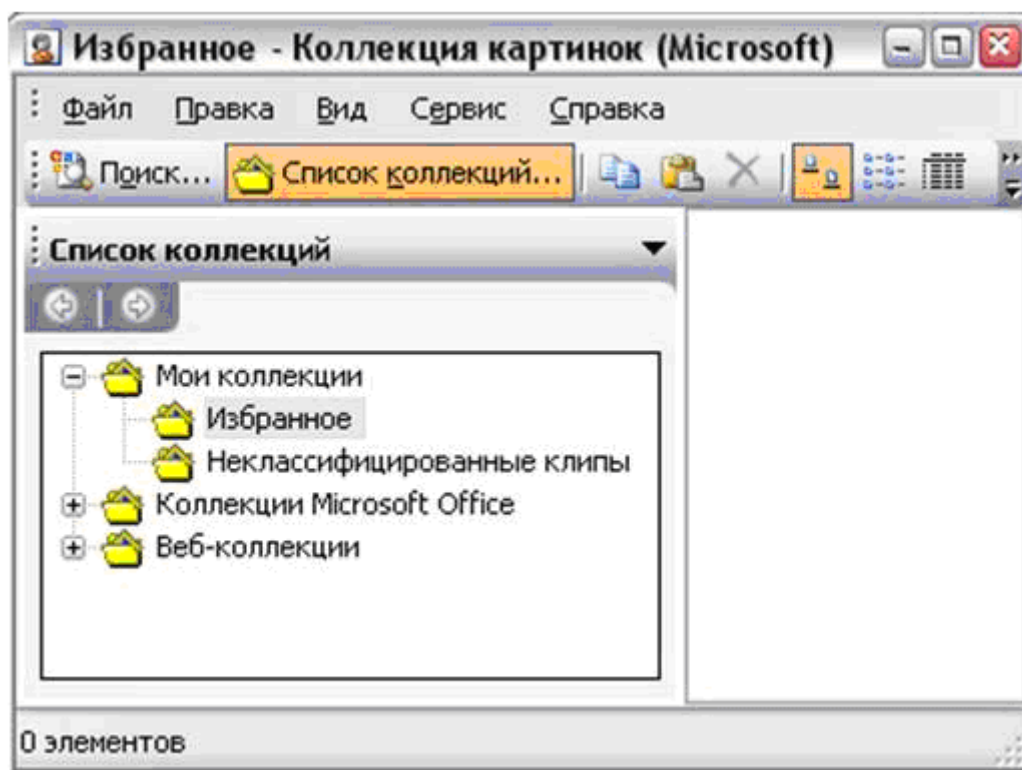


Рис. 3.69. Окно для работы с коллекциями клипов

Еще одним элементом оформления презентаций являются фильмы и анимированные рисунки GIF. Фильмами называются фоновые видеофайлы в формате AVI, Quick Time и MPEG, имеющие в качестве расширения имен AVI, MOV, QT, MPG и MPEG. Обычно фильм включает речь докладчика (например, реферат), который не может лично присутствовать на презентации. Фильм можно использовать также для демонстрации способа обучения.

Анимированный файл формата GIF также позволяет отобразить движущееся изображение. Анимированные рисунки GIF не являются настоящими фильмами, однако они содержат несколько изображений, которые сменяют друг друга, создавая эффект анимации. Файлы этого формата часто используются для привлечения внимания к оформлению или к веб-узлу.

Фильмы и анимированные рисунки GIF можно добавлять на слайды из файлов, имеющих на компьютере, из коллекции картинок Microsoft, из локальной сети, интранета или из сети Интернет. Для добавления фильма или анимационного рисунка GIF его следует вставить на отдельный слайд с помощью команды меню *Вставка / Рисунок / Из файла....*

Существует несколько способов запуска вставленного видео-файла: он может воспроизводиться автоматически при показе слайда, по щелчку мыши или через определенный интервал времени. Он также может быть частью анимационного эпизода.

3.5.9. Диаграммы

Диаграммы представляют собой средство наглядного отображения данных. Чтобы вставить в презентацию диаграмму, необходимо выполнить следующие действия:

- 1) выделить слайд, на который требуется поместить диаграмму;
- 2) в меню *Вставка* выбрать команду *Диаграмма...* ;
- 3) в таблицу ввести собственные данные (рис. 3.70);
- 4) с помощью команд контекстного меню выбрать тип диаграммы, изменить ее параметры;
- 5) вернуться к слайду, для чего достаточно щелкнуть за пределами диаграммы.

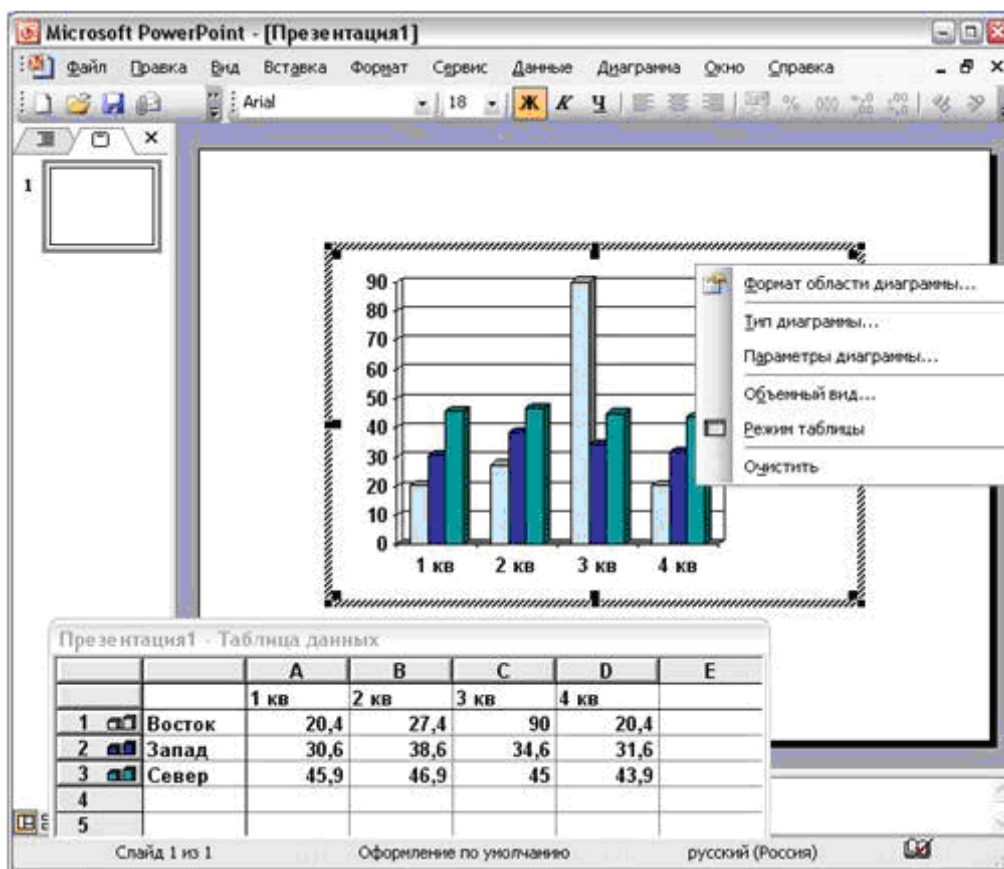


Рис. 3.70. Работа с диаграммами

Работа с диаграммами в MS PowerPoint практически ничем не отличается от работы с диаграммами в MS Word. Поэтому больше внимания следует уделить так называемым *организационным диаграммам*. Существует несколько различных типов диаграмм:

- 1) *организационные диаграммы* используют для графического описания иерархических отношений в организациях, например, между руководителями отделов и сотрудниками;
- 2) *диаграммы Венна* применяются для отображения областей перекрывания элементов;
- 3) *циклические диаграммы* позволяют проиллюстрировать процессы, имеющие непрерывный цикл;
- 4) *пирамидальные диаграммы* отображают, как правило, базисно-надстроечные связи;
- 5) *целевые диаграммы* используются для отображения шагов, ведущих к намеченной цели;
- 6) *радиальные диаграммы* применяются для отображения связей отдельных элементов с основным.

Для создания любой диаграммы можно нажать кнопку *Добавить диаграмму* или *организационную диаграмму* на панели инструментов *Рисование* или выбрать команду меню *Вставка / Организационная диаграмма...*, а затем в диалоговом окне *Библиотека диаграмм* (рис. 3.71) выбрать тип диаграммы и нажать кнопку *ОК*.

Для создания только организационной диаграммы можно также выполнить команду меню *Вставка / Рисунок / Организационная диаграмма*. После чего на листе появится рамка со штриховыми границами (*полотно*) и заготовка (*макет*) диаграммы, как показано на рис. 3.72, а также панель инструментов *Диаграмма* (при создании циклической, целевой, радиальной, пирамидальной диаграммы или диаграммы Венна) или *Организационная диаграмма* (при создании организационной диаграммы).

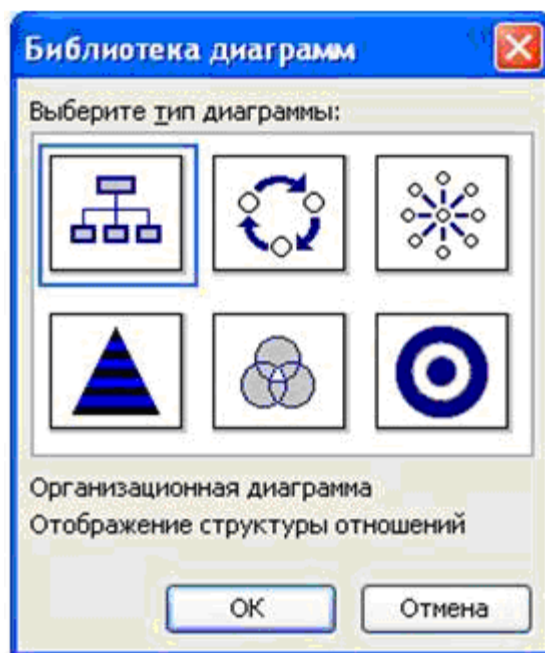
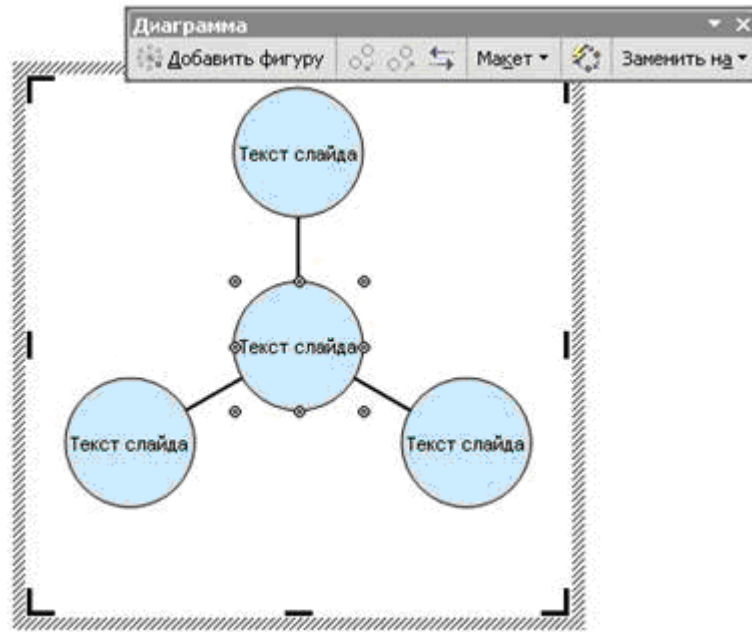
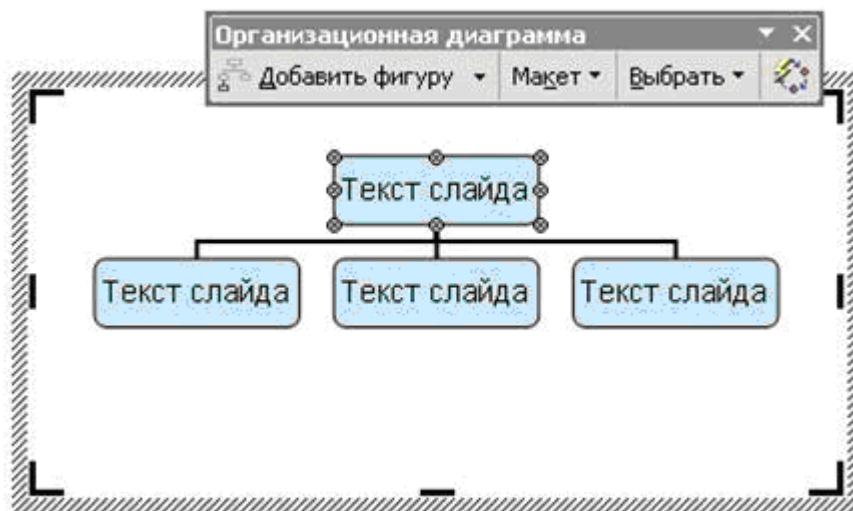


Рис. 3.71. Выбор организационной диаграммы



а) радиальная диаграмма



б) организационная диаграмма

Рис. 3.72. Создание диаграмм

Диаграмма формируется добавлением и/или удалением фигур, изменением их расположения, а также добавлением текста к фигурам и полям (слайдам) диаграммы. Работать можно только с выделенной фигурой диаграммы. Для ее выделения достаточно один раз щелкнуть левой кнопкой мыши по рамке фигуры. Для удаления любой фигуры диаграммы достаточно ее выделить и нажать клавишу **DEL**. Для ввода текста в фигуру диаграммы необходимо щелкнуть по словам *Текст слайда* и ввести нужный текст. Органи-

зационная диаграмма состоит из фигур, подразделяющихся на четыре группы:

- 1) *руководитель* – фигура, расположенная выше любой другой фигуры, такой, как сотрудник (подчиненный или коллега) или помощник, и соединенная с ней;
- 2) *помощник* – фигура, расположенная ниже какой-либо другой фигуры и соединенная с ней уступом (фигура помощника располагается выше всех фигур, являющихся подчиненными для соединенной с ней фигуры);
- 3) *подчиненные* – фигуры организационной диаграммы, расположенные ниже фигуры руководителя и связанные с ней (помощник и сотрудник по отношению к руководителю);
- 4) *сотрудники (коллеги)* – фигуры, расположенные рядом с другими и соединенные с одной и той же фигурой руководителя (также являющиеся подчиненными по отношению к руководителю).

Организационную диаграмму можно отформатировать целиком с помощью стандартных стилей или отформатировать отдельные ее части подобно форматированию фигур – добавлением цвета и текста, изменением толщины и типа линий, а также добавлением заливок, рисунков заливок и фона.

Чтобы во время презентации по мере изложения материала обращать внимание зрителей на определенные области диаграммы, можно включить анимацию.

3.5.10. Анимация объектов презентации

Анимация – добавление к тексту или объекту специального видео- или звукового эффекта. Например, можно создать элементы текстового списка, влетающие на страницу слева по одному слову, или добавить звук аплодисментов при открытии рисунка. Анимация позволяет «оживить» представление текста, графики, диаграмм и других объектов на слайдах, подчеркивает различные аспекты содержания, управляет ходом изложения материалов и делает презентацию более интересной.

Для упрощения разработки анимации следует пользоваться готовыми схемами (эффектами) анимации для элементов на всех или только выбранных слайдах, а также для определенных элементов на образце слайдов.

Выполнить настройку анимации можно с помощью области задач *Настройка анимации*, т. е. задать, где и в какой момент элемент должен появляться на слайде во время презентации, например, вылетать из-за левой границы по щелчку мыши.

Эффекты анимации могут применяться к элементам на слайде, находящимся в рамках, а также к абзацам, содержащим одиночные маркеры или пункты списков. Например, можно применить определенный эффект анимации ко всем элементам на слайде или только к отдельному абзацу с маркированным списком.

Кроме стандартных и специально заданных путей перемещения, можно добавить эффекты вхождения, выделения или выхода. Для одного элемента можно применить одновременно несколько эффектов анимации – например, сначала к маркеру списка применяется эффект вхождения, а затем – выхода.

Большинство параметров анимации включает ряд соответствующих эффектов. Это относится к средствам звукового сопровождения анимаций. Эффекты анимации текста, как правило, можно применить к буквам, словам и абзацам. Например, заголовок может появляться по отдельным словам, а не весь сразу. Эффекты анимации для текста или объектов можно просмотреть как для отдельного слайда, так и для всей презентации. Для использования готовой схемы анимации следует:

- если схему анимации требуется добавить только на определенные слайды, нужно эти слайды выделить;
- выбрать команду меню *Показ слайдов / Эффекты анимации...*;
- в области задач *Дизайн слайда* выбрать из списка *Применить к выделенным слайдам* анимационный эффект;
- если схему анимации требуется применить ко всем слайдам, достаточно нажать кнопку *Применить ко всем слайдам*.

Для использования специальной анимации нужно:

- в обычном режиме открыть слайд, к тексту или объектам которого требуется применить анимацию;
- выбрать объект анимации;
- выбрать команду меню *Показ слайдов / Настройка анимации*;

- в области задач *Настройка анимации* нажать кнопку *Добавить эффект* (рис. 3.73) и выполнить одно или несколько следующих действий:
- если во время показа слайдов требуется ввести текст или объект в сопровождении определенного визуального эффекта, следует выбрать команду *Вход*, а затем – нужный эффект;
- если требуется добавить определенный визуальный эффект в текст или объект, находящиеся на самом слайде, нужно выбрать команду *Выделение*, а затем – нужный эффект;
- если требуется добавить определенный визуальный эффект в текст или объект, который вызывает удаление текста или объекта со слайда в заданный момент, то следует использовать команду *Выход*, а затем – нужный эффект;
- если требуется добавить эффект, заставляющий объект перемещаться в пределах слайда по определенному маршруту, то нужно выбирать команду *Пути перемещения*, а затем – нужный эффект.

Эффекты отображаются в списке настройки анимации сверху вниз в порядке их применения (рис. 3.73). Изменить порядок следования эффектов можно с помощью кнопок *Порядок*. Объекты с эффектами анимации отмечаются на слайдах непечатаемыми

пронумерованными маркерами, соответствующими эффектам в списке. Эти маркеры не отображаются в режиме показа слайдов.

К параметрам эффектов анимации относятся разнообразные настройки, уточняющие выбранный эффект и меняющиеся в зависимости от используемого типа анимации. Например, могут

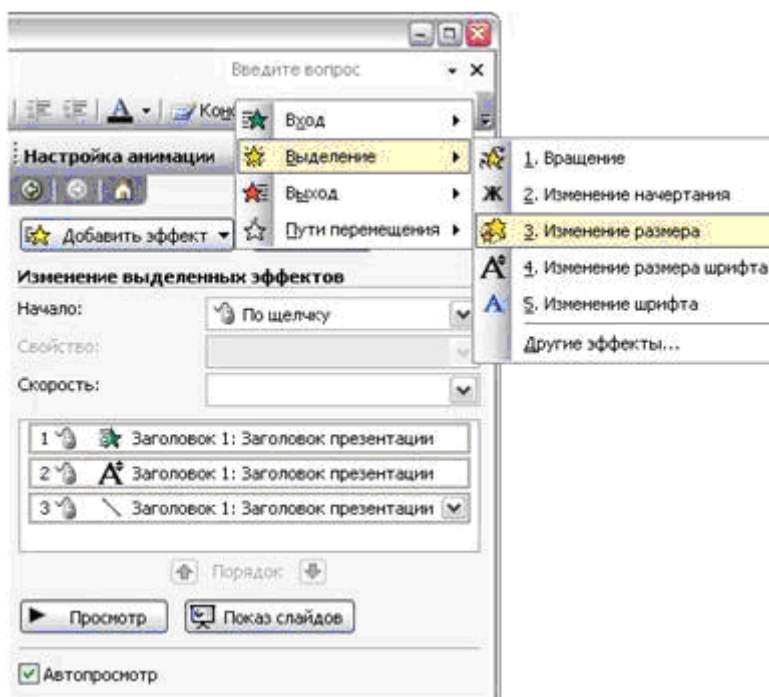


Рис. 3.73. Анимация объектов в MS PowerPoint

использоваться следующие настройки (для эффекта, показанного на рис. 3.73):

- 1) начало анимации – определяет момент времени или событие, после которого выбранный эффект анимации начинает выполняться (по щелчку мыши, следом за предыдущим или вместе с предыдущим);
- 2) направление анимации – по горизонтали или по вертикали;
- 3) скорость – очень медленно, медленно, средне, быстро или очень быстро.

3.5.11. Показ презентаций

Запуск показа слайдов можно выполнить несколькими способами. Например, можно открыть презентацию в MS PowerPoint и выбрать одно из следующих действий:

- нажать кнопку *Показ слайдов (с текущего слайда)* в левом нижнем углу окна MS PowerPoint;
- выбрать команду меню *Показ слайдов / Начать показ*;
- нажать клавишу **F5**.

Помимо этого можно начать показ слайдов, не запуская непосредственно MS PowerPoint. Для этого следует в проводнике выбрать файл презентации, который требуется показать, кликнуть на его имени правой кнопкой мыши и в контекстном меню выбрать команду *Демонстрация*.

3.5.11.1. Типы презентаций

3.5.11.1.1 Презентации на экране

Используя все возможности оформления MS PowerPoint, экранную версию презентации можно сделать законченной и интересной. Такие возможности, как смена слайдов, время показа слайда, фильмы, звуки, анимация, гиперссылки и смарт-теги, помогут разнообразить ее. После того, как для создания презентации вы решите использовать компьютер, вам откроется ряд способов ее представления.

Первый из них – презентация с докладчиком. Представление презентации в зале на мониторе или с помощью проекционного оборудования является наиболее распространенным способом. Докладчик имеет полный контроль над презентацией. Он может

проводить ее вручную или в автоматическом режиме и даже записывать во время презентации речевое сопровождение.

Второй способ – автоматическая презентация. Может потребоваться подготовить презентацию для автоматического показа на стенде компании во время выставки-ярмарки или конференции. Большинство элементов управления можно сделать недоступными с целью защиты от возможного изменения презентации пользователями. После завершения автоматическая презентация запускается повторно; то же самое происходит при смене слайдов вручную, когда какой-либо слайд отображается более пяти минут.

3.5.11.1.2. Интерактивные презентации

Проведение совместных конференций является одним из возможных вариантов использования MS PowerPoint. Использование MS NetMeeting совместно с MS PowerPoint позволяет совместно работать с презентациями и обмениваться данными в реальном времени между людьми, находящимися в разных городах, так, как если бы они находились в одной комнате.

В ходе конференции NetMeeting можно совместно использовать программы и документы, участвовать в разговоре, обмениваясь текстовыми сообщениями, передавать файлы. При совместной работе участники могут просматривать и изменять презентацию. Во время собрания по сети, если режим совместной работы отключен, в каждый момент времени только один человек может изменять презентацию, но в разговоре и на доске могут одновременно общаться несколько пользователей.

Презентацию можно передавать по каналам вещания, включая видео- и аудиоканалы, в сети Интернет. Каналы вещания можно использовать для собраний организации, презентаций для удаленных групп или собраний коллективов, сотрудники которых находятся на удалении друг от друга. С помощью MS Outlook или другой почтовой программы вещания презентации можно назначать так же, как и любое другое собрание. Презентация сохраняется в формате Hypertext Markup Language (HTML), что обеспечивает аудиторию всеми необходимыми средствами для ее просмотра с помощью браузера Microsoft Internet Explorer. Передача может быть записана и сохранена на веб-сервере в месте, доступном для ее воспроизведения в любое время.

Презентацию можно специально разработать для сети Интернет или интрасети, опубликовав ее в формате веб-страницы. Публикация презентации – это размещение ее копии в формате HTML в интернете. Несколько копий одной и той же презентации можно опубликовать в разных местах. Можно опубликовать всю презентацию, произвольный показ (содержит группу слайдов общей презентации, которые планируется показать определенной аудитории), отдельный слайд или ряд слайдов. Так как важным моментом при этом является наличие средств перемещения в презентации, в презентацию MS PowerPoint в формате HTML включается панель ссылок, позволяющая перемещаться по слайдам с помощью области структуры. В презентации, опубликованной в интернете, всем пользователям доступны заметки докладчика, которые можно использовать в качестве титров для доклада.

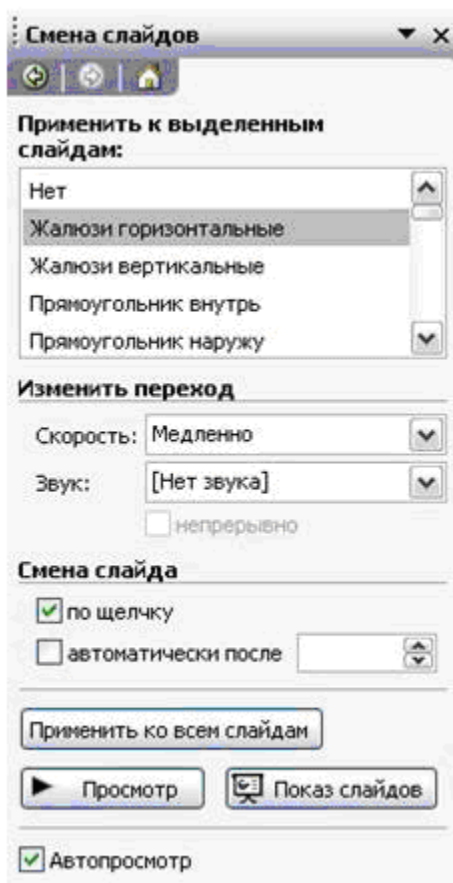


Рис. 3.74. Область задач Смена слайдов

3.5.11.1.3. Смена слайдов презентации

Смену слайдов презентации, так же как и отображение объектов оформления слайдов, можно «оживить» с помощью специальных средств MS PowerPoint: добавление переходов между слайдами и установка интервалов времени показа слайдов.

Добавить переход между слайдами и определить такие его параметры, как тип, временные интервалы, можно с помощью области задач MS Office *Смена слайдов*, изображенной на рис. 3.74.

Область *Применить к выделенным слайдам* позволяет указать нужный эффект переключения, после чего можно настроить его параметры.

В представленном на рис. 3.74 примере выбран эффект смены кадров *Жалюзи горизонтальные*. Для этого эффекта доступны такие параметры, как скорость испол-

нения, наличие звукового сопровождения (если флажок *Непрерывно* установлен, то выбранный звук повторяется по циклу).

Выбранный эффект смены слайдов можно применить ко всей презентации с помощью кнопки *Применить ко всем слайдам* (в этом случае все слайды будут меняться с использованием одного эффекта) и посмотреть этот эффект в действии, нажав на кнопку *Просмотр*.

В зависимости от типа экранной презентации и ее назначения в MS PowerPoint имеется возможность использовать разные способы смены слайдов. Так, например, если докладчик самостоятельно переключает слайды, то существуют определенные команды перемещения по ним. Для каждого вида перемещений можно выбрать один из нескольких способов.

1. Для перехода к следующему слайду можно:
 - а) нажать левую кнопку мыши;
 - б) нажать клавишу **Пробел** или **Enter**;
 - в) нажать правую кнопку мыши и выбрать в контекстном меню команду *Далее*.
2. Для перехода к предыдущему слайду можно:
 - а) нажать клавишу **BackSpace**;
 - б) нажать правую кнопку мыши и выбрать в контекстном меню команду *Назад*.
3. Для перехода к конкретному слайду можно:
 - а) ввести номер слайда и нажать клавишу **Enter**;
 - б) нажать правую кнопку мыши, выбрать в контекстном меню команду *Переход*, а затем указать номер.

В случае, когда презентация показывается в автоматическом режиме, смена кадров осуществляется автоматически по факту исполнения какого-либо эффекта слайда или по истечении заданного интервала времени. Рассмотрим способы определения автоматической смены кадров MS PowerPoint.

Для определения интервала смены слайда используется флажок *автоматически после* (рис. 3.74). После его установки определяется интервал в минутах и секундах, по истечении которого осуществляется переход к следующему слайду.

Следует отметить одну особенность, связанную с настройкой презентации. Если разработчик задал тип смены кадров как *вручную* (в диалоговом окне *Настройка презентации*, доступном при выборе

команды меню *Показ слайдов / Настройка презентации...*), то смена кадров будет происходить по любому из следующих событий:

- истек заданный интервал времени;
- пользователь ввел команду смены слайда.

Если же установлен тип смены кадров *по времени*, то переход к следующему кадру будет осуществляться только по истечении заданного временного интервала. Соответственно, если в последнем случае интервал времени смены слайдов не задан, то показываться будет только один единственный слайд – тот, с которого начался показ презентации.

Завершить показ презентации, независимо от выбранного режима смены слайдов, можно нажатием клавиши **Esc**.

3.5.11.1.4. Показ слайдов на втором мониторе

Прежде чем включить поддержку использования двух мониторов (ее иногда называют поддержкой нескольких мониторов), необходимо убедиться в наличии необходимого оборудования, а затем включить один из режимов поддержки нескольких мониторов операционной системы. Поддержка нескольких мониторов доступна для настольных компьютеров с операционной системой Microsoft Windows 2000 SP 3 или более поздней версии, а также для переносных компьютеров с системой Microsoft Windows XP.

Показ слайдов на втором мониторе позволяет продемонстрировать слайды в полноэкранном режиме на втором мониторе при одновременном их просмотре в обычном режиме (способ отображения презентации и предоставления пользователю средств работы с ней) на основном мониторе. Для этого следует выбрать команду меню *Показ слайдов / Настройка презентации...* и в группе *Несколько мониторов* в списке *Выводить слайды на:* выбрать монитор, на котором требуется проводить показ слайдов.

3.5.12. Упаковка презентаций для переноса

Средство упаковки для компакт-диска Microsoft Office PowerPoint позволяет скопировать одну или несколько презентаций на компакт-диск вместе со вспомогательными файлами. При этом на компакт-диск по умолчанию записывается также средство просмотра для запуска презентаций на другом компьютере в том

случае, если на нем не установлена программа MS PowerPoint. При упаковке презентации необходимые файлы добавляются автоматически, однако предусмотрена возможность их исключения, а также включения других файлов в комплект упаковки.

Средство просмотра (MS PowerPoint Viewer) автоматически упаковывается вместе с презентацией, но его тоже можно исключить, если заранее известно, что на компьютере, предназначенном для запуска презентации, установлена программа MS PowerPoint.

При использовании шрифтов TrueType их также можно включить в упаковку, что гарантирует их доступность при показе презентации на другом компьютере. Однако мастер упаковки не может включать шрифты TrueType, имеющие встроенные ограничения на распространение.

Можно защитить содержимое компакт-диска, установив пароли для открытия и изменения всех упакованных презентаций. Если для открытия презентации был установлен пароль до ее упаковки, то он будет сохранен в упакованной презентации. Чтобы упакованная презентация была доступна без введения пароля, необходимо снять защиту паролем до упаковки.

3.5.13. Вывод слайдов презентации на печать

Всю презентацию (слайды, структуру, страницы заметок и раздаточные материалы) можно напечатать в цвете, оттенках серого или в черно-белом режиме без серого. Также можно напечатать указанные слайды, страницы заметок, выдачи и страницы структуры.

В общем, вывод на печать слайдов презентации похож на процедуру печати, рассмотренную в MS Word и MS Excel, однако имеется ряд специфических особенностей, которые рассмотрим далее.

3.5.13.1. Цветная печать, черно-белая печать и печать с оттенками серого

Большинство презентаций предназначено для показа в цвете, но слайды и выдачи обычно печатаются в черно-белом режиме. При запуске печати в MS PowerPoint цвета презентации изменяются в соответствии с возможностями выбранного принтера. Например,

если выбранный принтер является черно-белым, то презентация будет автоматически распечатана в оттенках серого.

Режим предварительного просмотра позволяет оценить, как будут выглядеть слайды, заметки и раздаточные материалы в черно-белом варианте или при печати с оттенками серого, и внести нужные изменения перед запуском печати. В этом режиме предварительного просмотра можно внести определенные изменения, например, выбрать следующие параметры:

- 1) определить печатаемые материалы: презентация, выдачи, страницы заметок или структура;
- 2) задать макет для печати выдач;
- 3) сформировать рамку вокруг каждого слайда (только при печати);
- 4) определить ориентацию листов бумаги (книжная или альбомная) для выдач, заметок или структуры;
- 5) установить колонтитулы.

3.5.13.2. Слайды и прозрачки

Можно распечатать только слайды и использовать их в качестве раздаточных материалов. Слайды печатаются по одному на листе, и их масштаб может быть выбран в соответствии с различными размерами бумаги. Также можно выбрать масштаб слайдов для печати на прозрачках в виде слайдов на 35-мил-лиметровой пленке или установить специальные размеры и ориентацию.

3.5.13.3. Структура страниц заметок и раздаточного материала

Для структуры можно распечатать весь ее текст или только заголовки слайдов как в альбомной (горизонтальной), так и в книжной (вертикальной) ориентации. Вид распечатанных материалов может отличаться от их вида на экране, так как при просмотре на экране можно скрыть или отобразить форматирование текста (такое, как полужирное или наклонное начертание) на панели *Структура*, а в распечатанном виде форматирование отображается всегда.

Распечатанные заметки можно использовать как для удобства проведения презентации, так и для использования в качестве раздаточного материала для слушателей. Заметки можно создать и

отформатировать, используя параметры цветов, фигур, диаграмм и разметки. Каждая заметка содержит копию слайда, на который она ссылается. Каждый слайд печатается на отдельной странице, а заметки печатаются под изображением соответствующего слайда. Чтобы печатать по два слайда с соответствующими заметками после них на одной странице, презентацию можно отправить в MS Word. Колонтитулы на страницах заметок не связаны с колонтитулами на слайдах.

Выдачи создаются и печатаются аналогично страницам заметок. Однако при печати доступно множество параметров макета – от 1 до 9 слайдов на листе. Макет с тремя слайдами на листе содержит свободное разлинованное поле для записи собственных заметок слушателями. Для использования дополнительных параметров макета презентацию можно отправить в MS Word.

Для печати слайдов рекомендуется выполнить следующие действия:

- 1) задать размеры слайда для печати (выбрать команду меню *Файл / Параметры страницы...*, указать размер слайдов, ориентацию и другие параметры);
- 2) проверить вид распечатываемых слайдов с помощью предварительного просмотра (команда меню *Файл / Предварительный просмотр*);
- 3) на панели инструментов *Предварительный просмотр* в поле *Печатать следующее:* выбрать вариант *Слайды*;
- 4) нажать кнопку *Печать...* панели инструментов *Предварительный просмотр*.

Закрыть режим предварительного просмотра можно с помощью кнопки *Закрыть*.

3.5.14. Технология создания и использования презентаций

1. Открыть приложение MS PowerPoint. Выполнить команду меню *Файл / Создать...* или щелкнуть мышью по кнопке *Создать*.
2. В окне *Создание презентации* выбрать один из способов создания:
 - при создании пустой презентации выбрать способ *Новая презентация*;

- при использовании готовых шаблонов оформления выбрать способ *Из шаблона оформления* и нужный шаблон.
3. Слайд 1. В элемент разметки *Заголовок слайда* вписать нужный текст.

Установку параметров оформления шрифта можно выполнить, выбрав меню *Формат / Шрифт...* или с помощью контекстного меню, нажав правую кнопку мыши и выбрав строку *Шрифт...*. Можно создать заголовок с помощью редактора WordArt, для этого необходимо удалить элемент разметки *Заголовок слайда*, установив на его границу курсор и нажав правую кнопку мыши, выбрать команду *Вырезать*. На место этого элемента вставить фигурный текст. С этой целью вывести панель инструментов *Рисование*, если ее нет, и на ней нажать кнопку *Добавить объект WordArt*. В появившемся диалоговом окне выбрать стиль надписи и нажать кнопку *ОК*. В следующем диалоговом окне *Изменение текста WordArt* внести текст нужной надписи, выбрать шрифт, его размер и нажать кнопку *ОК*. На панели инструментов *WordArt* нажать кнопку *Форма WordArt* и выбрать нужную форму.

Для создания фона слайда нужно на панели инструментов *Форматирование* или нажать кнопку *Фон*, или выбрать команду меню *Формат / Фон...*, или щелкнуть правой кнопкой мыши по полю слайда и выбрать *Фон...*. В диалоговом окне *Фон* открыть список *Заливка фона* и выбрать *Способы заливки*. В диалоговом окне *Способы заливки* выбрать один из желаемых вариантов: *Текстура*, *Узор* или *Рисунок*. После выбора текстуры, узора или рисунка в диалоговом окне *Способы заливки* нажать кнопку *ОК*. В диалоговом окне *Фон* нажать кнопку *Применить*.

Для наложения эффекта анимации на элемент разметки (слайд) необходимо выбрать пункт меню *Показ слайдов / Настройка анимации...*. В появившемся окне *Настройка анимации* установить флажки на тех объектах, на которые нужно наложить анимацию. На вкладке *Порядок и время в группе Анимация* включить переключатель *Автоматически через 00:01 секунд после предыдущего события*, на вкладке *Видоизменение для каждого объекта* выбрать эффект и звук в

соответствующих раскрывающихся списках. Эффекты анимации можно посмотреть, нажав кнопку *Просмотр*.

4. Внести нужный текст в элемент разметки *Подзаголовок слайда*. Если требуется использовать списки, то на панели инструментов *Форматирование* нажать кнопку *Маркеры* или нажать правую кнопку мыши и в контекстном меню выбрать *Список*. В диалоговом окне *Список* выбрать вариант *маркированный*, а затем соответствующий вид, цвет, рисунок (если нужно) и нажать кнопку *ОК*. Далее внести нужный текст с соответствующим необходимым оформлением, как и для элемента разметки *Заголовок слайда*.
5. Создать необходимое количество следующих слайдов. Для этого нужно выбрать пункт меню *Вставка / Создать слайд* или использовать кнопку *Создать слайд* на панели инструментов *Форматирование*. В диалоговом окне *Новый слайд* выбрать нужный автомакет (*Текст и графика*, или *Текст в две колонки*, или *Графика и текст*, или *Таблица*, или *Диаграмма*).

Для вставки картинки в элемент разметки *Вставка картинки* нужно выбрать команду меню *Вставка / Рисунок / Картинки...* или *Из файла...* или дважды щелкнуть мышью по элементу разметки. В диалоговом окне *Добавить рисунок* (из файла) выбрать нужный или в диалоговом окне *Вставка рисунка* нажать кнопку *Импорт клипов* и в открывшемся окне *Добавление клипов в коллекцию* отметить нужную картинку и нажать кнопку *Импорт*.

Для вставки диаграммы нужно выбрать меню *Вставка / Диаграмма...* или выполнить двойной щелчок мышью на элементе разметки *Диаграмма*. В диалоговом окне *Таблица данных* значения в таблице заменить нужными. Для подписей *оси значений* и названия *оси категорий* нажать правой кнопкой мыши на диаграмме, в контекстном меню выбрать *Параметры диаграммы* и заполнить нужные вкладки.

Для вставки таблицы активизировать в элементе разметки *Вставка таблицы* этот элемент и выполнить команду меню *Вставка / Таблица...*. В диалоговом окне *Вставка таблицы* выбрать нужное число столбцов и строк в создаваемой таблице. Заданная таблица автоматически вставляется, после чего в

ее ячейки необходимо внести требуемые значения. Создать таблицу можно также, выполнив команду меню *Вставка / Объект...*. В окне *Тип объекта* выбрать *Лист MS Excel*. В появившуюся таблицу можно вносить свои данные. Особенность этой таблицы в том, что, активизировав ее, можно изменять ширину и высоту ячеек, а также расположение текста. Для этого, встав на нужную ячейку и нажав правую кнопку мыши, в контекстном меню выбрать *Формат ячейки*. При щелчке по полю слайда таблица перестает быть активной.

6. На все созданные слайды наложить эффекты анимации и отформатировать текст.

7. Настроить просмотр презентации (существует несколько вариантов). Рассмотрим один из них. Выполнить команду меню *Показ слайдов / Настройка презентации...*. В диалоговом окне *Настройка презентации* выбрать переключатель *Автоматически*. Отметить слайды, входящие в презентацию. Установить смену кадров *по времени* и нажать кнопку *ОК*.

Для внесения заметок к слайду установить курсор на поле для заметок слайда и активизировать его.

Для установки времени просмотра слайда выполнить команду меню *Показ слайдов / Смена слайдов...*. В окне *Смена слайдов* выбрать нужные свойства и нажать кнопку *Применить*. Установить время перехода от одного слайда к другому можно, просмотрев слайд в режиме репетиции. Для этого команду меню *Показ слайдов / Настройка времени*. Появится панель *Репетиция*, на которой есть кнопка *Далее* (ее нужно нажать после просмотра слайда для перехода к следующему) и кнопка *Пауза* для приостановки репетиции. Для перехода от одного слайда к другому можно нажать кнопку *Следующий слайд* или *Предыдущий слайд* в вертикальной полосе прокрутки.

8. Сохранить созданную презентацию в файле с соответствующим именем. Эту операцию можно выполнить в начале создания презентации.

3.5.15. Контрольные вопросы

- 1) Как называется отдельная страница презентации?
- 2) В каком режиме выполняется корректировка логической последовательности слайдов?

- 3) Как называется область, указываемая по умолчанию на образце слайдов?
- 4) Что называется презентацией?
- 5) Из каких элементов состоит макет слайда?
- 6) Каким образом можно завершить показ слайдов?
- 7) Каким образом осуществляется демонстрация презентации?
- 8) Из каких элементов состоит презентация?
- 9) Какие из перечисленных ниже групп содержит область *Разметка слайда*: а) макеты текста; б) макеты содержимого; в) макеты графики; г) макеты формул; д) макеты текста и содержимого?
- 10) Как называется файл, созданный средствами MS PowerPoint?
- 11) Выполняется ли проверка орфографии в MS PowerPoint?
- 12) Возможно ли создание слайда на основе шаблонов?

3.6. Средства создания и обработки компьютерной графики

3.6.1. Виды компьютерной графики

Существует специальная область информатики – *компьютерная графика*, в рамках которой изучаются методы и средства создания и обработки изображений с помощью программно-аппаратных вычислительных комплексов. В зависимости от способа формирования изображений компьютерная графика подразделяется на *растровую, векторную, фрактальную и трехмерную (3D) графику*.

Для *растровых изображений*, состоящих из точек, особую важность имеет понятие *разрешения*, задающее количество точек, приходящихся на единицу длины. Принято различать:

- разрешение оригинала;
- разрешение экранного изображения;
- разрешение печатного изображения.

Разрешение оригинала измеряется в точках на дюйм и зависит от требований к качеству изображения и размеру файла, способу оцифровки или методу создания исходной иллюстрации, избранному формату файла и другим параметрам. Чем выше требования к качеству, тем выше должно быть разрешение оригинала.

Разрешение экранного изображения измеряется в элементарных точках раstra, называемых *пикселями*. Размер пикселя варьируется в зависимости от выбранного экранного разрешения

(из диапазона стандартных значений), разрешения оригинала и масштаба отображения.

Разрешение печатного изображения зависит от примененного метода и параметров растривания оригинала. При растривании на оригинал как бы накладывается сетка линий, ячейки которой образуют *элемент растра*. Частота сетки растра измеряется числом линий на дюйм и называется *линиатурой*.

Для *векторной графики* базовым элементом изображения является *линия*. *Линия* – элементарный объект векторной графики. Как и любой объект, линия обладает свойствами: формой (прямая, кривая), толщиной, цветом, начертанием (сплошная, пунктирная).

Объекты векторной графики имеют следующие способы представления.

- *Точка* на плоскости представляется двумя числами (координатами x , y), указывающими ее положение относительно начала координат.
- *Прямая линия* описывается уравнением $y = kx + b$. Указав параметры k и b , всегда можно отобразить бесконечную прямую линию в системе координат, то есть для задания прямой достаточно двух параметров.
- *Отрезок прямой* отличается тем, что для его описания требуется задать координаты начала и конца.
- *Кривая второго порядка* (парабола, гипербола, эллипс, окружность) описывается уравнениями, которые содержат степени не выше второй, и не имеет точек перегиба. Для описания бесконечной кривой второго порядка достаточно пяти параметров, а для части кривой еще два параметра (координаты конечных точек). Формула кривой второго порядка в общем случае может выглядеть, например, так:

$$x^2 + a_1 \cdot y^2 + a_2 \cdot x \cdot y + a_3 \cdot x + a_4 \cdot y + a_5 = 0.$$

- *Кривая третьего порядка* отличается от кривой второго порядка возможным наличием точки перегиба. Для ее описания потребуется девять параметров, а для описания ее части потребуется еще два параметра. В общем случае уравнение кривой третьего порядка может выглядеть, например, так:

$$x^3 + a_1 \cdot y^3 + a_2 \cdot x^2 \cdot y + a_3 \cdot x \cdot y^2 + a_4 \cdot x^2 + a_5 \cdot y^2 + a_6 \cdot x \cdot y + a_7 \cdot x + a_8 \cdot y + a_9 = 0$$

Фрактальная графика, как и векторная, основана на математических вычислениях. Однако ее базовым элементом является сама

математическая формула, то есть никаких объектов в памяти компьютера не хранится и изображение строится исключительно по уравнениям.

Трехмерная графика сочетает, как правило, векторный и растровый способы формирования изображений. Трехмерная графика изучает приемы и методы построения объемных моделей объектов в виртуальном пространстве. Она нашла широкое применение в областях научных расчетов, инженерного проектирования, компьютерного моделирования физических объектов.

3.6.2. Средства для работы с компьютерной графикой

3.6.2.1. Средства для работы с растровой графикой

Среди программных средств, предназначенных для создания компьютерных двумерных изображений, самыми популярными считаются Painter, FreeHand, Adobe PhotoShop.

Пакет Painter обладает достаточно широким спектром средств рисования и работы с цветом. Он моделирует различные инструменты: кисти, карандаш, перо, уголь, аэрограф и другие. Позволяет имитировать материалы: акварель, масло, тушь, а также добиться эффекта натуральной среды.

Пакет FreeHand обладает богатыми средствами редактирования изображений и текста, содержит библиотеку спецэффектов и набор инструментов для работы с цветом.

Пакет Adobe PhotoShop является стандартом в компьютерной графике, и все другие программы неизменно сравнивают с ним.

К аппаратным средствам получения цифровых растровых оригиналов в основном относятся *сканеры* и *цифровые фотоаппараты*. Для создания изображений вручную предназначены *графические планшеты*, на которых рисуют специальным электронным пером.

Сканеры по способу восприятия изображения делятся на две группы.

- Устройства с электронными фотоумножителями (ФЭУ). Эти устройства называют барабанными – внутри аппарата помещен прозрачный барабан, на который крепится оригинал. Конструктивно эти сканеры выполняют с вертикальным или горизонтальным барабаном, съемным или несъемным.

- Устройства на приборах с зарядовой связью (ПЗС). К этим устройствам относятся все прочие сканеры. Приборы с зарядовой связью представляют собой фотоприемник, выполненный на кремниевых элементах, объединенных в линейку. Сканеры на ПЗС бывают листовые, планшетные, проекционные, ручные и так называемые слайдовые (для сканирования оригиналов на просвет).

Цифровые фотокамеры основаны на матрице ПЗС, состоящей из двумерного массива элементов. Для целей электронной публикации и непрофессионального применения достаточное число элементов на матрице около 1,5 миллиона. Полупрофессиональные камеры должны иметь разрешение матрицы не ниже 2 миллионов элементов. Профессиональные аппараты – 2,5÷3 миллиона.

Графические планшеты представляют собой координатную двумерную электронную сетку, каждый элемент которой способен воспринимать и передавать ряд сигналов от электронного пера. К таким сигналам относятся:

- координаты точки контакта пера с планшетом;
- сила нажима;
- угол наклона;
- скорость прохода (время экспозиции) и ряд других.

Обладая достаточным навыком работы с графическим планшетом, удается достаточно точно имитировать различную живописную технику – письмо маслом, рисунок углем, аэрографом, карандашом и другие.

3.6.2.2. Средства создания и обработки векторной графики

К программным средствам создания и обработки векторной графики относятся графические редакторы (Adobe Illustrator, CorelDraw, Macromedia Freehand) и векторизаторы (трассировщики) – специальные пакеты преобразования растровых изображений в векторные (Adobe Streamline, CorelTrace).

Векторный редактор Adobe Illustrator является одним из общепризнанных лидеров среди программ этого класса. Его особое преимущество заключается в отлаженном взаимодействии с другими продуктами компании Adobe (PhotoShop и PageMaker).

Векторный редактор CorelDraw считается основным, особенно в России, пакетом создания и обработки векторной графики на платформе MS Windows. К его преимуществам относятся развитая

система управления и обширные средства настройки параметров инструментов. Однако интерфейс программы сложен для освоения.

Векторный редактор Macromedia Freehand с простым и дружелюбным интерфейсом служит удобным инструментом работы для начинающих. Программа отличается небольшим размером и хорошим быстродействием. Нетребовательность к аппаратным ресурсам позволяет работать на компьютерах среднего уровня. Инструментальные средства этого редактора достаточны для создания сложных документов, и он лишь в некоторых элементах уступает более мощным средствам Adobe Illustrator и CorelDraw.

Трассировщик Adobe Streamline занимает ведущее место в своем классе программ. Он позволяет проводить тонкую настройку параметров векторизации, что улучшает его точность. Более всего векторизация удобна для преобразования чертежей, черно-белых рисунков и другой простой графики без полутонов.

3.6.2.3. Программные средства обработки трехмерной графики

Программа создания и обработки трехмерной графики 3D Studio Max предназначена для работы в среде MS Windows. Этот пакет считается полупрофессиональным. Однако его средств вполне хватает для разработки качественных изображений объектов неживой природы. Отличительными особенностями пакета являются поддержка большого числа аппаратных ускорителей трехмерной графики, мощные световые эффекты, большое число дополнений, созданных сторонними фирмами. Сравнительная нетребовательность к аппаратным ресурсам позволяет работать даже на компьютерах среднего уровня. Вместе с тем по средствам моделирования и анимации этот пакет уступает более развитым программным средствам.

Программа MS Softimage 3D также приспособлена для работы в среде MS Windows. Программу отличают богатые возможности моделирования, наличие большого числа регулируемых физических и кинематографических параметров. Эта программа считается стандартом в мире специализированных графических станций SGI. Однако при применении на компьютерах среднего уровня выглядит несколько тяжеловато и требует мощных аппаратных ресурсов.

Программа Maya обладает большими возможностями интерфейса. Пакет существует в вариантах для разных операционных

систем, в том числе и для MS Windows. Он имеет модульное построение и включает следующие блоки:

- *Base* – содержит ядро программы. Обеспечивает поддержку основных инструментов моделирования, инверсной кинематики, обработки звука, имитации физических твердых тел, захвата движения и основных наборов эффектов.
- *Maya F/X* – набор дополнительных модулей.
- *Maya Power Modeler* – в основном содержит мощные средства полигонального и сплайнового моделирования объектов.
- *Maya Artisan* – наиболее передовой модуль, позволяющий обрабатывать виртуальные модели методами, характерными для реальной работы скульпторов и художников.
- *Maya Cloth* – предназначен для моделирования одежды.
- *Maya Fur* – модуль для имитации поверхностей, покрытых шерстью или мехом.
- *Maya Live* – сценарный модуль, обеспечивающий сопряжение реальных съемок с компьютерной анимацией.

Инструментарий *Maya* сведен в четыре группы:

- *Animation* (анимация);
- *Modeling* (моделирование);
- *Dynamic* (физическое моделирование);
- *Rendering* (визуализация).

Удобный настраиваемый интерфейс выполнен в соответствии с современными требованиями.

3.6.3. Представление графических данных

Для хранения изображений в компьютерной графике применяют, по меньшей мере, три десятка форматов файлов. Однако лишь часть из них стала стандартом и применяется в большинстве программ. Поэтому рассмотрим следующие форматы.

- TIF (Tagged Image File Format) предназначен для хранения растровых изображений высокого качества (расширение имени файла .TIF). Относится к числу широко распространенных, обеспечен поддержкой со стороны большинства графических, верстальных и дизайнерских программ.
- PSD (PhotoShop Document) является собственным форматом программы Adobe PhotoShop (расширение имени файла .PSD), по

возможностям хранения растровой информации – один из наиболее мощных. Поддерживаются 48-разрядное кодирование цвета, цветоделение и различные цветовые модели. Основной недостаток – отсутствие эффективного алгоритма сжатия информации, что приводит к большому объему файлов.

- BMP (*Windows Bitmap*) предназначен для хранения растровых изображений в операционной системе MS Windows (расширение имени файла .BMP) и поддерживается всеми приложениями, работающими в этой среде.
- PDF (*Portable Document Format*) в основном предназначен для хранения документов целиком, однако его впечатляющие возможности позволяют обеспечить эффективное представление изображений (расширение имени файла .PDF). Формат является аппаратно-независимым, поэтому вывод изображений допустим на любые устройства – от экрана монитора до фотоэкспонирующего устройства. Мощный алгоритм сжатия со средствами управления итоговым разрешением изображения обеспечивает компактность файлов при высоком качестве иллюстраций.

3.6.4. Контрольные вопросы

- 1) Назовите основные типы графической информации в компьютере.
- 2) Какие виды компьютерной графики вам известны?
- 3) Что считается стандартом для создания двумерных изображений в компьютерной графике?
- 4) Какие программные средства для работы с растровой графикой вам известны?
- 5) Перечислите средства создания и обработки векторной графики?
- 6) Какие вам известны программные средства обработки трехмерной графики?
- 7) Установите соответствие расширений имен файлов их форматам:

| | |
|--------|------------------------------|
| а) TIF | а) PowerPoint |
| б) PSD | б) Windows Bitmap |
| в) BMP | в) Tagged Image File Format |
| г) PDF | г) PhotoShop Document |
| | д) Portable Document Format. |
- 8) Какая из нижеприведенных заливок называется градиентной:
 - а) с переходом (от одного цвета к другому);
 - б) с использованием внешней текстуры;

- в) с узором;
 - г) сплошная (одним цветом)?
- 9) Какие проекции используются в компьютерной графике?
- 10) Какие графические форматы известны вам?
- 11) Что называется пикселем:
- а) размер шрифта;
 - б) объект векторной графики;
 - в) объем памяти, необходимый для хранения одной точки изображения;
 - г) минимальный элемент растрового изображения?
- 12) Как называется графический формат, позволяющий при сохранении фотографий получить наименьший объем?

3.7. Базы данных

База данных – это организованная структура для хранения информации. Такое понятие базы данных обусловлено тем, что современные системы управления базами данных (СУБД) позволяют размещать в своих структурах не только данные, но и методы (то есть программный код), с помощью которых происходит взаимодействие с потребителем или с другими программно-аппаратными комплексами. Таким образом, в современных базах данных хранятся не только данные, но и информация. Если в базе данных нет никаких данных, то это все равно полноценная база. В такой базе данных все-таки есть информация – это структура базы. Она определяет методы занесения и хранения данных в базе.

С понятием базы данных тесно связано понятие системы управления базой данных. *Система управления базой данных (СУБД)* – это комплекс программных средств, предназначенных для создания структуры новой базы, наполнения ее содержимым, редактирования содержимого и визуализации информации. Под визуализацией информации базы понимается отбор отображаемых данных в соответствии с заданным критерием, их упорядочение, оформление и последующая выдача на устройство вывода или передача по каналам связи.

В мире существует множество СУБД. Однако большинство из них опирается на единый устоявшийся комплекс основных понятий. Поэтому рассмотрим и обобщим понятия только одной системы управления базами данных – MS Access, входящей в состав пакета

Microsoft Office. Базы данных могут содержать различные объекты, но основными объектами являются ее таблицы. Структуру двумерной таблицы образуют столбцы и строки. Их аналогами в структуре базы данных являются соответственно *поля и записи*. Поля базы определяют групповые свойства данных, записываемых в ячейки, принадлежащие каждому из полей. Изменение состава полей исходной таблицы (или их свойств) приводит к изменению структуры базы данных, то есть создается новая база данных.

Любая система (структура) базы данных является моделью организации хранения данных (информации). Существуют следующие виды типовых моделей баз данных: реляционная (табличная), иерархическая, сетевая. В настоящее время наиболее распространены реляционные модели.

3.7.1. Основные свойства полей таблиц

В СУБД MS Access реализуются следующие основные свойства полей в структурах таблиц базы данных.

- *Имя поля* (заголовок столбца таблицы) – определяет порядок обращения с данными этого поля при автоматических операциях с базой.
- *Тип поля* – определяет тип данных, которые могут содержаться в данном поле.
- *Размер поля* – определяет предельную длину (в символах) данных, которые могут размещаться в данном поле.
- *Формат поля* – определяет способ форматирования данных в ячейках, принадлежащих полю.
- *Маска ввода* (средство автоматизации ввода) – определяет форму, в которой вводятся данные в поле.
- *Подпись* – определяет заголовок столбца таблицы для данного поля, если подпись не указана, то в качестве заголовка столбца используется свойство *Имя поля*.
- *Значение по умолчанию* – то значение, которое вводится в ячейки поля автоматически (средство автоматизации ввода данных).
- *Условие на значение* – ограничение, используемое для проверки правильности ввода данных (средство автоматизации

ввода, которое используется, как правило, для данных, имеющих числовой тип, денежный или тип даты).

- *Сообщение об ошибке* – текстовое сообщение, которое выдается автоматически при попытке ввода в поле ошибочных данных (проверка ошибочности выполняется автоматически, если задано свойство *Условие на значение*).
- *Обязательное поле* – свойство, определяющее обязательность заполнения данного поля при наполнении базы.
- *Пустые строки* – свойство, разрешающее ввод пустых строчковых данных (от свойства *Обязательное поле* отличается тем, что относится не ко всем типам данных, а лишь к некоторым, например, текстовым).
- *Индексированное поле* – если поле обладает этим свойством, все операции, связанные с поиском или сортировкой записей по значению, хранящемуся в данном поле, существенно ускоряются.

Примечание: перечень приведенных свойств полей относится в основном к полям текстового типа.

Поскольку в разных полях могут содержаться данные разного типа, то и свойства у полей могут различаться в зависимости от типа данных.

3.7.2. Типы данных

Система управления базами данных MS Access поддерживает следующие типы данных.

- *Текстовый* – используется для хранения обычного неформатированного текста ограниченного размера (до 255 символов).
- *Поле Мемо* – специальный тип данных для хранения больших объемов текста (до 65535 символов).
- *Числовой* – тип данных для хранения действительных чисел.
- *Дата/время* – тип данных для хранения дат и текущего времени.
- *Денежный* – тип данных для хранения денежных сумм.
- *Счетчик* – специальный тип данных для натуральных уникальных чисел с автоматическим наращиванием (для порядковой нумерации записей).

- *Логический* – тип для хранения логических данных (только два значения *Да* или *Нет*).
- *Поле объекта OLE* – специальный тип данных для хранения объектов OLE, например мультимедийных.
- *Гиперссылка* – специальное поле для хранения адресов URL веб-объектов интернета.
- *Мастер подстановок* – это не тип, а объект, настройкой которого можно автоматизировать ввод данных в поле так, чтобы не вводить их вручную, а выбирать из раскрывающегося списка.

3.7.3. Основные типы объектов

СУБД MS Access позволяет создавать объекты шести основных типов.

- 1) *Таблицы* – это определяющие (основные) объекты любой базы данных. В таблицах хранятся все данные, имеющиеся в базе, и структура базы (поля, их типы и свойства).
- 2) *Запросы* – это объекты, предназначенные для извлечения данных из таблиц и предоставления их пользователям в удобном виде. С помощью запросов можно выполнять отбор данных; их сортировку и фильтрацию; преобразовывать данные по заданному алгоритму; создавать новые таблицы; выполнять автоматическое заполнение таблиц данными, импортированными из других источников; выполнять простейшие вычисления в таблицах и другое.
- 3) *Формы* – это средства для ввода и отображения данных.
- 4) *Отчеты* – это средства для вывода данных на печатающее устройство (принтер).
- 5) *Страницы* – это специальные объекты баз данных (страницы доступа к данным). Это объекты, выполненные в коде HTML, размещаемый на веб-странице и передаваемые клиенту вместе с ней.
- 6) *Макросы и модули* – это категории объектов, предназначенные для автоматизации повторяющихся операций при работе с СУБД и для создания новых функций путем программирования. Макросы состоят из последовательности внутренних команд СУБД и являются средством автоматизации работы с базой. Модули создаются средствами

внешнего языка программирования Visual Basic for Applications (VBA).

СУБД MS Access предоставляет следующие средства создания основных объектов базы:

- ручные средства разработки объектов в режиме *Конструктора*;
- автоматизированные средства разработки объектов с помощью программ-мастеров;
- автоматические средства ускоренной разработки простейших объектов.

Примечание: при изучении и овладении СУБД MS Access рекомендуется начинать работу в режиме *Конструктора* для разработки учебных таблиц и запросов. При разработке учебных форм, отчетов и страниц доступа рекомендуется использовать автоматизированные средства, предоставляемые мастерами.

3.7.4. Работа с СУБД MS Access

Технологии по созданию базовых таблиц, межтабличных связей и запросов рассмотрим на конкретном примере.

Постановка задачи. Создать базу данных, содержащую информацию о товарах, имеющихся на складе, о клиентах, выполняющих заказы и покупки. Предусмотреть возможность выборки информации по различным критериям, создать форму для ввода информации и отчеты для подготовки печатных документов.

3.7.4.1. Создание таблиц

- 1) Запустить программу MS Access (двойной щелчок мышью на рабочем столе по ярлычку *Microsoft Access* или меню *Пуск*, в нем *Программы (Все программы)*, затем строка *Microsoft Office* и в появившемся меню щелкнуть по строке *Microsoft Access*).
- 2) В окне Microsoft Access включить переключатель *Новая база данных* и щелкнуть по кнопке *Ok*.
- 3) В окне *Файл новой базы данных* выбрать папку *Мои документы* и ввести имя файла *Торговая организация*. Убедиться, что в качестве типа файла выбрано *Базы данных Microsoft Access*, и щелкнуть по кнопке *Создать*. Откроется окно новой базы – *Торговая организация: база данных*.

- 4) Открыть панель *Таблицы*.
- 5) Открыть бланк создания структуры таблицы двойным щелчком мыши по ярлычку *Создание таблицы в режиме конструктора*.
- 6) Для первой таблицы ввести следующие поля:

| Имя поля | Тип данных |
|-----------------|-------------------|
| КодТовара | Числовой |
| Наименование | Текстовый |
| Цена | Денежный |
| НаСкладе | Числовой |

- 7) Для поля *КодТовара* выбрать тип *Числовой* и в нижней части бланка задать свойство *Размер поля – Байт*.
- 8) Для поля *Наименование* выбрать тип *Текстовый* и в нижней части бланка задать свойство *Размер поля – 25*.
- 9) Для поля *Цена* выбрать тип *Денежный* и в нижней части бланка задать свойство *Число десятичных знаков* равным 2.
- 10) Для поля *НаСкладе* выбрать тип *Числовой* и в нижней части бланка задать свойство – *Число*.
- 11) Для связи с будущими таблицами нужно задать ключевое поле. Для этого установить курсор на строку поля *КодТовара*, щелчком правой кнопки мыши открыть контекстное меню и выбрать в нем пункт *Ключевое поле*.
- 12) Закрыть окно конструктора. При закрытии ввести имя таблицы *Товары*.
- 13) По аналогии с предыдущей создать таблицу *Клиенты*, имеющую следующую структуру:

| Имя поля | Тип данных |
|--------------------|-------------------|
| КодКлиента | Числовой |
| ФамилияИмяОтчество | Текстовый |
| Телефон | Текстовый |
| АдресДоставки | Текстовый |

Примечание: Ключевым для этой таблицы будет поле *КодКлиента*. Размеры текстовых полей необходимо выбирать в соответствии с фактическими размерами содержимого.

- 14) По аналогии с предыдущими таблицами создать таблицу *Заказы* со следующей структурой:

| Имя поля | Тип данных |
|-----------------|-------------------|
| КодЗаказа | Числовой |
| КодКлиента | Числовой |
| КодТовара | Числовой |
| Количество | Числовой |
| ДатаЗаказа | Дата/время |

15) Для поля *ДатаЗаказа* выбрать тип *Дата/время* и в нижней части бланка задать свойство – *Краткая форма даты*.

Примечание: Ключевым для этой таблицы будет поле *КодЗаказа*.

3.7.4.2. Создание межтабличных связей

1) Запустить программу MS Access.

2) В окне Microsoft Access включить переключатель *Открыть базу данных*, выбрать ранее созданную базу *Торговая организация* и щелкнуть по кнопке *Ok*.

3) В окне *Торговая организация: базы данных* открыть панель *Таблицы* и убедиться, что на панели присутствуют ярлычки ранее созданных таблиц *Товары*, *Клиенты* и *Заказы*.

Примечание: Если работа продолжается без закрытия предыдущего этапа, то продолжать с пункта 4.

4) Щелчком мыши по кнопке *Схема данных* на панели инструментов (если такой кнопки нет, то раскрыть меню *Сервис* и выбрать строку *Схема данных*) открыть окно *Схема данных*. Одновременно с открытием окна *Схема данных* откроется диалоговое окно *Добавление таблицы*. На вкладке *Таблицы* этого окна выбрать таблицы, между которыми создаются связи.

5) Выбрать таблицы *Товары*, *Клиенты* и *Заказы* щелчком мыши по кнопке *Добавить*. В окне *Схема данных* откроются списки полей этих таблиц.

6) В таблице *Товары* при нажатой клавише **Ctrl** выделить поле *КодТовара* и перетащить это поле в поле *КодТовара* таблицы *Заказы*. После перетаскивания автоматически открывается диалоговое окно *Изменение связей*.

7) Выбрать на правой панели окна *Изменение связей* поле *КодТовара* таблицы *Заказы*.

- 8) В таблице *Клиенты* при нажатой клавише **Ctrl** выделить поле *КодКлиента* и перетащить это поле в поле *КодКлиента* таблицы *Заказы*. После перетаскивания автоматически открывается диалоговое окно *Изменение связей*.
- 9) Выбрать на правой панели окна *Изменение связей* поле *КодКлиента* таблицы *Заказы*.
- 10) Закрывать окно *Изменение связей* и в окне *Схема данных* проанализировать образовавшуюся связь. Щелчком левой кнопки мыши проверить возможность выделения линии связи, а щелчком правой кнопки мыши проверить открытие контекстного меню, позволяющего разорвать связь или отредактировать ее.
- 11) Закрывать окно *Схема данных*.
- 12) В окне *Торговая организация: база данных* двойным щелчком левой кнопки мыши открыть по очереди все три таблицы и ввести в них записи по каждому полю из предложенных таблиц. После заполнения таблиц их закрыть и завершить работу с программой.

Примечание: Заполнение таблиц можно выполнить, используя предварительно созданные формы для ввода. При создании форм необходимо выполнить следующие операции:

- в окне *Торговая организация: база данных* выделить строку *Создание формы с помощью мастера* и щелкнуть мышью по кнопке *Создать*;
- в появившемся окне *Новая форма* щелкнуть мышью по строке *Мастер форм*. В окне *Выберите в качестве источника данных таблицу или запрос* выбрать таблицу, для которой создается форма и щелкнуть по кнопке *Ok*;
- в появившемся окне *Создание форм* из окна *Доступные поля* перенести выделяемые поля в окно *Выбранные поля*, щелкая по кнопке *>*. Потом щелкнуть по кнопке *Далее*. Если необходимо перенести все поля таблицы, то надо щелкнуть по кнопке *>>*;
- в изменившемся окне *Создание форм: Выберите требуемый стиль*: выбрать строку *Табличный* и щелкнуть по кнопке *Далее*;
- в изменившемся окне *Создание форм: Выберите внешний вид формы*: выбрать строку *Стандартный* и щелкнуть по кнопке *Далее*;

- в изменившемся окне *Создание форм: Задайте имя формы:* ввести имя таблицы, для которой создается форма. В разделе *Дальнейшие действия:* щелкнуть в кружочке *Открыть форму для просмотра и ввода данных.* Затем щелкнуть по кнопке *Готово.*

Ниже приведено возможное содержимое таблиц.

Таблица № 1 – Товары

| Код товара | Наименование | Цена (в руб.) | На складе |
|------------|--------------------|---------------|-----------|
| 111 | Системный блок | 10999 | 500 |
| 112 | Системный блок | 8999 | 400 |
| 113 | Дисплей (ЖК) | 5650 | 500 |
| 114 | Дисплей (ЭЛ) | 4560 | 400 |
| 123 | Принтер (Лазерный) | 4675 | 350 |
| 124 | Принтер (Струйный) | 2450 | 200 |
| 132 | Клавиатура | 435 | 900 |
| 143 | Мышь | 125 | 900 |
| 154 | Модем | 1025 | 150 |
| 165 | DVD-дисковод | 999 | 200 |

Таблица № 2 – Клиенты

| Код клиента | Фамилия И.О. | Телефон | Адрес доставки |
|-------------|---------------|----------|-------------------|
| 11 | Дымов С.В. | 23-75-63 | ул. Веселая 35-3 |
| 12 | Снегов О.Л. | 74-23-87 | ул. Каменная 14-5 |
| 13 | Сугробов Ф.Д. | 28-34-12 | ул. Кривая 12-6 |
| 14 | Огнев А.С. | 54-62-23 | ул. Сонная 45-1 |
| 15 | Кошкин Е.Г. | 21-39-45 | ул. Красивая 21-4 |
| 16 | Мышкин Л.Я. | 37-86-51 | ул. Овражная 7-12 |

Таблица № 3 – Заказы

| Код заказа | Код клиента | Код товара | Количество | Дата заказа |
|------------|-------------|------------|------------|-------------|
| 1 | 11 | 165 | 12 | 25.03.10 |
| 2 | 11 | 111 | 12 | 01.04.10 |
| 3 | 11 | 113 | 12 | 03.04.10 |
| 4 | 14 | 132 | 6 | 11.04.10 |
| 5 | 15 | 143 | 8 | 26.03.10 |
| 6 | 16 | 154 | 10 | 27.03.10 |

Примечание: Содержимое таблиц вымышленное.

3.7.4.3. Создание запросов на выборку

Задание. Создать запрос на выборку заказа клиента Дымова С.В. Результирующая таблица должна содержать: фамилию, адрес и номер его телефона, наименование, цену, количество товара и дату заказа.

- 1) Запустить программу MS Access.
- 2) В окне Microsoft Access включить переключатель *Открыть базу данных*, выбрать ранее созданную базу *Торговая организация* и щелкнуть по кнопке *Ок*.
- 3) В окне *Торговая организация: базы данных* открыть панель *Запросы*. Двойным щелчком мыши по ярлычку *Создание запроса в режиме Конструктора*. Откроется бланк запроса по образцу и диалоговое окно *Добавление таблицы*.
- 4) В окне *Добавление таблицы* выбрать последовательно таблицы *Клиенты*, *Товары*, *Заказы* и для каждой таблицы щелкнуть мышью по кнопке *Добавить*. Закрывать окно *Добавление таблицы*.
- 5) Двойными щелчками мышью на именах полей в списке полей сначала таблицы *Клиенты* выбрать поля, включаемые в результирующую таблицу, *Фамилия И. О.*, *Телефон* и *Адрес доставки*. Затем из таблицы *Товары* выбрать поля *Наименование* и *Цена*. Потом из таблицы *Заказы* выбрать поля *Количество* и *Дата заказа*.
- 6) Задать условие отбора для поля *Количество* (>0 and <11). Для второго запроса задать условие отбора для поля *Дата заказа* ($>25.03.10$).
- 7) В окне *Торговая организация: база данных* открыть поочередно созданные запросы и проанализировать результирующие таблицы.

Примечание: Следует отметить, что в этом случае созданные запросы имеют существенный недостаток, а именно, пользователь базы данных работает с запросами, которые подготовил ему разработчик. Поэтому пользователь не имеет возможности изменить критерий (условие) отбора.

3.7.4.4. Создание запросов «с параметром»

Специальный тип запросов «с параметром» позволяет пользователю самому ввести критерий отбора данных на этапе запуска запроса. Таким критерием обеспечивается гибкость работы с базой.

Задание. Создать запрос, позволяющий выбирать информацию о клиентах, сделавших заказы, наименование заказанных товаров, их стоимость и количество. Эти условия пользователь должен иметь возможность задавать сам при запуске запроса.

- 1) Запустить программу MS Access.
- 2) В окне Microsoft Access включить переключатель *Открыть базу данных*, выбрать ранее созданную базу *Торговая организация: базы данных* и щелкнуть по кнопке *Ok*.
- 3) В окне *Торговая организация: базы данных* открыть панель *Запросы*. Двойным щелчком мыши по ярлычку *Создание запроса в режиме Конструктора* открыть бланк запроса по образцу.
- 4) Создать запрос на выборку, основанный на таблицах *Товары*, *Клиенты* и *Заказы*. В этот запрос должны войти следующие поля:
 - *Фамилия клиента*;
 - *Наименование товара*;
 - *Цена*;
 - *Количество*.
- 5) В строке *Условие отбора* поля *Фамилия* ввести текст [*Введите фамилию клиента*].
- 6) В строку *Условие отбора* ввести текст [*Введите максимальную цену*].
- 7) Закрывать запрос. При закрытии сохранить его под именем *Заказы клиента*.
- 8) В окне *Торговая организация: база данных* открыть панель *Запросы* и запустить запрос *Заказы клиента* – на экране появится диалоговое окно *Введите фамилию клиента*. В этом окне ввести критерий цены системного блока, например 10000, и щелкнуть по кнопке *Ok*. По результатам запроса будет сформирована результирующая таблица.
- 9) Закрывать все объекты базы данных и программу MS Access.

3.7.5. Контрольные вопросы

- 1) Что из нижеперечисленных не является объектом базы данных:
а) запрос; б) форма; в) отчет; г) справка?
- 2) Как именуется элементы структуры таблицы базы данных:
а) поля; б) имена; в) записи; г) строки; д) столбцы?
- 3) Какой из типов данных не поддерживается системой управления базами данных:
а) текстовый; б) дата/время; в) логический; г) строковый?
- 4) Что означает аббревиатура СУБД:
а) свойства удаленной базы данных;
б) система управления большими данными;
в) система удаления заблокированных данных;
г) система управления базами данных?
- 5) Что является основной поддержкой любой СУБД:
а) встроенный язык PL/SQL;
б) структурированный язык запросов SQL;
в) встроенный язык INFORMIX 4GL;
г) встроенный язык SQL Server?
- 6) Что представляет собой запрос в базе данных:
а) форму ввода информации в базу данных;
б) инструкцию на отбор записей в базе данных;
в) формат хранения информации;
г) вопрос к операционной системе?
- 7) Что собой представляет поиск данных в базе:
а) процедуру определения дескрипторов базы данных;
б) определение значений данных в текущей записи;
в) процедуру выделения значений данных, однозначно определяющих ключевой признак записи;
г) процедуру выделения из множества записей подмножества, которое удовлетворяет заданному условию?
- 8) Что является основным объектом любой реляционной базы данных:
а) таблица; б) запрос; в) форма; г) отчет?
- 9) Что определяет вид хранимой информации:
а) тип соответствующего поля данных;
б) связи между данными;
в) устойчивость структур данных;
г) вложенность структур данных?
- 10) Что не является характеристикой поля базы данных:

- а) длина; б) имя; в) тип данных; г) запись?
- 11) Что относится к основным объектам СУБД MS Access:
- а) отношение, нормальная форма, запись;
 - б) таблица, форма, запрос, отчет;
 - в) запись, файл, структура, модель;
 - г) ключ, поле, точность, безопасность?
- 12) Какую структуру имеет реляционная таблица базы данных:
- а) объемную; б) многослойную; в) квадратную;
 - г) прямоугольную?
- 13) На что ориентирована структура данных в СУБД:
- а) табличное представление; б) древовидную структуру;
 - в) модель данных; г) сетевую структуру?
- 14) Что представляет собой запрос к БД
- а) форму ввода информации в БД;
 - б) формат хранения информации;
 - в) инструкцию на отбор записей в базе данных;
 - г) вопрос к операционной системе?
- 15) Какой из языков является языком запроса к реляционным базам данных:
- а) Pascal; б) C#; в) SQL; г) SSH?
- 16) Назовите верное из утверждений:
- а) язык SQL не имеет международного стандарта;
 - б) язык SQL расшифровывается как Structured Query Language;
 - в) язык SQL прекратил свое развитие в 1996 году;
 - г) язык SQL используется для доступа к данным в иерархических базах.
- 17) Иерархическая, сетевая, реляционная – это:
- а) системы обработки данных;
 - б) модели предметной области;
 - в) модели структур данных;
 - г) структуры формирования запросов к базе данных?
- 18) Для организации чего предназначен язык манипулирования данными СУБД:
- а) обработки данных в базе;
 - б) семантической обработки информации;
 - в) типов данных, представленных в файлах СУБД;
 - г) структуры базы данных?
- 19) Что представляет собой информационный запрос:

- а) сигнал или сообщение оператору о необходимости ввода данных;
 - б) запрос к информационной системе, содержащий условие на поиск и выборку данных;
 - в) входное сообщение с требованием выделения необходимых ресурсов;
 - г) необработанное обращение к операционной системе?
- 20) Что используется для получения таблицы из совокупности связанных таблиц путем выбора полей, удовлетворяющих заданным условиям:
- а) запросы; б) схемы; в) отчеты; г) формы?
- 21) Что такое время отклика на запрос:
- а) время на сервере в момент запроса;
 - б) время выполнения транзакции;
 - в) интервал, который проходит от момента передачи запроса до момента начала получения информации;
 - г) интервал исполнения SQL запроса?
- 22) Каким может быть ключ к записям в базе данных:
- а) простым; б) составным; в) первичным; г) вторичным;
 - д) дополнительным; е) включающим; ж) отчетным;
 - з) запросным?
- 23) Какого вида изменений записей в базе данных не существует:
- а) замена; б) исключение; в) вставка; г) объединение?
- 24) Какие возможности при работе с файлами предоставляют пользователю современные СУБД:
- а) создания новых объектов БД;
 - б) демонстрации инфологической модели;
 - в) модификации уже существующих объектов в БД;
 - г) определения схемы информационных обменов;
 - д) создания и переименования ранее созданных объектов;
 - е) дополнения функциональных возможностей?
- 25) Что представляет собой поиск данных в базе:
- а) определение значений данных в текущей записи;
 - б) процедуру определения дескрипторов базы данных;
 - в) процедуру выделения значений данных, однозначно определяющих ключевой признак записи;

- г) процедуру выделения из множества записей подмножества, записи которого удовлетворяют заранее поставленному условию?
- 26) Каким образом в СУБД MS Access создаются отчеты:
- а) с помощью мастера отчетов;
 - б) путем ввода данных;
 - в) с помощью мастера таблиц;
 - г) в режиме конструктора;
 - д) в режиме предварительного просмотра?
- 27) Что такое ключ базы данных:
- а) набор символов, ограничивающий вход в автоматизированную систему с базой данных;
 - б) язык запроса к базе данных;
 - в) часть записи, совокупность ее полей, предназначенных для формирования индексного файла;
 - г) уникальный идентификатор записи в базе данных?

3.8. Справочная система MS Office

Организация справочной системы для всех приложений, входящих в состав MS Office (на примере MS Office 2003), выполнена одинаково, поэтому, изучая ее в данной главе, не будем уточнять какое-либо конкретное приложение, за исключением случаев, когда это принципиально важно. То есть будем приводить примеры использования справки в любых из рассмотренных в пособии приложений MS Office.

Вызов справки осуществляется нажатием клавиши **F1** или с помощью команды меню *Справка / Справка: Microsoft <приложение>*. Здесь *<приложение>* определяет название активного приложения (Word, Excel, PowerPoint и др.), с которым пользователь работает в данный момент. Выбор названной команды приводит к появлению окна *Справка <приложение>*. Чтобы завершить работу со справкой, достаточно нажать кнопку *Заккрыть*, расположенную в строке заголовка. Рассмотрим возможности справочной системы.

- 1) *Навигационная панель* предназначена для просмотра истории работы с областями задач. Кнопка *Назад* позволяет перейти к предыдущей области задач (кнопка не активна (серая), если достигнут конец истории), а кнопка *Вперед* – вернуться к следующей (кнопка не активна (серая), если достигнуто начало истории). Нажатие кнопки *Домашняя*

страница приводит к активизации области задач *Приступая к работе*.

- 2) *Поле ввода ключевых слов* позволяет задать тему поиска.
- 3) Нажатие кнопки *Начать поиск* приводит к запуску процедуры поиска. Инициировать поиск также можно, нажав клавишу **Enter** в момент, когда курсор мыши находится в поле ввода ключевых слов.
- 4) *Команды Office Online* представляют собой мощный инструмент получения обновленной справочной информации по MS Office, однако для их использования требуется подключение к интернету. Следует отметить, что информации инсталлированной справочной системы MS Office чаще всего достаточно для поиска ответов на возникающие вопросы.
- 5) *Ссылки на дополнительную информацию*, как правило, содержат информацию о новых функциях, реализованных в данной версии MS Office.

Полезным элементом области задач *Справка* является *Оглавление* (расположено под полем ввода ключевых слов), при использовании которого в текущую область задач загружается оглавление справочной системы, включающее в себя все существующие разделы справки активного приложения MS Office. Раскрыть содержимое раздела справки можно двойным кликом мыши на его имени. Имена разделов отмечены картинкой в виде книжки, а страниц справочных данных – вопросительным знаком, заключенным в кружок. После двойного клика мышью на имени страницы открывается дополнительное окно с ее содержимым.

Возможны два способа взаимного размещения окна справочной системы MS Office и активного приложения:

- 1) отображать с перекрытием окон (активизируется кнопкой *Развернуть* панели инструментов окна справочной системы);
- 2) уместить окна на экране (активизируется командой *Автомозаика* панели инструментов окна справочной системы).

В первом случае активное приложение MS Office и справочная система представлены на экране как два независимых окна MS Windows. Во втором случае размеры окна активного приложения и справочной системы автоматически подбираются таким образом, чтобы вся информация этих двух окон полностью отображалась пользователю (умещалась) на экране.

Информация в справочной системе MS Office представлена в виде *гипертекста* – текста, содержащего ссылки на другие разделы. Если кликнуть по ссылке, то, в зависимости от ее типа, возможны две реакции:

- 1) система перейдет к другому разделу, отвечающему данной ссылке;
- 2) ссылка представляет собой некий термин, и в этом случае пользователю будет отображена дополнительная информация, поясняющая выбранный термин (повторный клик мыши по ссылке скроет эту информацию).

Перемещаться по ранее просмотренным страницам справочной системы можно с помощью кнопок *Назад* и *Вперед* панели инструментов окна справочной системы MS Office. Данные кнопки неактивны, когда достигнуты конец или начало списка истории просмотренных страниц соответственно.

Вывод текущей страницы справки на печать возможен при нажатии на кнопку *Печать*, расположенную также на панели инструментов окна справочной системы.

Имея представление об общих способах работы со справочной системой MS Office, рассмотрим некоторые ее дополнительные возможности.

С помощью кнопки *Другие области задач* можно переключаться между доступными различными областями. После нажатия на эту кнопку появляется контекстное меню, состоящее из нескольких групп команд (группы команд меню разделены горизонтальной сплошной линией). Верхняя часть (группа команд) этого контекстного меню для любых приложений MS Office одинакова, так как в ней располагаются названия рабочих областей, относящиеся к справочной системе.

Для переключения между рабочими областями следует кликнуть мышью на имени нужной области рассматриваемого контекстного меню. В приложениях MS Office доступны следующие области задач.

- 1) *Приступая к работе* – активизируется при запуске приложения и содержит команды Office Online, а также имена последних открытых файлов и команду создания нового файла открытого приложения MS Office.
- 2) *Справка*.

- 3) *Поиск* – позволяет выполнить расширенный поиск информации.
- 4) *Коллекция клипов* – предназначена для поиска картинок и клипов из коллекции MS Office.
- 5) *Справочные материалы* – данная область задач позволяет искать произвольную информацию, используя соответствующие серверы интернета.
- 6) *Буфер обмена* – отображает пользователю содержимое буфера обмена MS Windows и предоставляет средства управления им, такие, как удаление и изменение хранящейся информации.

Особого внимания заслуживает область задач MS Office *Поиск*. В ее центральной области отображаются результаты поиска либо информационные сообщения справочной системы. Область *Поиск* включает два элемента: непосредственно строку ввода ключевых слов (снизу) и область определения границ поиска. Выделяются следующие границы поиска:

- 1) *Office в интернете*;
- 2) *Автономная справка*;
- 3) *Картинки и мультимедиа*;
- 4) *Справочные материалы*.

Используя эти границы, пользователь может уменьшить область поиска, что значительно повысит скорость отыскания информации.

В качестве рекомендации можно назвать следующие правила, позволяющие наиболее эффективно использовать справочную систему MS Office для поиска информации:

- 1) если требуется выполнить поиск картинок, клипов и т.п., то в качестве границ поиска следует выбирать *Картинки и мультимедиа*;
- 2) для поиска произвольной справочной информации (не относящейся к MS Office) лучше всего использовать *Справочные материалы*;
- 3) информацию о возможностях MS Office, способах и приемах работы в нем следует искать в разделе *Автономная справка* и только в случае, когда автономный поиск результатов не дал, – искать в *Office в интернете*.

В заключение, в качестве примера использования справочной системы MS Office, рассмотрим задачу поиска информации о форматировании значений ячеек в MS Excel и о способах форматирования текста абзаца MS Word.

Пример 1. Найти возможные способы форматирования значений ячеек MS Excel.

Для решения этой задачи запустим MS Excel. После запуска справа будет отображаться область задач *Приступая к работе*. Перейдем в область задач *Поиск*, для чего кликнем мышью на кнопке *Другие области задач* и в появившемся меню выберем команду *Поиск*. Укажем границу поиска *Автономная справка*, в поле ключевых слов введем фразу “*способы форматирования содержимого ячеек*” и нажмем кнопку *Поиск*. Результаты поиска будут отображены в виде названия страниц справочной системы. Таких страниц было найдено всего 8 (отображается в скобках после слова *Результаты*), поэтому не составляет труда, просмотрев их все, определить, что нужная информация находится на странице *Форматирование листов и данных*.

Пример 2. Найти возможные способы выравнивания абзацного текста.

Для выполнения этого задания поступаем аналогично предыдущему примеру с той разницей, что запускаем MS Word, и в поле ключевых слов введем фразу “*выравнивание абзацного текста*” (можно также ввести фразу “*выравнивание текста в абзаце*”). В результатах поиска (их всего 16) присутствует страница справочной системы с названием *О выравнивании текста и интервалах*. Нетрудно догадаться – это именно то, что было нужно найти.

Глава 4. Модели решения функциональных и вычислительных задач

Моделирование – исследование каких-либо явлений, процессов или систем объектов путем построения и изучения их моделей. Модели используются для определения или уточнения характеристик и рационализации способов построения вновь конструируемых объектов.

Моделирование – одна из основных категорий теории познания: на идее моделирования по существу базируется любой метод научного исследования – как теоретический (при котором используются различного рода знаковые, абстрактные модели), так и экспериментальный (использующий предметные модели).

4.1. Понятие модели и объекта

В своей деятельности человек повсеместно использует модели, то есть создает образ того объекта, с которым ему приходится иметь дело.

Что же такое модель? В переводе с латинского языка *modelium* означает меру, образ, способ. Модель может представлять собой материальный объект или схему, математические формулы или набор данных и алгоритм их обработки и тому подобное.

Модель – это искусственно созданный объект, дающий упрощенное представление о реальном объекте, процессе или явлении.

Объект, для которого создается модель, называют оригиналом или прототипом. Любая модель не является абсолютной копией своего оригинала, она лишь отражает некоторые его качества и свойства, наиболее существенные для выбранной цели исследования. При создании модели всегда присутствуют определенные допущения и гипотезы.

Для любого объекта может быть создано множество моделей, различных по сложности и степени сходства с оригиналом. Однако, если разные объекты имеют общие свойства, то для них может быть создана одна модель. Например, для одного и того же человека, как объекта, можно создать следующие модели: модель внешних параметров (рост, вес, цвет волос, цвет глаз и другие), модель биоритмов, социальная модель и другие. В то же время каждая из названных моделей может быть общей для разных людей.

Однако никакая модель не может заменить сам объект. Но при решении конкретной задачи для изучения определенных свойств объекта модель оказывается полезным, а часто и единственным инструментом исследования. Моделирование – это метод, позволяющий переносить информацию от реальной системы к модели и наоборот.

Итак, модель:

- это упрощенный образ оригинала, неразрывно с ним связанный, отражающий существенные свойства, связи и отношения оригинала;
- это мысленно представляемая или материально реализуемая система, исследование которой дает новую или подтверждает имеющуюся информацию об оригинале.

4.2. Классификация моделей

Границы между моделями различных типов или классов, а также отнесение модели к какому-то типу или классу чаще всего условны. Наиболее распространенные признаки, по которым классифицируются модели:

- цель исследования;
- фактор времени;
- область знаний;
- способ представления.

По целям исследования выделяются следующие:

- учебные модели;
- опытные модели;
- имитационные модели;
- игровые модели;
- научно-технические модели.

По фактору времени разделяются:

- динамические модели;
- статические модели.

Статическая модель отражает строение и параметры объекта, поэтому ее называют также структурной моделью. Она описывает объект в определенный момент времени, дает срез информации о нем. Динамическая модель отражает процесс функционирования объекта или изменение и развитие процесса во времени.

По областям знаний выделяются модели:

- биологические;
- экономические;
- исторические;
- социологические.

Любая модель имеет конкретный вид, форму или способ представления, она всегда из чего-то и как-то сделана. В этом классе, прежде всего, рассматриваются:

- материальные модели;
- нематериальные (информационные) модели.

Материальная модель – это материальная копия объекта моделирования. Такая модель всегда имеет реальное воплощение, воспроизводит внешние свойства или внутреннее строение, либо действия объекта-оригинала. Материальное моделирование использует экспериментальный (опытный) метод познания. Примеры: глобус – географическая модель земного шара, робот – модель действий человека на производстве.

Нематериальное моделирование использует теоретический метод познания (мысленный, абстрактный, идеальный, информационный).

Информационная модель – это модель, с помощью которой описываются и изучаются свойства и состояние объекта, процесса или явления, а также их связи и отношения с окружающим миром.

В информационных моделях объекты представляются в виде мыслей, словесных описаний, текстов, рисунков, таблиц, схем, чертежей, формул и так далее. У информационной модели нет материального воплощения, она строится только на основе информации. Одна и та же модель может одновременно относиться к разным классам. Например, программы, имитирующие движение тел, используемые на занятиях по физике (область знания) с целями обучения (цель использования). Они являются динамическими, так как учитывают положение тела в разные моменты времени, и алгоритмическими по способу реализации.

Форма представления информационных моделей зависит от способа кодирования и материального носителя. В этом смысле информационные модели подразделяются на:

- мысленные;
- вербальные;

- образные;
- образно-знаковые;
- знаковые.

Мысленная (интуитивная) модель – это мысленное представление об объекте. Разработка такой модели всегда предшествует созданию объекта, являясь одним из этапов творческого процесса.

Вербальная модель – это представление информационной модели средствами естественного разговорного языка. Форма представления такой модели – устное или письменное сообщение.

Образная модель – это выражение свойств оригинала с помощью образов. Например, рисунки, художественные полотна, фотографии, кинофильмы. Сюда же относятся геометрические модели – информационные модели, представленные средствами графики.

Образно-знаковые модели – это модели, построенные с использованием знаковых образов какого-либо вида: схемы, чертежи, графы, графики, планы, карты. К этой группе относятся структурные информационные модели, создаваемые для наглядного изображения составных частей и связей объектов. Например: таблицы, схемы, блок-схемы.

Знаковые (символические) модели – это модели, в которых используются алфавиты формальных языков (условные знаки, специальные символы, буквы, цифры) и предусматривающие совокупность правил оперирования с этими знаками. Примеры: специальные языковые системы, физические или химические, формулы, математические выражения. К знаковым моделям относятся математические, логические, программные тексты

Математическая модель – это способ представления информационной модели, отображающий связь различных параметров объекта через математические формулы и понятия. Такие модели предназначены для решения численных задач.

В тех случаях, когда моделирование ориентировано на исследование моделей с помощью компьютера, одним из его этапов является разработка компьютерной модели. *Компьютерная модель* – это созданный за счет ресурсов компьютера виртуальный образ, качественно и количественно отражающий внутренние свойства и связи моделируемого объекта, иногда передающий и его внешние характеристики. Компьютерная модель представляет собой матери-

альную модель, воспроизводящую внешний вид, строение или действие моделируемого объекта. Разработке компьютерной модели предшествуют мысленные, вербальные, структурные, математические и алгоритмические модели.

4.3. Этапы компьютерного моделирования

- 1) Описание (постановка) задачи. Задача формулируется на обычном языке. Главное здесь – определить объект моделирования и представить конечный результат.
- 2) Определение целей моделирования. Прежде всего, необходимо определить цели моделирования в соответствии с поставленной задачей, которые оказывают направляющее влияние на весь процесс моделирования.
- 3) Разработка информационной модели. Этот этап включает содержательное описание объекта с дальнейшей его формализацией. Выделяются объекты моделирования и дается их развернутое содержательное описание. Описание включает сведения о природе объектов, их зависимости и связи, перечисление свойств и характеристик отдельных объектов и моделируемой системы в целом. Свойства учитываются не все, а лишь те, которые можно считать существенными в зависимости от выбранной цели. В результате выстраивается описательная информационная модель объекта, которую называют вербальной. При формализации модели осуществляется переход от описательной модели к конкретному математическому наполнению. Указывается перечень параметров, которые влияют на поведение объекта – исходные данные, и которые желательно получить – результат. Формализуются зависимости между выделенными параметрами, накладываются ограничения на их допустимые значения. *Формализация* – процесс построения информационных моделей с помощью формальных языков. Является этапом перехода от словесного описания связей между признаками объекта к описанию, использующему некоторый формальный язык кодирования. Как правило, результатом формализации является математическая модель. Помимо алгебраических зависимостей, математическая модель может иметь геометрическое или логическое представление. Другими знаковыми формами модели, воз-

можными здесь, являются таблицы, блок-схемы, чертежи – все, что способствует лучшему представлению задачи.

- 4) Разработка компьютерной модели. На этом этапе формализованная модель преобразуется в модель компьютерную. Существует множество программных комплексов и сред, которые позволяют проводить построение и исследование моделей. К ним относятся графические среды, текстовые редакторы, среды программирования, электронные таблицы, математические пакеты, СУБД, HTML-редакторы. Одну и ту же задачу можно решить, используя различные среды. От выбора программной среды зависит алгоритм построения компьютерной модели и форма его представления. Это может быть программа, реализуемая в какой-то среде программирования, или последовательность технических приемов в прикладной среде. Выполняется реализация компьютерной модели по законам выбранной среды. Получаемые результаты более информативны и наглядны, если они выведены в виде графиков, диаграмм, траекторий и тому подобное. Модель нельзя назвать готовой, пока в ней содержатся ошибки. Обязателен этап, который называется тестированием модели. В программировании этот этап состоит из процессов трансляции и отладки программы. Уже на этапе тестирования может выявиться необходимость корректировки (изменения) исходной модели, прежде всего в той ее части, где заложено смысловое содержание.
- 5) Исследование модели. Исследование заключается в проведении серии экспериментов, удовлетворяющих целям моделирования, и накоплении результатов. *Эксперимент* – это опыт, который производится с объектом или моделью. Он заключается в выполнении некоторых действий, чтобы определить, как реагирует экспериментальный образец на эти действия. Каждый эксперимент должен сопровождаться осмыслением итогов, что служит основой для анализа результатов и принятия решений.
- 6) Анализ результатов моделирования. Полученные результаты позволяют ответить на вопрос: «Продолжить исследование или закончить?» Если результаты не соответствуют целям поставленной задачи, значит, на предыдущих этапах были

допущены ошибки. Если такие ошибки выявлены, то требуется корректировка модели.



Итак, модели, предназначенные для имитации и выполнения некоторых существенных функций реальных объектов и соответствующие поставленным целям моделирования, являются моделями решения функциональных задач. Математические и компьютерные модели, созданные на основе математических, являются моделями решения вычислительных задач.

4.4. Объекты моделирования. Свойства объектов

Объект – это то, о чем идет речь. Это может быть предмет, явление, процесс, действие, чувство или ощущение и тому подобное. *Объект* – это то, что имеет название, является чем-то целым и обладает некоторыми свойствами или характеристиками.

Свойства объекта – это присущие ему качества, характеристики, признаки. Свойства объекта, которым присваивается

конкретное значение, называются параметрами объекта. Параметры могут быть числовыми (количество, размер, вес, возраст), тогда их называют величинами. Параметры могут быть качественными (цвет, форма, вкус, материал), в этом случае их называют признаками. Параметры могут быть неизменными либо меняются со временем. В этом случае говорят об изменении состояния объекта.

Действия, которые производятся над объектами или которые они производят сами, относятся к характеристикам объектов. Действия, свойственные только данному объекту, позволяют отличить его от других и могут быть связаны с его назначением.

Когда присутствует последовательное воздействие одного объекта на другой и при этом происходит изменение некоторых параметров самого объекта или объекта взаимодействия, говорят о процессе. *Процесс* – это последовательная смена состояний объекта как результат некоторых воздействий.

Среда и условия, в которых существует объект, являются характеристиками объекта, которые также обладают свойствами и параметрами.

4.5. Системы объектов

Любой объект можно рассматривать как совокупность взаимодействующих между собой частей. В этом случае говорят о системе. Входящие в систему объекты называют элементами системы. *Система* – это совокупность взаимодействующих между собой составных частей (элементов), составляющих единое целое, выделенных из окружающей среды и функционирующих с определенной целью.

Как правило, любая система сама является элементом более крупной системы. Так же и элемент некоторой системы является системой входящих в него частей. Степень дробления может быть бесконечной и определяется целью исследования.

Системы могут быть естественными, появившимися без участия человека. Другие системы, искусственные, созданные или создаваемые человеком с определенными целями. Системы могут быть материальными, нематериальными и смешанными.

Главное свойство любой системы – это возникновение системного эффекта. Суть его заключается в следующем: при объединении элементов в систему у нее появляются новые свойства, которыми не

обладал ни один из элементов системы в отдельности. Это свойство называется *эмерджентностью*.

Всякая система представляет собой некоторый объект, который можно выделить из окружающей среды. Система взаимодействует с окружающей средой. Воздействия среды на систему называют *входами* (входом) системы, а воздействия системы на среду называют *выходами* (выходом) системы. Под термином воздействие можно понимать и материальные объекты, которыми обмениваются среда и система, и физическое воздействие, и информацию. Для всякой системы можно перечислить множество входов и выходов, совсем не обязательно связанных с ее прямым назначением. Не всегда можно знать, как устроен внутри данный объект. Однако важно лишь знать, к каким результатам на выходе приведут определенные воздействия на входе системы. В таких случаях систему можно рассматривать как «*черный ящик*». Представить некоторую систему в виде «*черного ящика*» – это значит указать ее входы и выходы.

4.6. Некоторые разновидности моделей

Модель – образец (эталон, стандарт, тип, марка изделия) для массового изготовления какого-либо изделия или конструкции. Например, изделие (из легкообрабатываемого материала), с которого снимается форма для воспроизведения (например, посредством литья) в другом материале. Разновидности таких моделей – лекала, шаблоны. Позирующий художнику натурщик или изображаемые предметы («натура») – это модель. Устройство, воспроизводящее, имитирующее строение и действие какого-либо другого («моделируемого») устройства в научных, производственных (при испытаниях) или спортивных целях – это модель.

В широком смысле модель это любой образ или аналог (мысленный или условный) какого-либо объекта, процесса или явления («оригинала» данной модели), используемый в качестве его «заместителя», «представителя». Ими могут быть изображения, описания, схемы, чертежи, графики, планы, карты и другое.

В математике и логике моделью какой-либо системы аксиом называют любую совокупность (абстрактных) объектов, свойства которых и отношения между которыми удовлетворяют данным ак-

сиомам, служащим тем самым совместным (неявным) определением такой совокупности.

Модель в языкознании:

- абстрактное понятие эталона или образца какой-либо системы (фонологической, грамматической и т. п.);
- представление самых общих характеристик какого-либо языкового явления;
- общая схема описания системы языка или какой-либо его подсистемы.

4.7. Некоторые ключевые понятия и примеры

- 1) Модель может быть построена для любого объекта, явления или процесса.
- 2) Определение целей моделирования осуществляется на этапе постановки задачи.
- 3) Новый объект, отражающий существенные особенности изучаемого объекта, процесса или явления, называется моделью.
- 4) Модель считается адекватной, если она описывает все существенные свойства объекта, процесса или явления.
- 5) К отношению «модель – объект» не относится понятие «Земля – глобус».
- 6) Моделью Земли, используемой для определения законов ее движения вокруг Солнца, является материальная точка.
- 7) К предметным моделям относятся:
 - авиамодель истребителя;
 - полоса препятствий.
- 8) Информационной моделью является формула расчета сопротивления при последовательно-параллельном соединении.
- 9) Алгоритмы решения задач являются статическими информационными моделями.
- 10) Модель гравитационного взаимодействия двух тел, записанная в виде формул, является описательной информационной моделью.
- 11) Функциональные модели служат для имитации поведения реальных объектов в сложных условиях.
- 12) Моделями типа «черный ящик» являются модели, описывающие зависимость выходных параметров объекта от входных без учета его внутренней структуры.

- 13) Пара понятий «производство – продукция» описывается отношением «процесс – результат».
- 14) Пара понятий «молекула – атом» описывается отношением «целое – часть».
- 15) Пара понятий «самолет – двигатель» описывается отношением «система – элемент».
- 16) Результатом процесса формализации является математическая модель.
- 17) Пара понятий «автомобиль – чертеж» описывается отношением «объект – модель».
- 18) Представление файлов и каталогов в виде дерева является иерархической информационной моделью.

Глава 5. Основы алгоритмизации и программирования

5.1. Понятие алгоритма. Свойства и способы описания

Алгоритм – это конечная последовательность точно определенных действий, приводящих к решению поставленной задачи. При составлении алгоритмов следует учитывать ряд требований, выполнение которых приводит к формированию необходимых свойств.

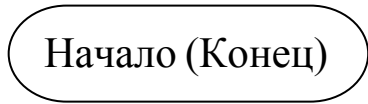
- Алгоритм должен быть однозначным, исключая произвольность толкования любого из предписаний и заданного порядка исполнения. Это свойство алгоритма называется *определенностью*.
- Любой алгоритм должен иметь только одну точку входа (начало) и одну точку выхода (окончание).
- В результате реализации процесса, предусмотренного алгоритмом, должны быть получены результаты или выдано сообщение о невозможности решения задачи. Это свойство алгоритма называется *результативностью*.
- Способность алгоритма обеспечить решение любого большого количества однотипных задач с различными исходными данными называется *массовостью*.
- Расчленение процесса, предусмотренного алгоритмом, на отдельные этапы (элементарные операции) принято называть свойством *дискретности*.

Для строгого задания различных структур данных и алгоритмов, их обработки требуется иметь такую систему формальных обозначений и правил, чтобы смысл всякого используемого предписания трактовался точно и однозначно. Для выполнения этого условия или требования существуют следующие способы описания алгоритмов:

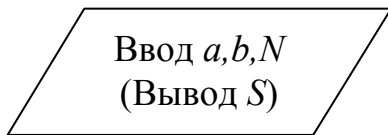
- словесное описание (запись на естественном языке);
- графическое описание (блок-схемы);
- программное описание (тексты программ на алгоритмическом языке).

5.1.1. Графический способ описания

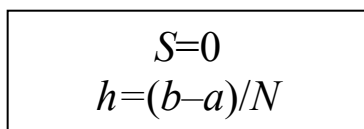
Для изображения алгоритма в виде блок-схемы применяются следующие основные геометрические фигуры.



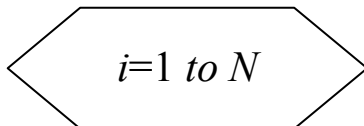
– Начало или конец алгоритма (пуск-останов).



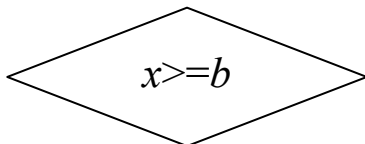
– Ввод исходных данных или вывод результатов (ввод-вывод).



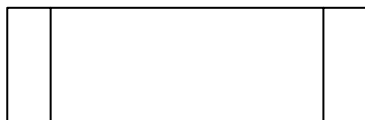
– Выполнение операции или группы операций (процесс).



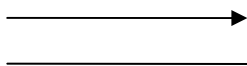
– Цикл по счетчику, то есть с определенным (конечным) числом шагов (подготовка или модификация).



– Выбор направления выполнения алгоритма в зависимости от некоторых условий (решение).



– Модуль подпрограммы-процедуры или подпрограммы-функции (типовой или predetermined процесс).



– Линии потока, соединяющие фигуры блок-схемы. Направление линии указывается при ее ходе слева направо. Изменение направления линии задается под прямым углом.



– Разрыв линии потока.



– Ссылка на другую страницу.

5.1.2. Базовые конструкции алгоритмов

При разработке и составлении блок-схемы того или иного алгоритма применяются следующие базовые конструкции: линейная, циклическая и ветвящаяся. В теории алгоритмов доказано, что на основе этих трех конструкций может быть построен любой алгоритм. Абсолютно все системы программирования располагают языковыми средствами для реализации основных алгоритмических конструкций.

Линейная конструкция (*следование*) – это последовательное выполнение операций без повторов и разветвлений (рис. 5.1).

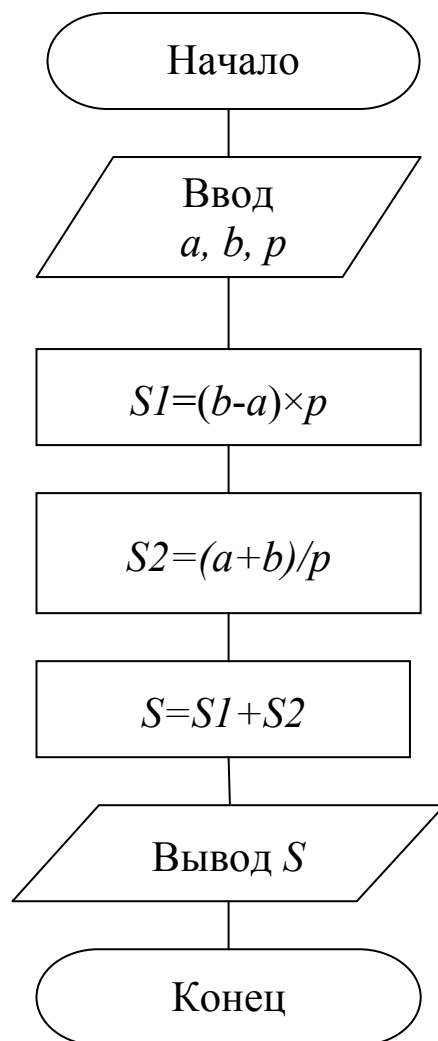


Рис. 5.1. Линейная конструкция

Ветвящаяся конструкция (*ветвление*) – это выполнение операций по одному из нескольких направлений в зависимости от заданных условий (рис. 5.2).

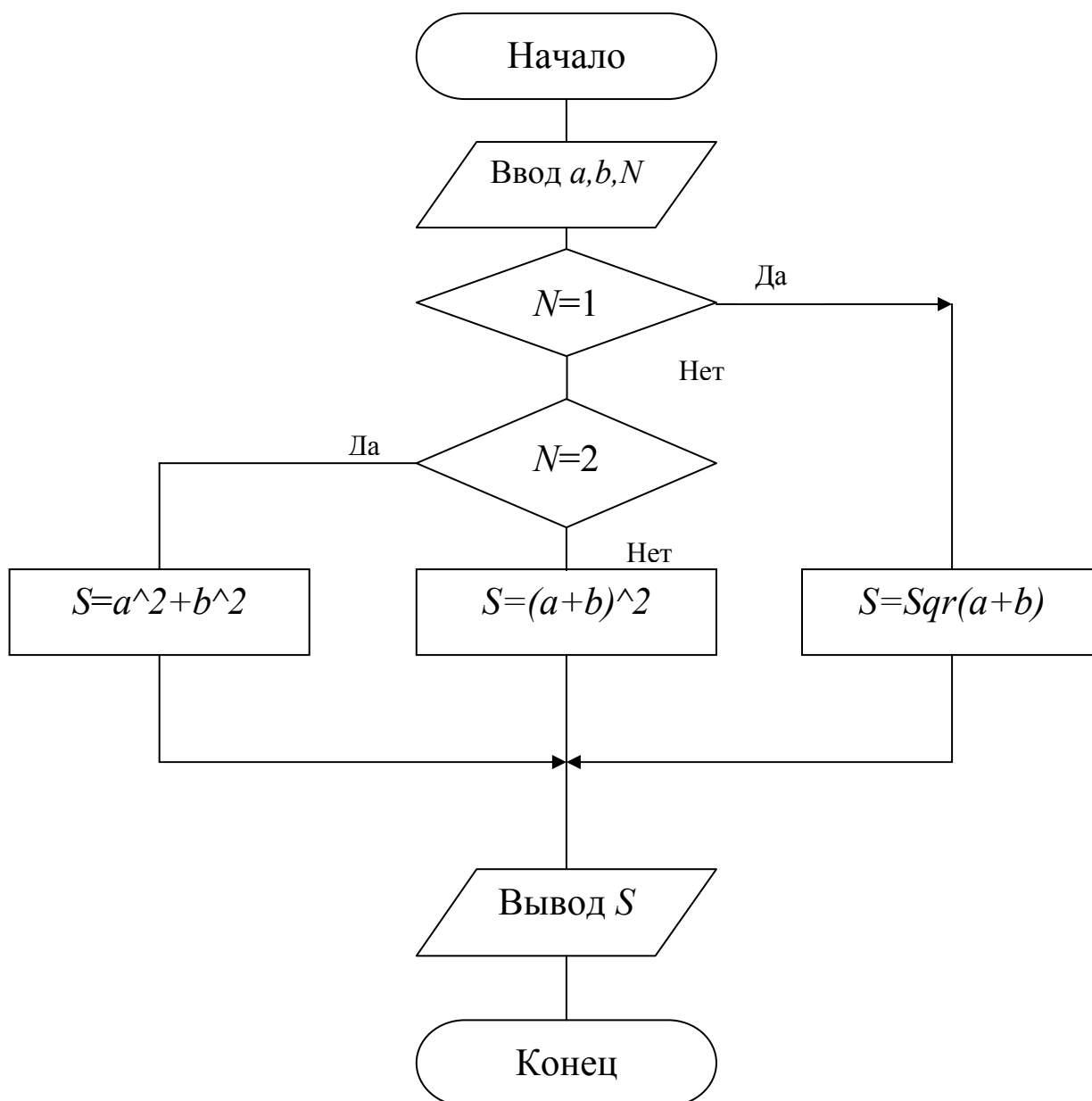


Рис. 5.2. Ветвящаяся конструкция

Циклы (конструкция *повторения*) используются для организации повторного выполнения какой-либо операции (инструкции) или блока операций (инструкций). Цикл состоит из двух частей: *условия цикла* и *тела цикла*. У каждого цикла есть *параметр*. *Параметр цикла* – это переменная, которая изменяется в теле цикла и задействована в условии его окончания. Для организации повторов могут применяться следующие виды циклических конструкций: цикл с предусловием, цикл с постусловием, цикл с фиксированным количеством повторов или цикл по счетчику.

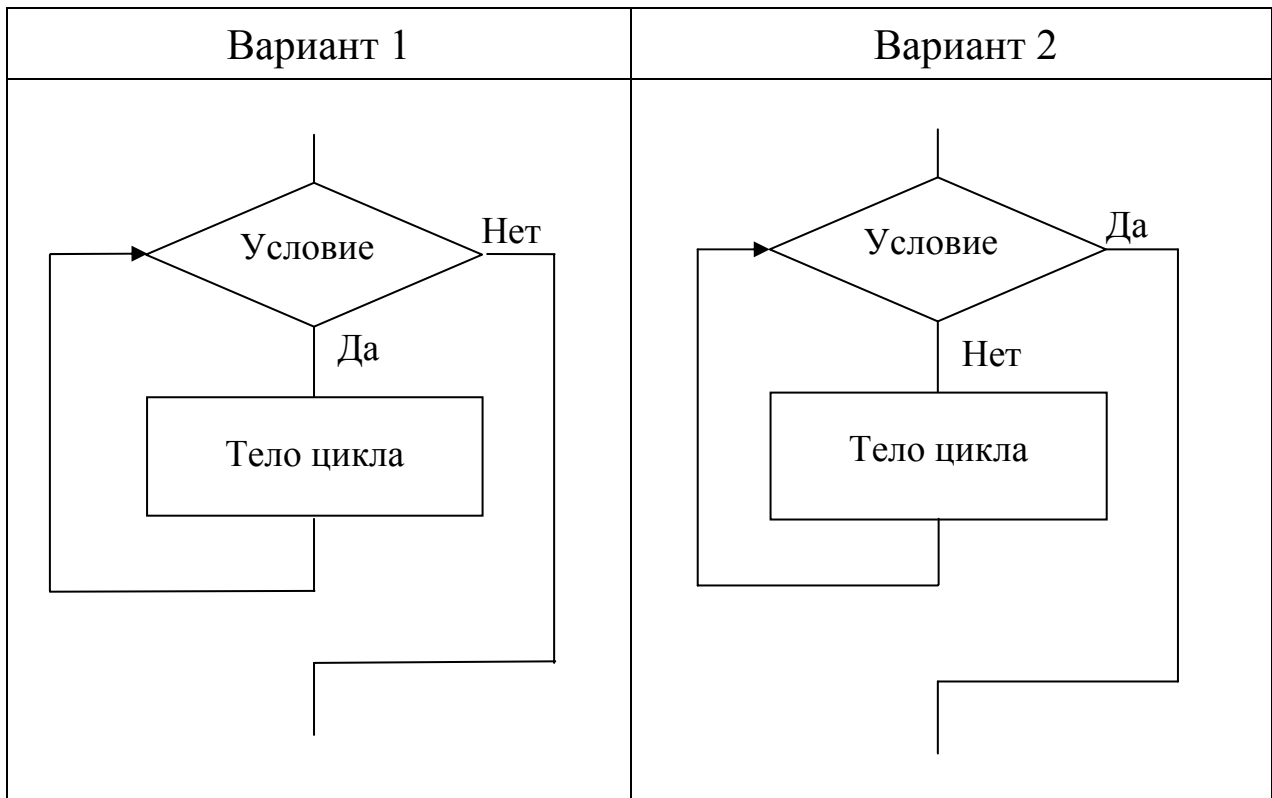


Рис. 5.3. Цикл с предусловием

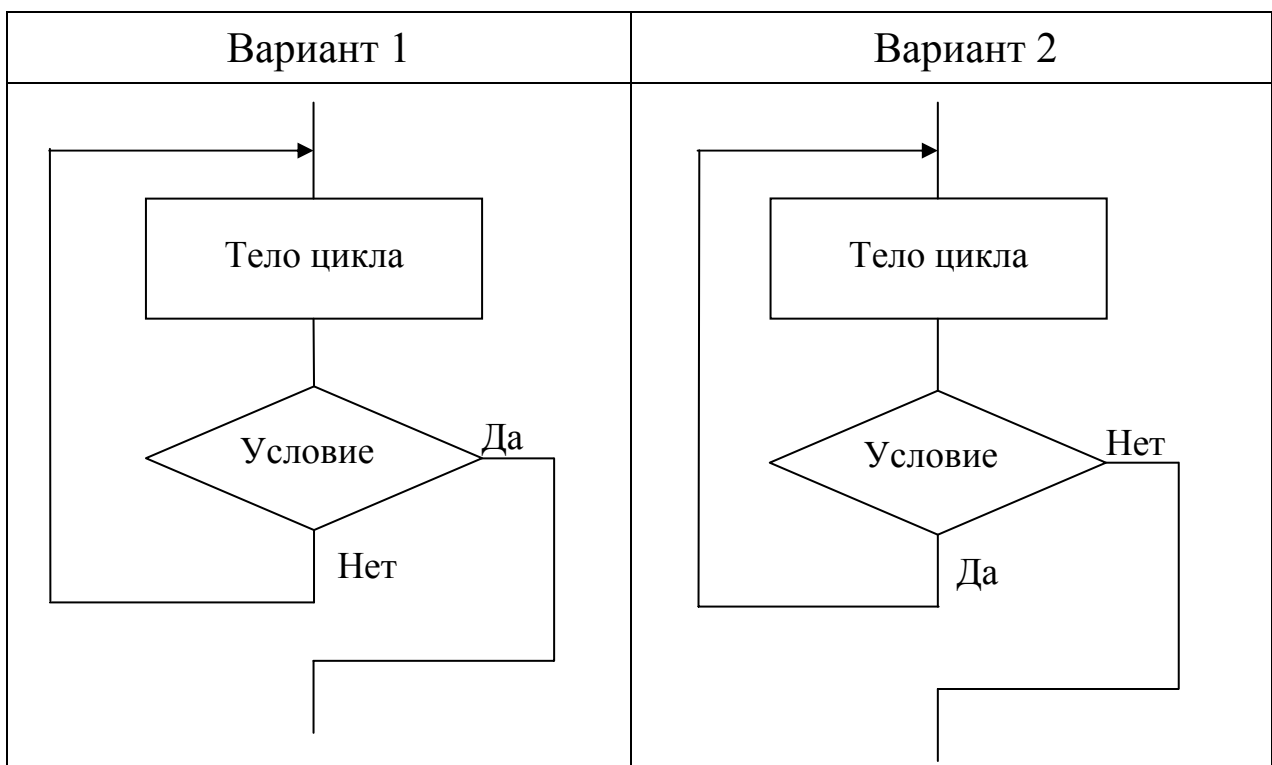


Рис. 5.4. Цикл с постусловием

Конструкция цикла с предусловием (рис. 5.3) в зависимости от результата выполнения условия может быть двух вариантов. В первом варианте повторение осуществляется до тех пор, пока условие

имеет значение *Истина* (True). В другом варианте повторение осуществляется до тех пор, пока условие имеет значение *Ложь* (False).

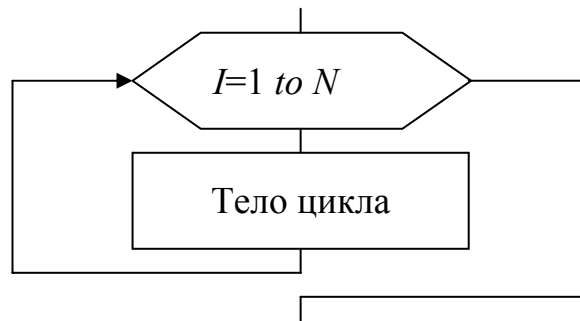


Рис. 5.5. Цикл по счетчику

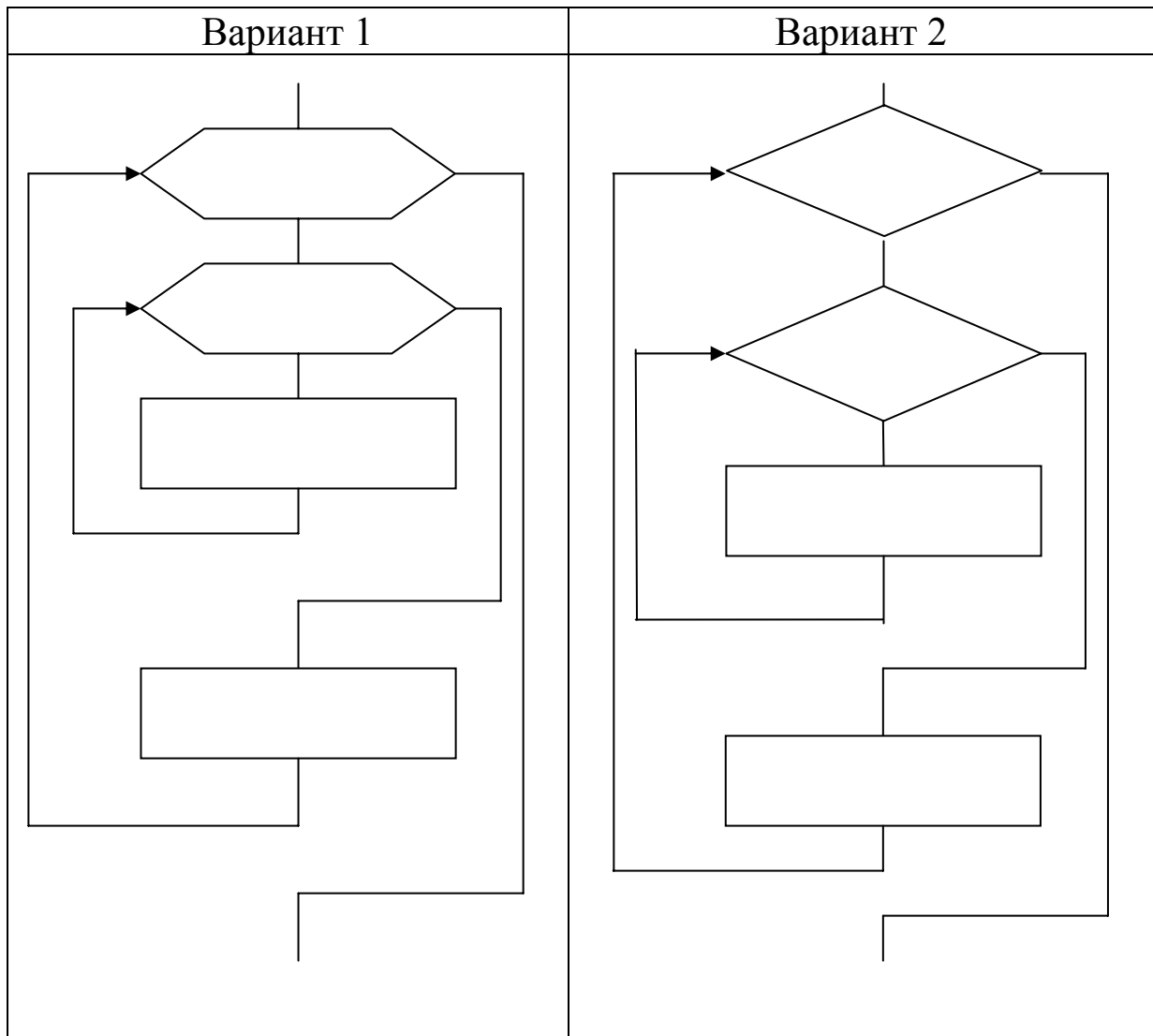


Рис. 5.6. Примеры вложенных циклов

Конструкция цикла с постусловием (рис. 5.4) в зависимости от результата выполнения условия может быть двух вариантов. В первом варианте повторение осуществляется до тех пор, пока условие

имеет значение *Ложь* (False). В другом варианте повторение осуществляется до тех пор, пока условие имеет значение *Истина* (True).

В цикле по счетчику (рис. 5.5) выполнение и повторение операций происходит от начального значения параметра (счетчика) до его конечного значения с указанным шагом. Если шаг не указан, то его значение полагается равным единице.

Простые циклические конструкции могут вкладываться в другую простую циклическую конструкцию, образуя тем самым *вложенный* (сложный) цикл (рис. 5.6). При этом необходимо придерживаться следующих правил:

- имена параметров всех простых циклов не должны повторяться;
- нельзя войти во внутренний (вложенный) цикл, минуя внешний;
- простые циклы в сложном цикле не должны пересекаться, то есть внешний цикл должен заканчиваться после внутреннего и инструкции тела внешнего цикла не должны быть в теле внутреннего цикла.

5.1.3. Контрольные вопросы

- 1) Как называется конечная последовательность точно определенных действий?
- 2) Перечислите основные свойства алгоритмов.
- 3) В каком свойстве выражается точность и однозначность каждого предписания алгоритма?
- 4) Что характеризует свойство алгоритма результативность?
- 5) Как называется возможность расчленения задачи на ряд элементарных шагов?
- 6) Перечислите типовые структуры (модели) алгоритмов.
- 7) Как называется алгоритмическая конструкция, в соответствии с которой инструкции выполняются в порядке естественного их следования?
- 8) Как называется алгоритм, содержащий два или более разветвленных вычислительного процесса?
- 9) Каково название алгоритма, обеспечивающего неоднократное выполнение последовательности инструкций?
- 10) Назовите способ, не применяемый для описания алгоритмов.

- 11) В виде какой фигуры изображается оператор вывода на блок-схеме?
- 12) В виде какой фигуры изображается оператор ветвления на блок-схеме?
- 13) Что можно вписать в блок-схеме внутри графической фигуры, похожей на овал?
- 14) Как называется и для чего в блок-схеме используется графическая фигура ромб?
- 15) Определите начальное значение x при выполнении алгоритма ($y = x + 6: x = y: y = x + y$), значение переменной $y = 14$.
- 16) Определите начальное значение x при выполнении алгоритма ($y = x + 2: x = 2 \times y: y = x + y$), значение переменной $y = 12$.
- 17) Какие значения получают c и d в результате выполнения алгоритма ($a = 8: b = 3: \text{если } a < b, \text{ то } c = b - a, \text{ иначе } c = 2 \times (a - b) : d = 0: \text{пока } c > a \text{ выполнить действия: } d = d + 1, c = c - 1$)?
- 18) Определите количество повторений цикла в фрагменте алгоритма ($a = 3: b = 7: \text{пока } a / b \leq b / 3: a = a + 2: b = b + 3$).
- 19) Какие значения получают элементы массива в результате выполнения алгоритма ($i = 2: N = 8: \text{пока } i \leq N: A(i) = i \times i: i = i + 2$)?
- 20) Какие значения получают элементы массива в результате выполнения алгоритма ($i = 1: N = 4: \text{пока } i \leq N: A(i) = i \times 2 + 2: i = i + 1$)?
- 21) Что представляет собой R , полученное в результате выполнения алгоритма ($R = 1: N = 10: i = 2: \text{пока } i \leq N: \text{если } x(i) < 0 \text{ тогда } R = R \times x(i): i = i + 2: \text{вывод } R$), по отношению к элементам заданного массива: $x(1), x(2), \dots, x(n)$?
- 22) Что представляет собой K , полученное в результате выполнения алгоритма ($K = 0: N = 10: i = 1: \text{пока } i \leq N: \text{если } x(i) = 0 \text{ тогда } K = K + 1: i = i + 1: \text{вывод } K$), по отношению к элементам заданного массива: $x(1), x(2), \dots, x(n)$?
- 23) Что представляет собой S , полученное в результате выполнения алгоритма ($S = 0: N = 10: i = 1: \text{пока } i \leq N: \text{если } x(i) > 0 \text{ тогда } S = S + x(i): i = i + 1: \text{вывод } S$), по отношению к элементам заданного массива: $x(1), x(2), \dots, x(n)$?
- 24) Что необходимо изменить, чтобы бесконечный цикл, содержащийся во фрагменте алгоритма ($x = 10: \text{пока } x \geq 10: x = x + 1$), стал конечным?
- 25) Какое значение будет выведено в результате выполнения алгоритма ($N = 0: y = 0: \text{пока } N < 10: y = y + N: N = N + 1: \text{вывод } y$)?

5.2. Программное обеспечение и технологии программирования

5.2.1. Средства создания программ

Любая система программирования, предназначенная для разработки программ на выбранном языке, включает в себя следующие компоненты.

- *Текстовый редактор.* Формировать текст программы можно в любом редакторе. Однако лучше использовать специализированные редакторы, которые ориентированы на конкретный язык программирования и позволяют в процессе ввода текста выделять ключевые слова и идентификаторы разными цветами и шрифтами. Подобные редакторы созданы для всех популярных языков и дополнительно могут автоматически проверять правильность синтаксиса программы непосредственно во время ввода ее кода.
- *Компилятор.* Последовательность шагов работы любого компилятора:
 - 1) разбор исходной программы и перевод ее во внутреннее представление, удобное для дальнейшей работы;
 - 2) перевод программы на промежуточный язык, не зависящий от системы команд конкретного компьютера;
 - 3) оптимизация кода программы;
 - 4) генерация выходного кода на машинном языке.

С помощью компилятора исходный текст (исходный модуль) программы переводится в машинный код (объектный модуль). Если в исходном тексте будут обнаружены синтаксические ошибки, то результирующий модуль не будет создан. Для небольшого исходного модуля на этом этапе уже возможно получение готовой программы. Однако исходный текст большой программы, как правило, состоит из нескольких модулей (файлов с исходными текстами), так как все исходные тексты хранить в одном файле неудобно, потому что в них трудно ориентироваться. Поэтому каждый модуль компилируется в отдельный файл с объектным кодом (двоичный файл со стандартным расширением имени `.OBJ`). Такие файлы затем необходимо объединять в одно целое. Кроме того, к ним нужно добавить машинный код подпрограмм, реализующих стандартные функции. Такие функции содержатся в библиоте-

ках (файлах со стандартным расширением имени `.LIB`), которые поставляются вместе с компилятором. Код модулей и подключенные к нему стандартные функции необходимо объединить с учетом требований операционной системы, то есть получить программу, отвечающую определенному формату.

- *Редактор связей (сборщик)*. Эта программа выполняет связывание объектных модулей и машинного кода стандартных функций, находя их в библиотеках. Она формирует на выходе работоспособное приложение – исполнимый код для конкретной платформы. Если по каким-то причинам один из объектных модулей или нужная библиотека не обнаружены, то сборщик сообщает об ошибке и тогда готовой программы не создается. Итоговый файл имеет в имени расширение `.EXE` или `.COM`.

Совокупность всех изложенных выше компонентов представляет собой интегрированную систему программирования. В ней почти все этапы создания программы автоматизированы. Таким образом, после ввода исходного текста программы его компиляция и сборка выполняются одним щелчком мыши или нажатием соответствующей клавиши.

5.2.2. Алгоритмическое программирование

Алгоритм – это формальное описание способа решения задачи путем разбиения ее на конечную по времени последовательность действий (элементарных операций). Такое формальное описание должно быть абсолютно полным и учитывать все возможные ситуации, которые могут встретиться по ходу решения задачи. Элементарные операции – это действия, которые выполняются по заранее определенным критериям и не требуют детализации.

Основная идея алгоритмического программирования – это разбиение программы на последовательность модулей, каждый из которых выполняет одно или несколько действий. При этом выполнение каждого модуля всегда начинается с первой команды и всегда заканчивается на самой последней.

Алгоритм на выбранном языке программирования записывается с помощью команд описания данных, вычисления значений и управления последовательностью выполнения программы. При этом текст программы представляет собой линейную последова-

тельность операторов присваивания, циклов и условных операторов. Таким способом можно решать не очень сложные задачи и составлять программы, содержащие несколько сот строк кода.

Для больших программ такая технология программирования мало приемлема из-за большого количества возникающих проблем. К ним относятся следующие.

- Резкое снижение понятности исходного текста из-за того, что общая структура алгоритма теряется за конкретными операторами языка, выполняющими слишком детальные, элементарные действия.
- Возникают многочисленные вложенные условные операторы и операторы циклов, логика становится запутанной. При попытке исправить один ошибочный оператор вносится несколько новых ошибок, связанных с особенностями работы этого оператора, результаты выполнения которого могут учитываться в самых разных местах программы.
- Набрать и отладить длинную (более 1000 строк кода) линейную последовательность операторов крайне затруднительно или практически невозможно.

5.2.3. Структурное программирование

Структурное программирование используется при создании средних по размеру приложений (несколько тысяч строк исходного текста). Идея структурного программирования заключается в том, что структура программы должна отражать структуру решаемой задачи так, чтобы алгоритм решения был ясно виден из исходного текста. С этой целью в программирование введено понятие подпрограммы – набора операторов, выполняющих нужное действие и не зависящих от других частей исходного кода. При использовании подпрограмм программа разбивается на мелкие подпрограммы (до 50 операторов – критический порог для быстрого понимания цели подпрограммы). Каждая такая подпрограмма выполняет одно из действий, предусмотренных исходным заданием. Комбинируя эти подпрограммы, удается формировать итоговый алгоритм уже не из простых операторов, а из законченных блоков кода, имеющих определенную смысловую нагрузку. При этом обращаться к таким блокам можно по их названиям (именам).

Наличие подпрограмм позволяет вести проектирование и раз-

работку приложения сверху вниз – *нисходящим проектированием*. Технология нисходящего проектирования предусматривает вначале выделение нескольких подпрограмм, решающих самые глобальные задачи, затем каждая из этих подпрограмм детализируется путем ее разделения на небольшое число других подпрограмм. Этот процесс продолжается до тех пор, пока вся задача не будет реализована.

Такой подход позволяет постоянно мыслить на предметном уровне, не опускаясь до конкретных операторов и переменных. При этом имеется возможность не реализовывать сразу некоторые подпрограммы, временно их откладывать, пока не будут реализованы другие части. Преимуществом данной технологии является также простота отладки небольших подпрограмм, что существенно повышает общую надежность всей программы. Готовые подпрограммы можно использовать повторно, если такая возможность имеется в решаемой задаче. Кроме того, вместе с интегрированными системами программирования поставляются большие библиотеки стандартных подпрограмм, которые позволяют значительно повысить производительность труда программиста за счет чужой работы по созданию часто используемых подпрограмм.

С появлением визуальных RAD-сред в операционной системе MS Windows, идеология которой основана на событиях, широкую популярность приобрел событийный подход к созданию программ – *событийно-ориентированное программирование*. Эта технология появилась в результате развития идей нисходящего проектирования, когда постепенно определяются и детализируются реакции программы на различные события. События могут быть *пользовательскими*, возникающими в результате действий пользователя, *системными*, возникающими в операционной системе, и *программными*, генерируемыми самой программой.

Предельный объем приложений, создаваемых в разумные сроки (несколько месяцев) с помощью структурного и событийного программирования составляет сотни тысяч строк программного кода.

5.2.4. Объектно-ориентированное программирование

Реальные объекты окружающего мира обладают тремя базовыми характеристиками:

- набором определенных свойств;
- способны разными методами изменять эти свойства;

- умеют реагировать на события, возникающие как вне объекта, так и внутри его.

На этих базовых характеристиках и основано понятие объекта в языках программирования.

Объект – это совокупность свойств (структур данных, характерных для этого объекта), методов их обработки (подпрограмм изменения свойств) и событий, на которые данный объект может реагировать и которые приводят, как правило, к изменению свойств объекта.

Суть парадигмы объектно-ориентированного программирования (ООП) состоит в том, что не программы управляют данными, а данные, связанные друг с другом и описывающие какой-либо объект (реальный или воображаемый), на базе которого реализована модель того или иного явления, управляют программами.

Примечание: парадигма (греч. *paradigma*) – это пример, образец, тип, модель. В нашем случае это модель.

В основе технологии ООП лежат три основных механизма:

- 1) инкапсуляция;
- 2) наследование;
- 3) полиморфизм.

Объекты могут иметь идентичную структуру и отличаться только значениями свойств. В таких случаях в программе создается новый тип, основанный на единой структуре объекта (по аналогии с тем, как создаются новые типы для структур данных). Такой тип называется *классом*, а каждый конкретный объект, имеющий структуру этого класса, называется *экземпляром класса*. Класс содержит данные и функции, с помощью которых обрабатываются эти данные. Данные называются свойствами класса, а функции – методами класса. Методы воздействуют на объект. При работе с объектами возникают события. При появлении события система посылает объекту сообщение, которое может быть обработано методом, создаваемым при разработке класса. Такие методы называют обработчиками событий или процедурами обработки события. События могут возникать как в результате действий пользователя, так и в результате работы системы.

Объединение данных с методами в одном классе (типе) называется *инкапсуляцией*. Это механизм, связывающий вместе код и

данные, которыми он манипулирует и одновременно защищает их от внешних помех и некорректного использования.

Важнейшей характеристикой класса является возможность создания на его основе новых классов, сохраняющих все свойства и методы исходного класса и добавляющих собственные свойства и методы. Эта характеристика называется *наследованием*. Наследование позволяет создавать новые классы, повторно используя уже готовый исходный код и не тратя времени на его переписывание. Класс, не имеющий предшественников, называется базовым. В большинстве случаев методы базового класса приходится переопределять для классов-наследников, так как методы для одних наследников могут выполняться совсем не так, как для других. Свойство объектов переопределять методы наследуемого класса называется *полиморфизмом*.

Технологии объектного, событийного и структурного программирования применены и объединены в визуальных RAD-системах. Эти системы содержат множество готовых классов, представленных в виде визуальных компонентов. Эти компоненты могут добавляться в программу одним щелчком мыши.

В рамки объектно-ориентированного подхода удачно вписывается концепция визуального программирования. Все визуальные составляющие, такие как формы, элементы управления, меню и панели инструментов, являются объектами со своими свойствами и методами. Эти объекты реагируют на определенные события. При визуальном подходе к разработке форм не требуется программного задания их основных свойств. Эти свойства можно задать при помощи мыши или установить их в окне свойств. Таким образом, визуальное программирование делает проектирование интерфейса программы намного более наглядным и быстрым.

Появление технологии ООП (середина 80-х годов XX века) позволило создавать в разумные сроки приложения размером до нескольких миллионов строк программного кода.

5.2.5. Контрольные вопросы

- 1) Назовите средства создания программ.
- 2) Последовательно перечислите этапы решения задач на ЭВМ.

- 3) Как называется технология программирования, основанная на модульной структуре программного продукта и типовых управляющих структурах алгоритмов?
- 4) Назовите основные понятия объектно-ориентированного подхода к программированию.
- 5) Что не изменяется в процессе написания программы?
- 6) Как называется алгоритм, записанный на языке программирования?
- 7) Как называется программа, записанная способом, допускающим ее непосредственное выполнение на ЭВМ?
- 8) Как называется предварительно разработанный и синтаксически выделенный блок операторов?
- 9) Что не свойственно подпрограммам?

5.3. Языки программирования

Языки программирования – это искусственные языки, которые отличаются от естественных ограниченным числом «слов», понятных транслятору и очень строгими правилами записи команд (операторов). Совокупность подобных требований образует синтаксис языка программирования, а смысл каждой команды и других конструкций языка – его семантику. При этом под прагматикой языка программирования понимают соответствие его возможностей и средств достижению поставленной цели (решаемой задачи). Нарушение формы записи программы приводит к тому, что транслятор не может понять назначение оператора и выдает сообщение о синтаксической ошибке. Правильно написанные, но не отвечающие алгоритму команды языка приводят к семантическим (логическим) ошибкам, которые называются ошибками времени выполнения. Процесс поиска ошибок в программе называется *тестированием*, а процесс устранения ошибок – *отладкой*.

5.3.1. Языки программирования низкого уровня

Языками программирования низкого уровня называют языки, которые ориентированы на конкретный тип процессора и учитывают его особенности. В таких языках операторы близки к машинному коду и ориентированы на конкретные команды процессора.

С помощью языков низкого уровня создаются очень эффективные и компактные программы. К их числу относятся небольшие системные приложения, драйверы устройств, модули стыковки с

нестандартным оборудованием и тому подобное. То есть низкоуровневые языки программирования используются в тех случаях, когда важнейшими требованиями становятся компактность, быстрое действие и возможность прямого доступа к аппаратным ресурсам. К языкам низкого уровня относятся различного вида *ассемблеры*.

5.3.2. Высокоуровневые языки программирования

Языками программирования высокого уровня называют языки, которые ориентированы на человека, более близки и понятны ему. В таких языках не учитываются особенности компьютерных архитектур. Создаваемые программы на уровне исходных текстов средствами этих языков легко переносимы на другие платформы, имеющие транслятор соответствующего языка. Создание языков высокого уровня по времени совпадает с появлением языков программирования третьего поколения (60-е годы XX века). Это универсальные языки, с их помощью удается решать задачи из любых областей. Относительная простота, независимость от конкретного компьютера и возможность использования мощных синтаксических конструкций позволили резко повысить производительность труда программистов. К числу языков этого поколения относится Basic (Бейсик). Для этого языка имеются и компиляторы, и интерпретаторы, а по популярности он занимает первое место в мире. Этот язык очень прост в изучении.

С начала 70-х годов XX века по настоящее время продолжается период языков четвертого поколения. Эти языки предназначены для реализации крупных проектов, повышения их надежности и скорости создания. Как правило, в эти языки встроены мощные операторы, позволяющие одной строкой описать такую функциональность, для реализации которой на языках младших поколений потребовались бы тысячи строк исходного кода. К числу языков этого поколения относятся: Pascal (Паскаль), C (Си), C++ (Си++), Java (Джава, Ява).

Паскаль во многом напоминает Алгол (язык 3-го поколения), но в нем ужесточен ряд требований к структуре программы и имеются возможности, позволяющие успешно применять его при создании крупных проектов.

Язык Си планировался для замены ассемблера, чтобы иметь возможность создавать столь же эффективные и компактные программы,

и в то же время не зависеть от конкретного типа процессора. Этот язык во многом похож на Паскаль, однако имеет дополнительные средства (указатели) для прямой работы с памятью.

Си++ – это объектно-ориентированное расширение языка Си. Он имеет множество мощных возможностей, позволяющих резко повысить производительность труда программистов. Однако этот язык требует от программистов высокого уровня профессиональной подготовки.

Язык Ява был создан в начале 90-х годов XX века компанией Sun на основе Си++. В этом языке исключены все низкоуровневые возможности языка Си++. Главной особенностью языка Ява является то, что компиляция происходит не в машинный код, а в платформу-независимый байт-код (каждая команда занимает один байт). Этот байт-код может выполняться с помощью интерпретатора виртуальной Ява-машины (Java Virtual Machine), версии которой сегодня созданы для любых платформ. Благодаря этому программы на Ява можно переносить не только на уровне исходных текстов, но и на уровне двоичного байт-кода. Поэтому по популярности этот язык сегодня занимает второе место в мире после Бейсика. Особое внимание в развитии языка Ява уделяется двум направлениям:

- 1) поддержке мобильных устройств и микрокомпьютеров, встраиваемых в бытовую технику (технология Jini);
- 2) созданию платформу-независимых программных модулей, способных работать на серверах в глобальных и локальных компьютерных сетях с различными операционными системами (технология Java Beans).

В середине 90-х годов XX века появились языки пятого поколения. Это языки объектно-ориентированного программирования. К ним относятся системы автоматического создания прикладных программ с помощью визуальных средств разработки. В этом случае становится необязательным знание основ программирования. Главная идея, которая заложена в основу этих языков, – возможность автоматического формирования результирующего текста на универсальных языках программирования. Этот результирующий текст программы потом необходимо откомпилировать. Инструкции в компьютер вводятся в максимально наглядном виде с помощью методов, наиболее удобных для человека, не знакомого с программированием. Из этого поколения наиболее популярны

следующие визуальные системы (среды) быстрого проектирования программ для MS Windows:

- Visual Basic for Applications (VBA) на базе языка программирования Бейсик;
- Borland Delphi (Делфи) на базе языка программирования Паскаль;
- Borland C++ Builder на базе языка программирования C++;
- Symantec Cafe на базе языка программирования Ява.

Для разработки серверных и распределенных приложений создана система программирования Microsoft Visual C++.

5.3.3. Программирование баз данных

Группа языков программирования баз данных отличается от алгоритмических языков решаемыми задачами. База данных – это файл (или группа файлов), представляющий собой упорядоченный набор записей, имеющих единообразную структуру и организованных по единому шаблону (как правило, в табличном виде). При работе с базами данных чаще всего требуется выполнять следующие операции:

- создание, модификация свойств, удаление таблиц в базе данных;
- поиск, отбор, сортировка информации по запросам пользователей;
- добавление новых записей;
- модификация, удаление существующих записей.

Все эти операции возможны и осуществляются с помощью системы управления базами данных (СУБД). Основной поддержкой любой СУБД является структурированный язык запросов SQL (Structured Query Language). Он основан на мощной математической теории и позволяет выполнять эффективную обработку баз данных, манипулируя не отдельными записями, а их группами.

Помимо поддержки языка SQL практически в каждой СУБД имеется также свой уникальный встроенный язык, ориентированный на особенности данной СУБД и не переносимый в другие системы. Ведущими производителями СУБД являются:

- Microsoft (встроенный язык SQL Server);
- IBM (встроенный язык DB2);

- Oracle (встроенный язык PL/SQL);
- Adabas (встроенный язык Natural);
- Informix (встроенный язык INFORMIX 4GL).

Для программирования персональных компьютеров в настоящее время используются языки Visual FoxPro фирмы Microsoft и Visual dBase фирмы Inprise.

При создании программ и формировании структур баз данных нередко применяются формальные способы их представления – формальные нотации. С помощью формальных нотаций можно визуально представить (изобразить) таблицы баз данных, поля, объекты программы и взаимосвязи между ними в системе, имеющей специализированный редактор и генератор исходных текстов программ на основе созданной модели. Такие системы называют CASE-системами (CASE – Computer-Aided Software Engineering). В них активно применяются нотации IDEF (Integrated DEfinition) и популярный язык графического моделирования UML (Unified Modeling Language).

5.3.4. Программирование для сети Интернет

Языки программирования для интернета называют скрипт-языками.

Для оформления гипертекстовых документов (веб-документов) предназначен язык HTML (HyperText Markup Language). Он очень прост и содержит элементарные команды (теги или метки) формирования текста, добавления рисунков, задания шрифтов и цветов, организации ссылок и таблиц.

Для эффективной обработки больших текстовых файлов, генерации текстовых отчетов и управления задачами предназначен язык Perl. Он располагает большим количеством часто используемых функций, предназначенных для работы со строками, массивами, всевозможными средствами преобразования данных, управления процессами, работы с системной информацией и др. По мощности этот язык значительно превосходит языки типа Си.

Для автоматизации рутинных процессов предназначен язык Tcl/Tk (Tool Command Language/Tool Kit), который имеет в своем арсенале мощные команды, предназначенные для работы с абстрактными нетипизированными объектами. Язык независим от

типа системы и позволяет создавать программы с графическим интерфейсом.

Для организации виртуальных трехмерных интерфейсов в интернете предназначен язык VRML (Virtual Reality Modeling Language). Он позволяет описывать в текстовом виде различные трехмерные сцены, освещение и тени, текстуры (покрытия объектов), создавать свои миры, путешествовать по ним, вращать в любых направлениях, масштабировать, регулировать освещенность и т. п.

5.3.5. Контрольные вопросы

- 1) Что входит в состав алфавита языка программирования?
- 2) Перечислите и охарактеризуйте известные вам типы выражений.
- 3) Как называется именованная область памяти, в которой хранится какое-либо значение?
- 4) Какие из перечисленных систем не относятся к системам программирования: Adobe PhotoShop, Visual C++, Borland Delphi, MS DOS, Java?
- 5) Какие из перечисленных ниже языков не относятся к языкам высокого уровня: ADA, АССЕМБЛЕР, LISP, PASCAL, МАКРОАССЕМБЛЕР?
- 6) Как называется совокупность фиксированного числа одноименных и одноптипных переменных?

5.4. Язык программирования Visual Basic for Applications

В середине 80-х годов в программировании возникло новое направление, основанное на понятии объект. Реальные объекты окружающего мира обладают тремя базовыми характеристиками: они имеют набор свойств, способны разными методами изменять эти свойства и реагировать на события, возникающие как в окружающем мире, так и внутри самого объекта. Именно в таком виде в языке программирования Visual Basic for Applications (VBA) и реализовано понятие объекта. Здесь объект рассматривается как совокупность свойств (структур данных, характерных для этого объекта), методов их обработки и событий, на которые данный объект может реагировать и которые приводят, как правило, к изменению свойств объекта.

Объекты могут иметь идентичную структуру и отличаться только значениями свойств. В таких случаях в программе создается

новый тип, основанный на единой структуре объекта. Он называется классом, а каждый конкретный объект, имеющий структуру этого класса, называется экземпляром класса.

5.4.1. Алфавит VBA

Как и все языки программирования высокого уровня, Visual Basic имеет свой набор допустимых для использования символов – алфавит. Алфавит языка содержит в себе:

- прописные и строчные буквы латинского алфавита:
A, B, ..., Z;
a, b, ..., z;
- прописные и строчные буквы русского алфавита:
А, Б, ..., Я;
а, б, ..., я;
- арабские цифры: 0, 1, ..., 9.

Для построения конструкций языка используются также нижеперечисленные специальные символы:

- + сложение;
- - вычитание;
- * умножение;
- / деление;
- \ целочисленное деление;
- ^ возведение в степень;
- () для составления сложных выражений;
- = оператор присваивания или операция отношения «равно», используемая в логических выражениях;
- & сцепление строк (конкатенация);
- ' комментарий;
- _ разрыв строки исходного кода программы или как связка в идентификаторах;
- > больше;
- < меньше;
- <= меньше или равно (не больше);
- >= больше или равно (не меньше);

- $\langle \rangle$ не равно;
- **пробел** разделитель слов (операторов) в языке.

Из вышеописанных символов, относящихся к алфавиту языка, конструируются все его слова (предложения). К их числу относятся имена типов, встроенных констант, процедур, функций, операторов, объектов, свойств, методов и др. При этом прописные и строчные буквы не различаются. Все слова, задействованные в конструкциях языка, являются зарезервированными словами и не могут применяться для других целей.

5.4.2. Переменные и константы

В VBA, как и в других языках программирования высокого уровня, для хранения значений используются два вида данных: переменные и константы.

Переменная – именованная область памяти, используемая для хранения значения, которое можно изменить при выполнении программы. Переменные подразделяются на простые и индексированные (переменные с индексом). Индексированными переменными являются элементы массивов.

Константа – именованная область памяти, используемая для хранения фиксированного значения, которое невозможно изменить при выполнении программы.

Имена переменных и констант составляются на основании следующих правил:

- первым символом всегда должна быть буква или символ подчеркивания “_”;
- в составе имени нельзя использовать символы: **!, @, &, \$, # , пробел**;
- в качестве имени нельзя использовать ключевые (зарезервированные) слова, входящие в конструкции языка VBA;
- длина имени не может быть более 255 символов;
- имя нельзя повторять в пределах области его видимости (действия).

5.4.2.1. Область действия

Область действия (видимости) переменных и констант определяется с помощью ключевых слов `Private`, `Public` и `Static`.

`Private` – область действия в пределах конкретного модуля. После завершения выполнения модуля память, отведенная под эти переменные, освобождается.

`Public` – область действия в пределах приложения.

`Static` – область действия в пределах конкретного модуля и внешних процедур, используемых в данном модуле. После завершения выполнения модуля значения этих переменных сохраняются и могут быть использованы при повторном выполнении модуля.

Переменные и константы, в зависимости от области действия, подразделяются на *глобальные* и *локальные*.

Если переменная или константа описана внутри процедуры, то она является *локальной*, то есть она определена и может использоваться только в пределах данной процедуры.

Если переменная или константа описана вне процедуры (в программе), то она будет *глобальной*. Такая переменная или константа может быть использована в других процедурах, внутренних по отношению к той, где она описана.

5.4.2.2. Объявление

Переменная объявляется (описывается) с помощью ключевых слов `Private`, `Public`, `Static`, `Dim`. Чтобы явно указать тип переменной, используется ключевое слово `As`.

Примеры описания простых переменных:

```
Private X
```

```
Public i As Integer, r As Long, c As Date
```

```
Static Строка As String
```

```
Dim Y
```

```
Dim Z As Single 'Явный способ объявления  
переменной. Самый простой и надежный
```

Примеры описания индексированных переменных:

```
Dim Массив1() As Integer
```

```
Dim Массив2(12) As String
```

```
Dim Массив3(1 to 20) As Single
```

```
Dim Массив4(1 to 5, 1 to 7) As Byte
```

Константа объявляется с помощью ключевого слова `Const`. При этом можно указать ее тип, область действия и присвоить ей значение.

Синтаксис объявления:

```
Const <имя константы> As <тип> = <значение>
```

или

```
Const <имя константы> = <значение>
```

Если в константе явно не указан тип данных, то VBA назначает ей тип, который соответствует присваиваемому значению.

Примеры:

```
Private Const q = 44,55
```

```
Public Const pi = 3,1459
```

```
Static Const QWER=2,54
```

```
Const y = 34
```

```
Const Con As Byte = 34
```

```
Const z As Single = -3,8374E-22
```

Все строковые константы указываются в кавычках.

Примеры:

```
Const prv As String = "Язык программирования VBA"
```

```
Public Const prv = "Язык программирования VBA"
```

5.4.2.3. Время жизни

Переменные и константы, объявленные как `Private`, сохраняют свое значение только во время выполнения блока кода, в котором они определены.

Переменные и константы, объявленные как `Public`, сохраняют свое значение и вне блока кода, в котором они определены, то есть до конца выполнения программы.

Переменные и константы, объявленные как `Static`, сохраняют свое значение и между вызовами процедур.

Переменные и константы, объявленные без ключевых слов `Public`, `Private`, `Static`, сохраняют свое значение согласно месту их объявления (описания).

5.4.2.4. Типы переменных и констант

Таблица 5.1

| Тип данных | Описание | Диапазон |
|------------------------|--|--|
| Byte | целое число | От 0 до 255 |
| Integer | целое число | От -32768 до 32767 |
| Long | длинное целое число | От -2147483648 до 2147483647 |
| Single | число с плавающей запятой одинарной точности | а) для отрицательных чисел: от -3,402823E38 до -1,401298E-45 б) для положительных чисел: от 1,401298E-45 до 3,402823E38 |
| Double | число с плавающей запятой двойной точности | а) для отрицательных чисел: от -1,79769313486231E308 до -4,94065645841247E-324 б) для положительных чисел: от 4,94065645841247E-324 до 1,79769313486231E308 |
| Currency (денежный) | число с фиксированной десятичной точкой | от -922.337.203.685.477.5808 до 922.337.203.685.477.5807 |
| String | строка символов | от 0 до 147483647 символов |
| Variant | универсальный | Значения любого типа |
| Boolean | логический | True или False |
| Date | дата | от 1.01.100 до 31.12.9999 |
| Object | объект | Ссылка на любой объект |

Примечание: для дробных чисел существует экспоненциальная форма записи, например, $1,456 \cdot 10^{23}$. В VBA число, записанное в этой форме, будет выглядеть так: 1,456E23, где E обозначает основание 10, а после E указывается степень числа. Запись этого же числа в отрицательной степени будет выглядеть так: 1,456E-23.

5.4.3. Управляющие конструкции

5.4.3.1. Операнды, операции, выражения

Операнды – это данные, над которыми выполняются какие-либо операции. В качестве операндов могут быть использованы константы, переменные, функции.

Операция – это элементарное действие над операндами.

Выражение – это последовательность операндов, отделенных друг от друга знаками операций. В выражении возможно присутствие парных круглых скобок. В зависимости от типа операндов и используемых операций выражения делятся на *арифметические, логические и строковые (текстовые)* выражения.

Арифметическим называется выражение, содержащее операнды только числового (вещественного и/или целого) типа (числа) и знаки арифметических операций (табл. 5.2).

Таблица 5.2

| Знак | Операция | Пример | Результат |
|------|------------------------------|----------|-----------|
| - | Знак числа (смена знака) | -6 | -6 |
| ^ | Возведение в степень | 3^2 | 9 |
| * | Умножение | 4*8 | 32 |
| / | Деление | 10/3 | 3, (3) |
| \ | Целочисленное деление | 10\3 | 3 |
| mod | Остаток от деления по модулю | 10 Mod 3 | 1 |
| + | Сложение | 3+2 | 5 |
| - | Вычитание | 7-5 | 2 |

Примеры: $Y = X*10$, $F = (x + y) / (3 + x^2)$, $G = \text{Sin}(x)$.

Выражение, содержащее хотя бы один знак операции отношения (табл. 5.3) или логической операции (табл. 5.4), называется *логическим*. Результатом выполнения такого выражения является логическая константа со значением либо True (истина), либо False (ложь). Логические выражения получаются в результате использования операций отношения (сравнения) либо логических операций. Операнды операций отношения должны быть одного типа (сравнимы между собой).

Таблица 5.3

| Знак | Операция | Примеры (при значениях $a = 6$, $b = 3$) | Результат |
|------|------------------|--|-----------|
| < | Меньше | $b < 7$ | True |
| > | Больше | "Вася" > "Маша" | False |
| <= | Меньше или равно | $a + b <= 12$ | True |
| >= | Больше или равно | $b^2 >= 9$ | True |
| <> | Не равно | $a <> b$ | True |
| = | Равно | $B = 10 - a$ | False |

Таблица 5.4

| Знак | Операция | Пример | Значение А | Значение В | Результат |
|------|---------------------------|---------|--------------------------------|--------------------------------|---------------------------------|
| not | Логическое отрицание (НЕ) | not A | True False | | False True |
| and | Логическое умножение (И) | A and B | True True False False | True False True False | True False False False |
| or | Логическое сложение (ИЛИ) | A or B | True True False False | True False True False | True True True False |
| xor | Исключающее ИЛИ | A xor B | True True False False | True False True False | False True True False |

Таблица 5.5

| Знак | Пример | Результат |
|------|--------------|-----------|
| & | "Крас"&"ное" | "Красное" |
| + | "Чер"+"ное" | "Черное" |

Строковое (текстовое) выражение (табл. 5.5) может содержать операнды только символьного (текстового или строкового) типа. В языках программирования, в том числе и в VBA, имеется

только одна строковая операция, которая носит название *конкатенация* или *слияние (сцепление)* строк. Эта операция обозначается знаками «&» (коммерческое И – амперсанд) или «+» (плюс).

5.4.3.2. Приоритет выполнения операций

Приоритет операций (табл. 5.6) определяет порядок их выполнения в выражениях. Выражения вычисляются слева направо в порядке приоритета операций. Если в выражении имеются парные круглые скобки, тогда в первую очередь вычисляется выражение, расположенное в самых внутренних скобках.

Таблица 5.6

| Приоритет | Операция |
|-----------|--|
| 1 | Вызов функций и выражения в скобках |
| 2 | ^ (возведение в степень) |
| 3 | - (смена знака числа) |
| 4 | * (умножение), / (деление), \ (целочисленное деление), mod (деление по модулю) |
| 5 | +, - и & (сложение, вычитание и конкатенация) |
| 6 | <, <=, =, <>, >, >= |
| 7 | not |
| 8 | and |
| 9 | or |
| 10 | xor |

5.4.3.3. Встроенные функции

В VBA имеется большой набор встроенных функций, использование которых существенно упрощает программирование. Эти функции можно разделить на следующие основные категории:

- математические функции (табл. 5.7);
- функции проверки типов (табл. 5.8);
- функции преобразования типов (табл. 5.9);
- функции обработки строк (табл. 5.10);
- функции времени и даты (табл. 5.11);
- функции выбора (табл. 5.12).

Таблица 5.7

| Имя функции | Математическая интерпретация |
|--------------------|--|
| Abs (x) | x (модуль числа x) |
| Atn (x) | Arctg x (арктангенс x) |
| Cos (x) | Cos x (косинус x) |
| Exp (x) | Экспонента (e в степени x) |
| Fix (x) | Отбрасывает дробную часть числа x |
| Int (x) | Округляет вещественное число x до целого в меньшую сторону |
| Log (x) | Ln x (натуральный логарифм x) |
| Rnd () | Генерирует случайное число от 0 до 1 |
| Rnd (x) | Генерирует случайное число от 0 до x |
| Sgn (x) | Sign x – знак числа (сигнум x) |
| Sin (x) | Sin x (синус x) |
| Sqr (x) | Корень квадратный числа x |
| Tan (x) | Tg x (тангенс x) |

Таблица 5.8

| Имя функции | Проверка |
|------------------------|---------------------------------------|
| IsArray (переменная) | Является ли переменная массивом? |
| IsDate (переменная) | Является ли переменная датой? |
| IsEmpty (переменная) | Инициализирована ли переменная? |
| IsError (переменная) | Является ли переменная кодом ошибки? |
| IsNull (переменная) | Является ли переменная пустой (Null)? |
| IsNumeric (переменная) | Является ли переменная числом? |
| IsObject (переменная) | Является ли переменная объектом? |

Таблица 5.9

| Имя функции | Тип, в который преобразуется выражение |
|--------------------|---|
| CBool (Выражение) | Boolean (логический) |
| CByte (Выражение) | Byte (байтовый) |
| CCur (Выражение) | Currency (денежный) |
| CDate (Выражение) | Date (дата) |
| CDbl (Выражение) | Double (число с плавающей запятой двойной точности) |

Продолжение табл. 5.9

| Имя функции | Тип, в который преобразуется выражение |
|--------------------|---|
| CInt (Выражение) | Integer (целое число) |
| CLng (Выражение) | Long (длинное целое число) |
| CSng (Выражение) | Single (число с плавающей запятой одинарной точности) |
| CStr (Выражение) | String (строка) |
| CVar (Выражение) | Variant (вариант) |

Таблица 5.10

| Имя функции | Описание |
|---|--|
| Mid (<строка>, <начало> [, <длина>]) | Возвращает из строки подстроку указанной длины, начиная с заданного символа. Если длина не указана, то возвращается вся подстрока, начиная от заданного символа |
| Left (<строка>, <длина>) | Возвращает из заданной строки подстроку указанной длины, начиная с левого края строки |
| Right (<строка>, <длина>) | Возвращает из заданной строки подстроку указанной длины, начиная с правого края строки |
| Len (<строка>) | Возвращает длину указанной строки |
| Lcase (<строка>) | Преобразует в заданной строке все прописные буквы в строчные |
| String (<количество>, <символ>) | Повторяет заданный символ указанное количество раз |
| InStr (<начало>, <исходная строка>, <искомая подстрока>, <тип сравнения>) | Ищет подстроку в заданной строке, начиная с указанного символа. Тип сравнения: 0 – с учетом регистра (vbBinaryCompare); 1 – без учета регистра (vbTextCompare) |
| Trim (<строка>) | Удаляет пробелы из начала и конца заданной строки |

Продолжение табл. 5.10

| Имя функции | Описание |
|----------------------|--|
| Ltrim(<строка>) | Удаляет все пробелы из начала заданной строки |
| Rtrim(<строка>) | Удаляет все пробелы из конца заданной строки |
| Space(<количество>) | Повторяет пробел указанное количество раз |
| Ucase(<строка>) | Преобразует в заданной строке все строчные буквы в прописные |
| StrReverse(<строка>) | Изменяет последовательность символов в заданной строке на обратную |
| Asc(<Символ>) | Возвращает код символа |
| Chr(<Код>) | Возвращает символ по его коду |
| Str(<Число>) | Преобразует заданное число в его строковое представление |
| Val(<Строка>) | Преобразует строку, представляющую собой число, в число |

Таблица 5.11

| Имя функции | Возвращаемое значение |
|----------------------------|---|
| Date | Возвращает значение типа Variant(Date), содержащее текущую системную дату |
| Time | Возвращает значение типа Variant(Date), содержащее текущее время по системным часам компьютера |
| Now | Возвращает значение типа Variant(Date), то есть текущую дату и время по системному календарю и часам компьютера |
| Hour, Minute, Second | Возвращает значение типа Variant(Date), содержащая целое число, представляющее часы, минуты и секунды в значении времени. <u>Пример:</u> BP=#4:35:17 PM# Час = Hour(BP) Минута = Minute(BP) Секунда = Second(BP) |

| Имя функции | Возвращаемое значение |
|------------------|--|
| Day, Month, Year | Возвращает значение типа Variant (Integer), содержащее целое число и представляющее день, месяц и год в значении даты. Синтаксис: Day(Дата), Month(Дата), Year(Дата) |

Таблица 5.12

| Имя функции | Возвращаемое значение |
|-------------|---|
| Iif | Возвращает одну из альтернатив. Синтаксис: Iif(expr, truepart, falsepart), где: expr – проверяемое значение; truepart – возвращаемое значение или выражение, если expr имеет значение true; falsepart – возвращаемое значение или выражение, если expr имеет значение false |
| Choose | Возвращает значение, выбранное из списка аргументов. Синтаксис: Choose(индекс, вариант_1, вариант_2, ..., вариант_n). Функцию Choose можно использовать для выбора одного из возможных значений, представленных в виде списка. |

5.4.3.4. Операторы альтернативы (ветвления)

Как и в любом другом языке программирования, в VBA можно проверять условия и выполнять действия в соответствии с результатами проверки этих условий. Для данной цели применяются следующие операторы (инструкции) принятия решения, позволяющие организовать в программе ветвление.

5.4.3.4.1. Условный оператор

IF <условие> THEN <оператор (код)>

Такая языковая конструкция позволяет выполнить один или несколько операторов в случае истинности проверяемого условия.

Применяется однострочный или блочный вариант записи условного оператора. Если необходимо выполнить более одной строки кода, нужно использовать блочный вариант с ключевым словом End IF. С помощью такой инструкции реализуется базовый алгоритм неполного ветвления.

Синтаксис:

```
IF <условие> Then <оператор (код)>
```

```
IF <условие> Then  
    <блок кода>
```

```
End IF
```

Примеры:

```
IF x<10 Then z=0
```

```
IF x>10 Then  
    z=2  
    z=z+x
```

```
End IF
```

Инструкция, реализующая базовый алгоритм полного ветвления, позволяет определить два блока операторов. Первый выполняется, когда условие истинно, а второй, когда оно ложно.

Пример:

```
IF x<>0 Then  
    y=Sin(x)/x  
Else  
    y=1
```

```
End IF
```

Пример ветвления по четырем направлениям:

```
IF <условие 1> Then  
    <блок кода 1>  
ElseIF <условие 2> Then  
    <блок кода 2>  
ElseIF <условие 3> Then  
    <блок кода 3>  
Else <блок кода 4>
```

```
End IF
```

В блоке IF допускается любое количество предложений ElseIF, но ни одно из них не может находиться после предложения Else. Однако с точки зрения методологии структурного

программирования уровень вложенности оператора IF не должен превышать трех.

Пример:

```
IF x=-1.57 Then
    y=-1
ElseIF x=0 Then
    y=0
ElseIF x=1.57 Then
    y=1
Else y=Sin(x)
End IF
```

5.4.3.4.2. Оператор выбора

При выборе для выполнения одного из нескольких операторов (блоков операторов) целесообразно и удобно использовать инструкцию Select Case. С помощью этого оператора в языке реализована алгоритмическая конструкция множественного выбора.

Синтаксис:

```
Select Case <переменная или выражение>
    Case <значение 1>
        <оператор (блок операторов) 1>
    Case <значение 2>
        <оператор (блок операторов) 2>
    Case <значение 3>
        <оператор (блок операторов) 3>
End Select
```

Пример использования оператора выбора варианта в подпрограмме-функции:

```
Function PR(ByVal S As Single,
            ByVal P As Integer)As Single
    Select Case P
        Case 0
            PR=S*0
        Case 1
            PR=S*0.10
        Case 2
            PR=S*0.15
        Case 3
            PR=S*0.20
```

```
End Select
End Function
```

Пример программы, вызывающей подпрограмму-функцию:

```
Sub Krb()
  Dim S As Single
  Dim Sum As Single
  Dim P As Integer
  Dim Prom As Variant
  S = 5000
  Do
    Prom=InputBox("Введите номер варианта <от 0 до 5>")
    IF Not IsNumeric(Prom) Then MsgBox("Повторите ввод!")
  Loop Until IsNumeric(Prom)
  P=Prom
  Sum=PR(S,P) `Вызов процедуры-функции. Оператор
               вызова PR(S,P)
  MsgBox("Значение S=") & CSng(S)
End Sub
```

Допускается вложенность инструкций Select Case. При этом каждой вложенной инструкции Select Case должна соответствовать инструкция End Select.

5.4.3.5. Операторы циклов

```
DO ... LOOP
```

С помощью такой инструкции в языке реализована базовая алгоритмическая конструкция повторения, и она позволяет многократно выполнить любой блок кода.

Существует несколько вариантов записи этого оператора, но в каждом из них проверяется условие, управляющее работой цикла, и по результатам проверки определяется необходимость его продолжения. Результатом вычисления логического выражения (управляющего условия) будет одна из констант True или False.

5.4.3.5.1. Циклы с предусловием

```
DO WHILE <условие> ... LOOP
```

Оператор DO WHILE <условие> ... LOOP позволяет проверить условие перед началом цикла и выполнять цикл до тех пор, пока оно имеет значение True. Как только условие, управляющее

работой цикла, примет значение False, выполнение тела цикла прекратится.

Пример:

```
Dim X As Integer 'Описание переменной X
                  целого типа
X=0 'Начальное значение переменной X
DO WHILE X<=10 'Пока X<=10, цикл будет
               повторяться
  X=X+1 'Изменение значения переменной X
LOOP 'Конец цикла
```

Другой вариант записи инструкции такого цикла:

```
WHILE <условие>... WEND
```

Пример:

```
X=0
WHILE X<12
  Y=Cos (X)
  X=X+1
WEND
```

```
DO UNTIL <условие> ... LOOP
```

Оператор Do Until <условие> ... Loop позволяет проверить условие перед началом цикла и выполнять тело цикла до тех пор, пока условие принимает значение False. Как только управляющее условие примет значение True, выполнение тела цикла прекратится.

Пример:

```
Dim X As Integer 'Описание переменной X
                  целого типа
X=0 'Начальное значение переменной X
Do Until X>10 'До тех пор, пока X<=10,
              цикл повторяется
  X=X+1 'Изменение значения переменной X
Loop 'Конец цикла
```

5.4.3.5.2. Циклы с постусловием

```
DO ... LOOP WHILE <условие>
```

Если операторы цикла необходимо выполнить хотя бы раз, то для этой цели нужно использовать цикл с постусловием. Инструкция Do ... Loop While <условие> позволяет проверить

условие после выполнения операторов тела цикла. Цикл будет повторяться до тех пор, пока выражение в условии цикла имеет значение True. Как только условие цикла примет значение False, выполнение тела цикла прекратится.

Пример:

```
Dim X As Integer
X=0
Do
  X=X+1
Loop While X<=10 'До тех пор, пока X<=10,
                 цикл повторяется
```

```
DO ... LOOP UNTIL <условие>
```

В отличие от предыдущего этот цикл будет выполняться до тех пор, пока значение управляющего условия равно False.

Пример:

```
Dim X As Integer
X=0
Do
  X=X+1
Loop Until X>10 'Как только переменная станет
                больше десяти, выполнение цикла прекратится
```

5.4.3.5.3. Цикл по счетчику

```
FOR ... NEXT
```

Цикл с определенным количеством повторений выполняется от начального до конечного значения параметра с заданным шагом.

Пример:

```
Dim X As Integer
For X=1 To 10 Step 1 'Повторять цикл для X,
                   изменяющегося от 1 до 10 с шагом1
  Веер 'Звук (тело цикла)
Next X 'Конец цикла
```

Exit For или Exit Do 'Досрочный выход из цикла

```
FOR EACH ... NEXT
```

Цикл For Each ... Next предназначен для перебора всех элементов из заданного массива или набора объектов.

Пример использования цикла для обработки массива:

```

Sub Mas6()
  Dim i As Integer
  Dim j As Integer
  Dim x(1 To 5, 1 To 5) As Single
  Dim S As Single
  Worksheets("Лист1").Activate
  For i=1 To 5
    For j=1 To 5
      x(i,j)=Cells(i,j)
    Next j
  Next i
  For Each e In x
    S=S+e
  Next e
  MsgBox("S=") & CStr(S)
End Sub

```

5.4.3.5.4. Вложенные циклы

Совокупность простых циклов, вложенных один в другой, называется сложным (вложенным) циклом. При конструировании сложных циклов необходимо руководствоваться следующими правилами:

- нельзя войти во внутренний цикл, минуя вход внешнего цикла;
- имена параметров простых циклов не должны повторяться в конструкции сложного цикла;
- простые циклы не должны пересекаться в конструкции сложного цикла, то есть окончание внешнего цикла не должно предшествовать окончанию внутреннего цикла.

Примеры:

```

For i=1 to n
  For j=1 to m
    A(i,j)=Int(Sin(j*i)*100)
  Next j
Next i

Do
  X=1
  Z=0

```

```

Do
    S=Int (Rnd (x) *100)
    Z=Z+S
    X=X+1
Loop Until X>=20
Zsr=Z/20
Loop Until Zsr>=25

```

5.4.3.6. Подпрограммы-процедуры и подпрограммы-функции

Подпрограмма – это блок кода между инструкциями Sub и End Sub или Function и End Function.

Подпрограмма-процедура – это блок кода, заключенный между инструкциями Sub и End Sub. Обычно подпрограмму-процедуру принято называть процедурой.

При написании программы нужно учесть одно правило: «Внутри одной процедуры не может быть описана другая процедура».

Синтаксис:

```

Sub <имя> (ByVal <параметр 1> As <тип>,
          ByVal <параметр 2> As <тип>,
          ByVal <параметр 3>,
          ByVal <параметр 4>)
    <блок кода процедуры>
End Sub

```

В скобках указываются необходимые параметры, если параметров нет, то просто пустые парные скобки. Например, напишем программу, выводящую на экран окно с приветствием:

```

Sub Программа_Привет()
    MsgBox ("ПРИВЕТ")
End Sub

```

Другой пример:

```

Sub qwer(ByVal x As Single, y As Single,
         ByVal S As Single)
    Dim Z As Single
    Dim P As Single
    Z=Sin(2*x+3*y)
    P=Cos(x^2+y^3)
    S=Z+P
End Sub

```

Параметры, указанные в скобках заголовка процедуры, называются *формальными*. Параметры, указанные в списке оператора вызова процедуры, называются *фактическими параметрами*.

Ключевые слова `ByVal` и `ByRef` определяют способ передачи значений параметров. `ByVal` используется для указания, что аргумент передается по значению. `ByRef` – аргумент передается по ссылке. Значения фактических параметров, передаваемых по способу `ByVal`, не могут изменяться в теле процедуры во время ее выполнения, то есть во время выполнения процедуры в программе сохраняются неизменными последние значения переменных. Значения фактических параметров, передаваемых по способу `ByRef`, изменяются в случае их изменения в вызываемой процедуре.

Вызов процедуры из другой процедуры можно произвести несколькими способами. Первый способ: `<Имя процедуры> <Список фактических параметров>`. Список должен соответствовать списку формальных параметров, заданному в заголовке процедуры, по количеству и типу.

Пример: `qwer x, y, s` `оператор вызова процедуры

Если требуется использовать несколько процедур с одинаковыми именами, расположенными в разных модулях, то при их вызове перед именем процедуры через точку необходимо указывать имя модуля, в котором расположена процедура.

Синтаксис:

`<Имя модуля>.<Имя процедуры> <Список фактических параметров>`

Второй способ вызова процедуры реализуется с помощью инструкции `Call`.

Синтаксис:

`Call <Имя процедуры> (<Список фактических параметров>)`

В отличие от первого способа здесь список фактических параметров заключается в скобки.

Пример: `Call qwer(x, y, s)`

Подпрограмма-функция – это блок кода, заключенный между инструкциями `Function` и `End Function`. В ней может быть реализован любой алгоритм, но при этом функция обязательно возвращает какое-нибудь значение. Значение возвращается через имя функции.

Синтаксис:

```
Function <имя функции> (ByVal <параметр>  
                        As <тип>) As <Тип>  
    <код функции>  
End Function
```

Пример:

```
Function f(ByVal x As Single) As Single  
    f=Sin(x^2)+Cos(3*x)  
End Function
```

Оператор вызова функции состоит из имени функции и списка фактических параметров, заключенных в скобки.

Пример: $y=f(x)$ `Оператор вызова функции

При необходимости можно указать область видимости процедуры или функции.

Private Sub Программа_Привет() – закрытая процедура. Возможен вызов из модуля, где она находится, то есть подпрограмма доступна для других процедур только того модуля, в котором она описана.

Public Sub Программа_Привет() – открытая процедура. Возможен вызов из любого модуля, то есть подпрограмма доступна для всех других процедур во всех модулях.

Static Sub Программа_Привет() – указывает, что значения локальных переменных процедуры сохраняются в промежутках времени между вызовами этой процедуры.

Private Function f(ByVal x As Single, ByVal y As Single) As Single – закрытая функция. Возможен вызов из модуля, где она находится.

Public Function f(ByVal x As Single, ByVal y As Single) As Single – открытая функция. Возможен вызов из любого модуля.

5.4.3.7. Ввод-вывод с помощью диалоговых окон

В сокращенном варианте инструкции ввода-вывода имеют вид:

```
InputBox (<"сообщение">)  
MsgBox (<"сообщение">)
```

Пример:

```
Sub Krb()
```



```

Dim S As Integer
Dim Sum As Single
Dim P As Integer
Dim Prom As Variant
S=5000
Do
    Prom=InputBox("Введите значение P=")
    IF Not IsNumeric(Prom) Then
        MsgBox("Повторите ввод!")
Loop Until IsNumeric(Prom)
P=Prom
Sum=S+S*0.01*P
MsgBox("Результат S=" & CSng(Sum))
End Sub

```

5.4.3.8. Массивы

Массив – набор однотипных переменных с одним именем, каждая из которых называется элементом массива и имеет свой номер (индекс).

Массивы могут быть: одномерные (для нумерации элементов используется один индекс), двумерные (для нумерации элементов используются два индекса: номер строки, номер столбца) и *N*-мерные. Число измерений может достигать 60.

Кроме того, по способу выделения оперативной памяти для хранения элементов массивы подразделяются на статические и динамические.

5.4.3.8.1. Статические массивы

Статическим называется массив с заранее известным количеством элементов. Синтаксис описания (объявления) статического массива:

```
Dim <Имя массива>(<верхняя граница>) As <Тип>
```

По умолчанию значение нижней границы равно нулю.

```
Dim <Имя массива>(<Нижняя граница>
    To <Верхняя граница>) As <Тип>
```

Примеры:

```
Dim a(10) As Single 'Одномерный массив
    с начальной границей, равной 0
```

```
Dim S(3 To 5) As String 'Одномерный массив
                        с явно заданными границами
Dim Z(1 To 3, 1 To 5) As Byte 'Двумерный массив
```

Для задания по умолчанию нижней границы массива, равной 1, используется инструкции Option Base 1, которая задается в начале модуля.

Примеры:

```
Option Base 1
```

```
-----
Sub Mas1()
  Dim a(5) As Integer
  Dim i As Integer, k As Integer
  Worksheets("Лист1").Select 'Выбрать Лист1
                              из семейства листов
  Cells.Clear 'Очистить ячейки рабочего листа
  K=2
  For i=1 To 5
    a(i)=Int(Rnd(i)*100) 'Формирование
                        массива a(i)
    Cells(k , i+1)=a(i) 'Вывод массива a(i) на
                        рабочий лист.

  Next i
End Sub
```

```
Sub Mas2()
  Dim a(1 to 5) As Integer, k As Integer
  Dim i As Integer
  Worksheets("Лист1").Select
  Cells.Clear
  K=2
  For i = 1 To 5 'Формирование одномерного
                массива a
    a(i)=Int(Rnd(i)*100)
  Next i
  For i=1 To 5 'Вывод элементов массива a(i)
                на рабочий лист.
    Cells(k , i+1)=a(i)
  Next i
End Sub
```

```

Sub Dmas1()
  Dim i As Integer
  Dim j As Integer
  Dim a2(1 to 3, 1 to 5) As Integer
  Worksheets("Лист1").Select
  Cells.Clear
  For i=1 To 3 `Формирование двумерного
                массива a2
    For j=1 To 5
      a2(i , j)=Int(Rnd(i*j)*100)
    Next j
  Next i
  For i = 1 To 3 `Вывод элементов массива
                  a2(i , j) на рабочий лист
    For j = 1 To 5
      Cells(i+1 , j+1) = a2(i , j)
    Next j
  Next i
End Sub

Sub Dmas2()
  Dim a(1 to 5) As Integer, k As Integer
  Dim i As Integer, prom As Variant
  Worksheets("Лист1").Select
  Cells.Clear
  K=2
  For i=1 To 5 `Ввод элементов массива a(i)
                с клавиатуры
    Do
      Prom=InputBox("Введите элемент
                    a(" & CInt(i) & ")=")
      If Not IsNumeric(prom)
        Then MsgBox("Повторите ввод!")
    Loop Until IsNumeric(prom)
    a(i)=prom
  Next i
  For i=1 To 5 `Вывод элементов массива a(i)
                на рабочий лист
    Cells(k,2)="a("& CInt(i) & ")="
    Cells(k,3)=a(i)
  Next i

```

```

    k=k+1
Next i
End Sub

```

5.4.3.8.2. Динамические массивы

Динамическим называется массив, размерность которого определяется в ходе выполнения программы. Синтаксис описания массива: `Dim <Имя массива> () As <Тип>`

Размерность массива устанавливается и изменяется с помощью инструкции `ReDim <Имя массива> (<размерность>)`

Для установки и изменения размерности массива без потери его содержимого применяется инструкция `ReDim Preserve <Имя массива> (<размерность>)`

Для определения параметров динамического массива используются функции:

```

LBound (<Имя> [, <Размерность>])

```

Эта функция возвращает нижнюю границу указанной размерности массива.

```

UBound (<Имя> [, <Размерность>])

```

Данная функция возвращает верхнюю границу указанной размерности массива. Если размерность не указана, то подразумевается значение, равное 1.

Пример:

```

Sub Mas4 ()
    Dim a () As Integer
    Dim i As Integer, k As Integer, j As Integer,
        N As Integer
    Dim prom As Variant
    Worksheets ("Лист1").Select
    Cells.Clear
    k=2
    i=0
    Do
        Prom=InputBox ("Введите количество
                        элементов N=")
        If Not IsNumeric (prom)
            Then MsgBox ("Повторите ввод!")
    Loop Until IsNumeric (prom)

```

```

N=prom
ReDim a(1 to N) As Integer 'Устанавливается
    фактическая размерность массива a
ReDim Preserve a(1 to N) As Integer 'Это
другой вариант установки размерности массива a
с сохранением значений его элементов
Do 'Формирование массива a
    a(i)=Int(Rnd(i)*100)
    i=i+1
Loop Until i=N
For j=1 To N 'Вывод массива a на рабочий лист
    Cells(k,j+1)=a(j)
Next j
End Sub

```

Array(<Список аргументов>)

С помощью такой инструкции создается массив типа Variant. Список аргументов представляет разделенный запятыми список значений, присваиваемых элементам массива.

Пример:

```

Dim День As Variant
День=Array("Пн", "Вт", "Ср", "Чт", "Пт", "Сб")

```

IsArray(<Имя переменной>)

Эта функция используется для проверки факта, является ли переменная типа Variant массивом. Она возвращает значение True, если переменная является массивом, и False в противном случае.

Erase(<Список массивов>)

С помощью этой инструкции повторно инициализируются элементы статических массивов и освобождается память, отведенная для динамических массивов. Список представляет собой имена очищаемых массивов, разделенных запятой. В статических массивах их элементам вместо чисел присваиваются нулевые значения, а строки переменной длины становятся пустыми. В массивах типа Variant каждому элементу присваивается значение Empty.

5.4.4. Основные объекты MS Excel

К числу основных объектов MS Excel, которые описываются в этом разделе, относятся следующие: рабочая книга (Workbook) и семейство рабочих книг (Workbooks), рабочий лист (Worksheet) и семейство рабочих листов (Worksheets), диапазон ячеек или ячейка (Range).

После объекта, обычно через точку «.», указывается свойство или метод. Иногда, чтобы добраться до определенного объекта, нужно пройти иерархию вышестоящих объектов.

Пример:

```
Workbooks ("Книга1.xls").Worksheets ("Лист1").Activate
```

Свойства и методы, которые обеспечивают ссылку на нужный объект в иерархии объектов, называются семействами (наборами).

Семейство `WorkBooks ("Книга1")` обеспечивает доступ к рабочей книге. В скобках указывается имя книги.

Семейство `WorkSheets ("Лист1")` обеспечивает доступ к рабочему листу. В скобках указывается имя листа.

Семейство `Range ("диапазон")` обеспечивает доступ к диапазону ячеек или к ячейке. В скобках указывается диапазон ячеек или имя ячейки.

Семейство `Cells (№ строки, № столбца)` обеспечивает доступ к ячейке. В скобках указываются координаты ячейки.

Примеры:

```
WorkBooks ("Книга1")
```

```
WorkSheets ("Лист1")
```

```
Range ("A1")
```

```
Range ("A1:B10")
```

```
Cells (2, 3)
```

```
Cells (k, i+1)
```

5.4.4.1. Объект Workbook и семейство Workbooks

В иерархии MS Excel объект Workbook (рабочая книга) идет сразу после объекта Application и представляет собой файл рабочей книги. Рабочая книга хранится либо в файлах формата XLS (стандартная рабочая книга) или XLA (полностью откомпилированное приложение). Свойства и методы рабочей книги позволяют

работать с файлами. Этот объект входит в семейство (набор) `Workbooks`.

Ссылку на объект можно получить через следующие свойства.

`Workbooks (<Индекс>)` возвращает объект по индексу в наборе.

`Workbooks ("<Имя>")` возвращает объект по имени в наборе.

`ActiveWorkbook` возвращает ссылку на активную книгу в момент выполнения команды.

`ThisWorkbooks` возвращает ссылку на книгу, в которой находится текст исполняемого модуля.

5.4.4.1.1. Свойства

`ActiveSheet` возвращает активный лист книги. Для получения имени листа используется свойство `Name` объекта `Sheet`.

Примеры:

```
MsgBox ("Имя активного листа " &  
        ActiveSheet.Name)
```

или

```
MsgBox (ActiveSheet.Name) – выводит  
имя активного рабочего листа в диалоговом окне.
```

`ActiveDialog` – возвращает активное диалоговое окно.

`ActiveChart` – возвращает активную диаграмму.

`Sheets` – возвращает семейство всех листов книги.

`Worksheets` – возвращает семейство всех рабочих листов книги.

Пример:

```
Sub xjfdhd()  
  For Each s In ActiveWorkbook.Sheets  
    MsgBox (s.Name)  
  Next s  
End Sub
```

`Charts` – возвращает семейство всех диаграмм книги, которые не внедрены в рабочие листы.

`Count` – возвращает число объектов семейства `Workbooks` (количество открытых приложением книг).

`FullName` – возвращает полное имя рабочей книги.

Пример:

```
MsgBox (ActiveWorkbook.FullName)
```

`Name` – возвращает имя активной рабочей книги.

Пример:

```
MsgBox (ActiveWorkbook.Name)
```

Path – возвращает путь к файлу рабочей книги.

Пример:

```
MsgBox (ActiveWorkbook.Path)
```

5.4.4.1.2. Методы

Метод `Activate` активизирует рабочую книгу так, что ее первый рабочий лист становится текущим (доступным для работы).

Пример:

```
WorkBooks ("Книга1").Activate
```

или

```
Workbook.Activate
```

Метод `Close` обеспечивает закрытие рабочей книги. `Close SaveChanges FileName` – закрывает книгу. Параметр `SaveChanges` позволяет управлять сохранением изменений в рабочей книге. Если его значение равно `True`, то изменения сохраняются, если – `False`, то не сохраняются. Параметр `FileName` – строка, содержащая имя файла рабочей книги.

Пример:

```
WorkBooks ("Книга1").Close
```

```
WorkBooks ("Книга1").Close SaveChanges:=True
```

```
Filename:="Книга2"
```

`New Window` предназначен для открытия указанной книги в новом окне.

Пример:

```
WorkBooks ("Книга1").NewWindow
```

`Save` служит для сохранения изменений в рабочей книге.

Пример:

```
WorkBooks ("Книга1").Save
```

`SaveAs FileName` – используется для сохранения книги под другим именем (в другом файле).

`SaveAsCopy` – предназначен для сохранения рабочей книги в другом файле, оставляя ее открытой с прежним именем.

Примеры:

```
WorkBooks ("Книга1").SaveAs FileName:="kdjf.xls"
```

```
ActiveBook.SaveAsCopy FileName:="Моя книга"
```


Open FileName – позволяет открыть рабочую книгу с именем, указанным в параметре FileName.

Пример:

```
Workbooks.Open FileName:="Книга1.xls"
```

5.4.4.1.3. События

Основные события объекта Workbook приведены в табл. 5.13.

Таблица 5.13

| Событие | Когда возникает событие |
|---------------|---------------------------------|
| BeforeClose | При закрытии рабочей книги |
| BeforePrint | Перед печатью рабочей книги |
| BeforeSave | Перед сохранением рабочей книги |
| NewSheet | При добавлении нового листа |
| Open | При открытии рабочей книги |
| SheetActivate | При активизации рабочего листа |

5.4.4.2. Объект Worksheet и семейство Worksheets

В иерархии MS Excel объект Worksheet идет сразу после объекта Workbook, представляет рабочий лист книги и входит в семейство (набор) Worksheets.

Ссылку на объект можно получить с помощью команды Worksheets(Index). Она возвращает ссылку на объект по индексу в наборе, в качестве индекса может выступать имя листа или его номер в наборе.

Примеры:

```
Worksheets("Лист1").Activate
```

```
Worksheets(1).Activate
```

ActiveSheet – возвращает ссылку на активный лист.

Пример:

```
ActiveSheet.Range("a1")=1
```

5.4.4.2.1. Свойства

Свойство Name позволяет работать с именем рабочего листа.

Пример:

```
Worksheets(1).Name="Итоги"
```

ActiveCell – возвращает активную ячейку активного рабочего листа.

Cells – возвращает ссылку на диапазон ячеек листа.

Cells (<строка>, <столбец>) – возвращает ссылку на ячейку с указанными координатами.

Columns (<столбец>) – возвращает ссылку на столбец.

В качестве параметра могут быть заданы имя или номер столбца.

Пример:

```
Worksheets(1).Columns("a")=1
```

или

```
Worksheets(1).Columns(1)=1
```

Rows (<строка>) – возвращает ссылку на строку. В качестве параметра может быть использован номер строки.

Пример:

```
Worksheets(1).Rows(1)=1
```

Range (<Диапазон ячеек>) – возвращает ссылку на указанный диапазон ячеек.

UsedRange – возвращает ссылку на используемый диапазон листа.

Пример:

```
Worksheets("Лист1").UsedRange.Value=1
```

Count – возвращает количество листов в книге.

Visible – определяет отображение рабочего листа в книге. Его допустимые значения: True – рабочий лист выводится на экран; False – рабочий лист невидим (скрыт), но его можно отобразить на экране с помощью последовательности команд *Формат, Лист, Отобразить (Format, Sheet, Show)*; xlVeryHidden – рабочий лист скрыт и его можно отобразить на экране только программно.

Примеры:

```
Sub Пусто()
```

```
    Worksheets("Лист3").Visible=False
```

```
End Sub
```

```
Sub Открыто()
```

```
    Worksheets("Лист3").Visible=True
```

```
End Sub
```

```
Sub Пусто1()
```

```
    Worksheets("Лист4").Visible=xlVeryHidden
```

```
End Sub
```

```
Sub Открыто1()
    Worksheets("Лист4").Visible=True
End Sub
```

UsedRange – возвращает диапазон, то есть объект Range, содержащий данные.

Пример:

```
Worksheets(1).UsedRange.Clear 'Очищается
    диапазон первого рабочего листа с данными
```

5.4.4.2.2. Методы

Метод Activate позволяет активизировать рабочий лист.

С помощью метода Evaluate выражения преобразуются в объекты или в значения.

Пример:

```
Sub Ввод_формул()
    Dim Значение As Single
    Dim Fx As String
    Ячейка InputBox("Введите имя ячейки") 'Ввод
        адреса ячейки в диалоговом окне
        ЗначениеEvaluate(Ячейка)
    Value 'Считывание значения из ячейки
    MsgBox CStr(Ячейка) &"=" & CSng(Значение)
        'Вывод значения в диалоговое окно
    Fx=InputBox("Введите формулу (функцию)")
        'Ввод функции в диалоговом окне
    ЗначениеEvaluate(Fx) 'Вычисление значения
        функции
    MsgBox CStr(Ячейка) &"=" & CSng(Значение)
        'Вывод значения в диалоговом окне
End Sub
```

5.4.4.3. Объект Range

В иерархии MS Excel объект Range (диапазон) идет сразу после объекта Worksheet и является одним из ключевых объектов VBA. Он не входит в состав никакого семейства объектов.

Объект Range описывает диапазон ячеек рабочего листа. При работе с ним имеются три способа записи ссылок (табл. 5.14) на

ячейки рабочего листа: относительная адресация (начало координат, задающее нумерацию строк и столбцов, связывается с объектом, вызвавшим Range), абсолютная и смешанная адресация.

Таблица 5.14

| Стили ссылок | Способы записи адресов ячеек |
|--------------|---|
| A1 | Имя ячейки состоит из имени столбца (их 256 – от A до IV) и номера строки (от 1 до 65536). Пример: A1, C2 – <i>относительный адрес</i> |
| R1C1 | Ссылка задается индексом строки и индексом столбца. Пример: R1C1, R2C2, R1C3 – <i>абсолютный адрес</i> |
| A1 | Признаком абсолютной адресации является символ «\$», предшествующий номеру строки (A\$12 – <i>смешанный адрес</i>), имени столбца (\$A12 – <i>смешанный адрес</i>), тому и другому (\$A\$12 – <i>абсолютный адрес</i>) |
| R1C1 | Можно указать смещение по отношению к активной ячейке. Смещение приводится в квадратных скобках, причем знак указывает на направление смещения. Пример: R2C3 ⇒ R[1]C[-1] ⇒ R3C2 |

Записанный в таком виде адрес ячейки рабочего листа является лишь частью полного адреса, который в общем случае включает имя рабочего листа и имя файла рабочей книги. При задании полного адреса за именем листа следует знак «!», а имя файла рабочей книги заключается в квадратные скобки.

Примеры:

A1 – относительная ссылка на ячейку A1 активного рабочего листа.

Лист2!A1 – относительная ссылка на ячейку A1 рабочего листа Лист2 активной книги.

[ВсепрВсе.xls]Лист2!A1 – относительная ссылка на ячейку A1 рабочего листа Лист2 книги ВсепрВсе.xls текущего рабочего каталога.

Если в диапазоне указываются только имена столбцов или строк, то объект Range задает диапазон, состоящий из указанных столбцов или строк.

Примеры:

Range ("A:C") – задает диапазон столбцов A, B, C.

Range ("2:4") – задает диапазон строк 2, 3, 4.

Так как ячейка является частным случаем диапазона, то объект Range позволяет также работать и с ней. Альтернативным способом работы с ячейкой является объект Cells (ячейки).

Пример:

Range ("A2") или Cells (1, 2) – ячейка A2 описывается как объект. В свою очередь объект Cells, вкладываясь в Range, также позволяет записывать диапазон в альтернативном виде: Range ("A2:C3") или Range (Cells (1, 2), Cells (3, 3)).

5.4.4.3.1. Свойства

Свойство Formula позволяет установить формулу в ячейке. Формула задается в виде строки.

FormulaArray – устанавливает формулу массива ячеек. Формула задается в виде строки. В качестве ссылок на ячейки используется стиль A1. Формула массива – это формула, которая в качестве исходных данных использует диапазон ячеек и возвращает одно или несколько значений.

Пример:

```
Sub Primal()  
  With Worksheets("Лист1")  
    For i = 1 To 3  
      For j = 1 To 3  
        Cells(i,j) = Int(Rnd(i*j)*100)  
      Next j  
    Next i  
    Range("D1:F3").FormulaArray="=MINVERSE(A1:C3)"  
  End With  
End Sub
```

FormulaR1C1 – устанавливает формулу в ячейке. Формула задается в строковом виде стилем ссылок R1C1.

Пример:

```
Sub Primal()  
  With Worksheets("Лист1")  
    For i = 1 To 3  
      For j = 1 To 3  
        Cells(i,j) = Int(Rnd(i*j)*100)  
      Next j  
    End With  
End Sub
```

```

Next i
Range("D1:F3").FormulaR1C1="=MINVERSE(R1C1:R3C3)"
End With
End Sub

```

Свойство `HasArray` возвращает значение `True`, если указанная ячейка является частью массива. Массивом на рабочем листе называется именованный диапазон ячеек.

Свойство `HasFormula` возвращает значение `True`, если в указанной ячейке содержится формула.

Примеры:

`MsgBox Worksheets(1).Range("A2").HasArray` – выводимое значение `True` или `False`

`MsgBox Worksheets(1).Range("A3").HasFormula` – выводимое значение `True` или `False`

Свойство `Text` возвращает содержимое ячейки в виде строки. Используется только для чтения.

Пример:

`MsgBox Worksheets(1).Range("A2").Text`

Свойство `Value` возвращает значение из ячейки или устанавливает значение в ячейку (в ячейки).

Примеры:

`x=Range("C1").Value` – значение из ячейки C1 присваивается переменной `x`

`Range("A1:B4").Value=12` – в диапазон ячеек A1:B4 устанавливается число 12

Инструкция `With` используется для указания текущего объекта. Внутри ее можно указывать, начиная с точки, только свойства и методы при обращении к текущему объекту.

Синтаксис:

```

With <объект>
    .<свойства и методы>
End With

```

Пример:

```

Sub qwe()
    Dim a As Single
    Dim b As Single
    With Worksheets("Лист1")
        a=.Range("A1").Value

```

```

    b=.Range("B1").Value
    .Range("C1").Value=a+b
    .Range("D1").Formula="=A1+B1"
End With
End Sub

```

Команда Set предназначена для закрепления объекта за переменной. Переменная должна быть типа Object или с типом объекта, который за ней будет закреплен.

Примеры:

```

Sub Prima1()
    Dim Lst As Object
    Set Lst=Workbooks("Книга1.xls").Worksheets("Лист1")
        `За переменной Lst закрепляется
        рабочий Лист1 рабочей Книги1
    Lst.Cells(1,1)=10 `В ячейку A1 рабочего
        Листа1 устанавливается значение 10
End Sub

```

```

Sub Prima2()
    Dim Lst As Worksheet `Переменная Lst
        с типом объекта WorkSheet
    Set Lst=Workbooks("Книга1.xls").Worksheets("Лист1")
    Lst.Cells(1,1)=10
End Sub

```

5.4.4.3.2. Методы

Метод Activate позволяет активизировать указанный диапазон ячеек.

С помощью метода AddComment <текст примечания> добавляются примечания к ячейке (пункт *Добавить примечание* контекстного меню).

Методом AutoFill <источник> производится автозаполнение диапазона данными из указанных ячеек.

Пример:

```

Sub Prima3()
    Worksheets(1).Activate
    For i = 1 To 10
        Cells(i,1)=i
    Next i

```

```

Range ("B1").Formula= "=sin(A1)"
Range ("B1").AutoFill Range ("B1:B10")
End Sub

```

Метод `Clear` очищает указанный диапазон ячеек.

`ClearContents` – очищает формулы и значения, содержащиеся в ячейках, представляемых объектом `Range`. При этом очищается только содержимое ячеек, форматирование же их сохраняется.

Метод `ClearFormats` удаляет все форматирование ячеек, сохраняя неизменными хранящиеся в них данные.

Метод `Sort Key1:=<ячейка> Order1:=<порядок> Orientation:=<направление>` позволяет осуществить сортировку указанного диапазона ячеек. `Key1` – ключ (строка или столбец), по которому будет производиться сортировка. `Order1` – задает порядок сортировки и может принимать два значения: `xlAscending` – по возрастанию (в алфавитном или хронологическом порядке); `xlDescending` – по убыванию (в обратном алфавитному или в ретроспективном порядке). `Orientation` – указывает направление сортировки: `xlSortRows` – сортировка данных в строке; `xlSortColumns` – сортировка в столбце.

Пример:

```

Sub bmv ()
Worksheets (1).Activate
Range ("A1:D10").Sort
    Key1:=Range ("B1") Okder1:=xlDescending
End Sub

```

5.4.4.4. Форматирование рабочего листа

Внешнее оформление ячеек и данных в них производится с помощью свойств объектов `Range`, `Interior`, `Font`, `Border` и свойства `NumberFormat`.

5.4.4.4.1. Оформление текста в ячейке

`Format` – свойство объекта, задающее формат вывода данных.

Пример:

```

Range ("A1").Format="шаблон"

```

Шаблоны, используемые в MS Excel, приведены в табл. 5.15.

Таблица 5.15

| Шаблон | Назначение |
|--------------|---|
| "#,##0.000" | Формат для вывода чисел с точностью до третьего знака после запятой. Специальные символы: # – необязательное число; 0 – обязательное число; . – разделитель тысячных; , – разделитель между целой и дробной частью |
| "#,##0.00\$" | \$ – обозначение национальной валюты |
| "dd/mm/yy" | Вывод даты в формате 11.01.99, где: d – день; m – месяц; yy – две последние цифры года |
| "dd/mm/yyyy" | Вывод даты в формате 01.01.1999 |
| "d/mm" | Вывод даты в формате 1.01 |
| "d mmm yy" | 1 янв. 99, где mmm – сокращенное название месяца |
| "d mmmm yy" | 1 января 99, где mmmm – полное название месяца |
| "h:mm" | Время в формате 1:23, где h – часы, mm – минуты |
| "h:mm:ss" | Время в формате 1:23:00, где ss – секунды |
| "0.00%" | Процентный формат, к примеру, 3 – это 300 % |
| "@" | Текстовый формат |
| "0.00E+00" | Экспоненциальный формат числа |

Примеры:

```

Range("C1:C6").Select
Selection.NumberFormat="0,0"
Range("D1:D6").Select
Selection.NumberFormat="dd/mm/yy"
Range("E1:E6").Select
Selection.NumberFormat="h:mm:ss"
Range("F1:F6").Select
Selection.NumberFormat="@"

```

Управление выравниванием текста в ячейках возможно в двух направлениях и задается следующими свойствами.

`HorizontalAlignment` – определяет выравнивание по горизонтали. Для выравнивания текста в ячейках по горизонтали используются следующие константы, список которых приведен в табл. 5.16.

`VerticalAlignment` – определяет выравнивание текста по вертикали. Для выравнивания текста в ячейках по вертикали могут использоваться константы, список которых приведен в табл. 5.17.

Таблица 5.16

| Константа | Вид выравнивания |
|--------------------------------------|---------------------|
| <code>xlLeft</code> | По левому краю |
| <code>xlRight</code> | По правому краю |
| <code>xlCenter</code> | По центру |
| <code>xlGeneral</code> | По значению |
| <code>xlJustify</code> | По ширине |
| <code>xlFill</code> | С заполнением |
| <code>xlCenterAcrossSelection</code> | По центру выделения |

Таблица 5.17

| Константа | Вид выравнивания |
|-----------------------|------------------|
| <code>xlTop</code> | По верхнему краю |
| <code>xlBottom</code> | По нижнему краю |
| <code>xlCenter</code> | По центру |

`WrapText` – определяет перенос по словам текста внутри ячейки. Возможные значения `True` – разрешить перенос и `False` – запретить.

`MergeCells` – задает объединение ячеек. Принимаемые значения `True` – ячейки объединены и `False` – ячейки не объединены.

`ShrinkToFit` – включает автоматический подбор ширины ячейки по содержимому. Возможные значения `True` – включено и `False` – выключено.

`Orientation` – определяет ориентацию текста в ячейке, для чего используются константы, список которых приведен в табл. 5.18.

Таблица 5.18

| Константа | Направление ориентации |
|------------------|---------------------------------|
| xlHorizontal | Горизонтальная ориентация |
| xlVertical | Вертикальная ориентация |
| от -90 до 90 | Угол поворота текста в градусах |

Примеры:

```

Range("B2:B5").Select
With Selection
    .HorizontalAlignment=xlRight
    .VerticalAlignment=xlBottom
    .WrapText=False
    .Orientation=0
    .MergeCells=False
End With
Range("C2:C5").Select
With Selection
    .HorizontalAlignment=xlLeft
    .VerticalAlignment=xlBottom
    .WrapText=False
    .Orientation=0
    .MergeCells=False
End With

```

Таблица 5.19

| Константа | Способ подчеркивания |
|----------------------------------|-----------------------------|
| xlUnderlineStyleNone | Нет подчеркивания |
| xlUnderlineStyleSingle | Одинарное по значению |
| xlUnderlineStyleDouble | Двойное по значению |
| xlUnderlineStyleSingleAccounting | Одинарное по ширине ячейки |
| xlUnderlineStyleDoubleAccounting | Двойное по ширине ячейки |

Параметры шрифта задаются с помощью свойств объекта Font. При этом возможно использование следующих параметров: Name – имя шрифта; Size – размер шрифта; FontStyle – начер-

тание (обычный, курсив, полужирный, полужирный курсив); ColorIndex – определяет цвет шрифта, в качестве которого задается его номер; xlAutomatic (константа) – цвет по умолчанию; Color – задает произвольный цвет с использованием палитры **RGB**; Underline – подчеркивание текста в ячейке (табл. 5.19).

Пример:

```
Range("D3:F3").Select
With Selection.Font
    .Name="Arial Cur"
    .Size=14
    .Underline=xlUnderlineStyleNone
    .Color Index=xlAutomatic
End With
Selection.Font.Bold=True
Selection.Font.ColorIndex=3
```

5.4.4.4.2. Оформление границ ячейки

Оформление границ ячеек задается с помощью свойства объекта Border. Для диапазона ячеек границы делятся на внешние и внутренние. Доступ к определенным границам производится через набор Borders (границы). Границы могут задаваться константами, список которых приведен в табл. 5.20.

Таблица 5.20

| Константа | Граница |
|--------------------|--|
| xlEdgeBottom | Внешняя нижняя |
| xlEdgeTop | Внешняя верхняя |
| xlEdgeLeft | Внешняя левая |
| xlEdgeRight | Внешняя правая |
| xlInsideHorizontal | Внутренняя горизонтальная |
| xlInsideVertical | Внутренняя вертикальная |
| xlDiagonalDown | От верхнего правого до нижнего левого угла |
| xlDiagonalUp | От нижнего левого до верхнего правого угла |

Параметры границ определяются за счет следующих свойств. LineStyle – задает стиль линии (табл. 5.21). Weight – определя-

ет толщину линии (табл. 5.22). Цвет границы задается свойствами ColorIndex и Color.

Таблица 5.21

| Константа | Стиль линии |
|------------------|--------------------------|
| xlContinuous | Сплошная линия |
| xlDash | Прерывистая линия |
| xlDashDot | Пунктирная линия |
| xlDashDotDot | Двойная пунктирная линия |
| xlDot | Точечная линия |
| xlDouble | Двойная линия |
| xlLineStyleNone | Нет линии |

Таблица 5.22

| Константа | Толщина линии |
|------------------|----------------------|
| xlHairline | Сверхтонкая |
| xlThin | Тонкая |
| xlMedium | Средняя |
| xlThick | Жирная |

Примеры:

```

Range ("C1:F3") .Select
Selection.Borders (xlDiagonalDown) .LineStyle=xlNone
Selection.Borders (xlDiagonalUp) .LineStyle=xlNone
With Selection.Borders (xlEdgeLeft)
    .LineStyle=xlContinuous
    .Weight=xlThin
    .ColorIndex=xlAutomatic
End With
With Selection.Borders (xlInsideHorizontal)
    .LineStyle=xlContinuous
    .Weight=xlThin
    .ColorIndex=xlAutomatic
End With
With Selection.Borders (xlInsideVertical)
    .LineStyle=xlContinuous
    .Weight=xlThin
    .ColorIndex=xlAutomatic
End With

```

5.4.4.3. Заливка ячейки

Цвет и узор, которыми заполняется ячейка, определяются с помощью свойства объекта Interior. Цвет заливки задается свойствами Color или ColorIndex. Pattern – определяет узор заливки ячейки (табл. 5.23).

Таблица 5.23

| Константа | Узор заливки |
|-------------------|-------------------------------------|
| xlSolid | Сплошной |
| xlGray75 | 75 % – серый |
| xlGray50 | 50 % – серый |
| xlGray25 | 25 % – серый |
| xlGray16 | 12,5 % – серый |
| xlGray8 | 6,25 % – серый |
| xlHorizontal | Горизонтальный штриховой |
| xlVertical | Вертикальный штриховой |
| xlDown | Перевернутый диагональный штриховой |
| xlUp | Диагональный штриховой |
| xlChecker | Диагональный клетчатый |
| xlSemiGray75 | Толстый диагональный клетчатый |
| xlLightHorizontal | Тонкий горизонтальный штриховой |
| xlLightVertical | Тонкий вертикальный штриховой |
| xlLightDown | Тонкий перевернутый диагональный |
| xlLightUp | Тонкий диагональный |
| xlGrid | Тонкий горизонтальный клетчатый |
| xlCrissCross | Тонкий диагональный клетчатый |

Цвет узора заливки задается свойствами PatternColorIndex и PatternColor.

Пример:

```
Range("C1:F3").Select  
With Selection.Interior  
    .ColorIndex=6  
    .Pattern=xlSolid  
End With
```

5.4.5. Контрольные вопросы

- 1) Выберите вариант записи чисел, соответствующий типу Integer.
 - а) 123; -12; 0,001 б) 24,1; 1E03 в) "2345"; 43; -1E03
 - г) 123; -12
- 2) В каком из вариантов переменные A и B целого типа описаны (объявлены) явно?
 - а) Dim A, B
 - б) Dim A as Integer, B as Byte
 - в) Dim A as Integer, B as Single
 - г) Dim A as Long, B as Single
- 3) Назовите вариант, в котором идентификаторы A и B объявлены константами.
 - а) Dim B=7, A="зима"
 - б) Public B=7, A="зима"
 - в) Const B=7, Dim A="зима"
 - г) Public Const a= "зима", Const B as Single=1.25
- 4) Какого типа значения возвращает функция IsNumeric()?
- 5) Для какого из вариантов заголовков передача аргументов A и B в процедуру будет организована по ссылке?
 - а) Sub pr(ByRef A as Long, B as String) ...
end Sub
 - б) Sub pr(ByVal A as Long, ByRef B as String) ...
end Sub
 - в) Sub pr(ByVal A as Long, ByVal B as String) ...
end Sub
 - г) Sub pr(A as Long, ByVal B as String) ...
end Sub
- 6) Для какого из вариантов заголовков передача аргументов A и B в процедуру будет организована по значению?
 - а) Sub pr(ByRef a as Long, b as String) ...
end Sub
 - б) Sub pr(ByVal a as Long, ByRef b as String) ...
end Sub
 - в) Sub pr(ByVal a as Long, ByVal b as String) ...
end Sub

```
г) Sub pr( a as Long, ByVal b as String) ...  
    end Sub
```

7) Укажите вариант записи условного оператора в полной форме.

а) `if x<>0 then y=x/sin(x)`

б) `if x<>0 then y=x/sin(x) else y=1 end if`

в) `if x<>0 then y=x/sin(x) else y=1`

8) Какой из условных операторов содержит сложное условие, состоящее из двух простых, соединенных логической операцией «И»?

а) `if x>-1 and x<>0 then y=ln(1+x)`

б) `if x>-1 end x<>0 then y=ln(1+x)`

в) `if x>-1 or x<>0 then y=ln(1+x)`

г) `if x>-1 and not x=0 then y=ln(1+x)`

9) Чем является конструкция "7359"?

а) числом, б) именем, в) строкой, г) недопустимой записью?

10) Какой результат получится при вычислении выражения `253 \ 10`?

11) Какой результат получится при вычислении выражения `258 mod 10`?

12) В каком из вариантов переменная объявлена (описана) явно?

а) `Private x` б) `r As Date` в) `Dim x As Byte`

г) `Public Z As Long`

13) Какая функция возвращает код символа?

14) С помощью какой функции можно выделить только целую часть вещественного числа и отбросить его дробную часть?

15) Какого типа цикл использован в данном фрагменте программы?

Do

```
Prom = InputBox ("Введите значение P=")
```

```
IF Not IsNumeric(Prom) Then MsgBox ("Повторите  
ввод!")
```

```
Loop Until IsNumeric (Prom)
```

16) Какая функция возвращает символ по его коду?

17) Как называются переменные, объявленные в теле подпрограммы-функции?

18) Какого типа цикл использован в данном фрагменте программы?

```
While x <=b
```

```
    x=x+b
```

```
    s=a+b*x
```


Wend

19) С помощью какой функции можно вычислить длину строки?

20) Какое имя у функции, выделяющей целую часть вещественного числа с округлением?

21) Какого типа цикл использован в данном фрагменте программы?

Do

y=f(x)

i=i+1

Loop While x<=b+h/2

22) Какого типа переменной могут присваиваться числовое или строковое значения?

23) Каким образом может быть продолжена строка описания переменной Dim St As?

24) Какого типа переменная Prom, значение которой вводится в этом фрагменте программы?

Do

Prom = InputBox ("Введите значение P=")

IF Not IsNumeric (Prom) Then MsgBox ("Повторите
ввод!")

Loop Until IsNumeric (Prom)

25) Как называется последовательная запись констант, переменных, имен функций, разделенных знаками операций и круглыми скобками?

26) Где хранится код программы во время ее выполнения?

27) С помощью какого ключевого слова осуществляется явное определение типа переменной?

28) Какие из переменных Dim x(), n, k As Byte, S As Single объявлены явно?

29) Каков тип переменной S в этом выражении?

S = 23 mod 4 + 234\10 + 42 mod 2 + 1234\10

30) Назовите номера элементов заданного массива, удовлетворяющих условию (A(i) mod 2=0) and (A(i)>0), если A(1)=6, A(2)=9, A(3)=-12, A(4)=13, A(5)=10.

31) Какие из логических выражений принимают значение ИСТИНА при x кратном 3?

a) x mod 3 = 0

b) x mod 3 =1 or x mod 3 = 2

c) not (x mod 3 = 1 or x mod 3 = 2)

32) Какие из явно объявленных переменных целого типа являются локальными?

```
Dim N As Byte
```

```
Sub Ex()
```

```
    Dim y() As Single, I, j, k As Integer
```

```
End Sub
```

33) С помощью каких ключевых слов определяется область действия переменных?

a) Dim, Variant

б) Dim, As, To

в) Public, Private, Static, Dim

г) Public, Private, Static, Dim, Const

34) Назовите опцию, требующую явного описания всех используемых переменных и констант.

35) С помощью какого ключевого слова объявляются переменные и константы, сохраняющие свои значения до конца выполнения программы?

Глава 6. Локальные и глобальные компьютерные сети

При соединении двух и более компьютеров образуется компьютерная сеть. Сеть взаимодействующих вычислительных систем, абонентских пунктов и терминалов, связанных между собой каналами передачи данных, называется *вычислительной сетью*.

Для создания компьютерных сетей необходимо специальное аппаратное обеспечение (сетевое оборудование) и программное обеспечение (сетевые программные средства).

Основной задачей, решаемой при создании компьютерных сетей, является обеспечение совместимости оборудования по электрическим и механическим характеристикам и совместимости информационного обеспечения (программ и данных) по системе кодирования и формату данных. Решение этой задачи относится к области стандартизации и основано на модели взаимодействия открытых систем – Model of Open System Interconnections (OSI). Она создана на основе технических предложений Международного института стандартов – International Standards Organization (ISO). Согласно модели ISO/OSI архитектура компьютерных сетей включает семь уровней: *прикладной, уровень представления, сеансовый уровень, транспортный, сетевой, уровень соединения и физический*. Самый верхний уровень – *прикладной*. На этом уровне пользователь взаимодействует с вычислительной системой. Самый нижний уровень – *физический*. Он обеспечивает обмен сигналами между устройствами. Обмен данными в системах связи происходит путем их перемещения с верхнего уровня на нижний. Затем транспортировка и, наконец, обратным воспроизведением на компьютере клиента в результате перемещения с нижнего уровня на верхний уровень.

Для обеспечения необходимой совместимости на каждом из семи возможных уровней действуют специальные стандарты – *протоколы*. Они определяют характер аппаратного взаимодействия компонентов сети (*аппаратные протоколы*) и взаимодействия программ (*программные протоколы*). Физически функции поддержки протоколов исполняют аппаратные устройства (*интерфейсы*) и программные средства (*программы поддержки протоколов*). В соответствии с используемыми протоколами компьютерные сети подразделяются на локальные и глобальные.

В локальных компьютерных сетях преимущественно используется единый комплект протоколов для всех участников. По территориальному признаку локальные сети отличаются компактностью. Они могут объединять компьютеры одного помещения, этажа, здания, группы компактно расположенных сооружений. Создание локальных сетей характерно для отдельных предприятий или отдельных подразделений предприятий.

Глобальные сети имеют, как правило, увеличенные географические размеры. Они могут объединять как отдельные компьютеры, так и отдельные локальные сети, в том числе и использующие различные протоколы. Создание глобальных сетей характерно для предприятия или отрасли, занимающих обширную территорию. В этом случае в одну глобальную сеть могут объединяться отдельные локальные сети. Такие локальные сети связывают между собой с помощью традиционных каналов связи (кабельных, радиорелейных, спутниковых). Для связи между собой нескольких локальных сетей, работающих по разным протоколам, служат специальные средства, называемые шлюзами. При подключении локальной сети предприятия к глобальной сети важную роль играет сетевая безопасность. К числу мер безопасности относятся:

- ограничение доступа в локальную сеть извне для посторонних лиц;
- ограничение выхода за пределы локальной сети для сотрудников предприятия, не имеющих соответствующих прав;
- установление между локальной и глобальной сетями *брандмауэров* (брандмауэром может быть специальный компьютер или компьютерная программа, препятствующая несанкционированному перемещению данных между сетями).

Предназначение всех видов компьютерных сетей определяется двумя функциями:

- обеспечением совместного использования аппаратных и программных ресурсов;
- обеспечением совместного доступа к ресурсам данных.

Все участники локальной сети могут совместно использовать одно общее устройство печати (*сетевой принтер*), ресурсы жестких дисков одного выделенного компьютера (*файлового сервера*). Это же относится и к программному, и к информационному обеспечению. Совокупность приемов разделения и ограничения прав

участников локальной сети называется *политикой сети*. Управление сетевыми политиками называется *администрированием сети*. Лицо, управляющее организацией работы участников сети, называется системным *администратором*.

В модели ISO/OSI обмен данными между удаленными клиентами сети происходит следующим образом.

- На прикладном уровне пользователь с помощью специальных приложений создает документ (сообщение, рисунок и тому подобное).
- На уровне представления операционная система его компьютера фиксирует, где находятся созданные данные (в оперативной памяти, в файле на жестком диске и тому подобное) и обеспечивает взаимодействие со следующим уровнем.
- На сеансовом уровне компьютер пользователя взаимодействует с локальной или глобальной сетью. Протоколы этого уровня проверяют права пользователя на выход в сеть и передают документ к протоколам транспортного уровня.
- На транспортном уровне документ преобразуется в ту форму, в которой положено передавать данные в используемой сети.
- Сетевой уровень определяет маршрут движения данных в сети.
- Уровень соединения необходим для того, чтобы промодулировать сигналы, циркулирующие на физическом уровне, в соответствии с данными, полученными с сетевого уровня.
- Реальная передача данных происходит на физическом уровне. Здесь нет ни документов, ни пакетов, ни даже байтов – только биты. Средства физического уровня находятся за пределами компьютера. В локальных сетях это оборудование самой сети. В глобальных сетях это линии телефонной связи, коммутационное оборудование телефонных станций и тому подобное.

На компьютере получателя информации происходит обратный процесс преобразования данных от битовых сигналов до документа.

Некоторые понятия и терминология

- 1) Компьютер, представляющий свои ресурсы другим компьютерам при совместной работе, называется *сервером*.
- 2) Сервер сети – это компьютер, предоставляющий доступ к ресурсам.
- 3) Верным является утверждение «В электронное письмо можно вкладывать файлы, рисунки, видеоролики».
- 4) Неверным является утверждение «Локальные сети имеют увеличенные географические размеры».
- 5) Для обеспечения необходимой совместимости в компьютерной сети действуют специальные стандарты – *протоколы*.
- 6) Топология компьютерной сети может быть *звездной, кольцевой, шинной*.
- 7) В компьютерных сетях используется коммутация *каналов, сообщений, пакетов*.
- 8) Интернет-служба E-mail обеспечивает *пересылку писем*.
- 9) Интернет-служба FTP обеспечивает *пересылку файлов*.
- 10) Интернет-служба DNS обеспечивает назначение *доменных имен*.
- 11) Системой обмена текстовой информацией, позволяющей всем пользователям интернета участвовать в групповых дискуссиях, является UseNet.
- 12) Для объединения сетей используются *мосты, маршрутизаторы, шлюзы*.
- 13) Программа-браузер Internet Explorer позволяет открывать и просматривать *веб-страницы*.
- 14) К службам сети Интернет относятся FTP (служба передачи файлов), E-mail (электронная почта), WWW (World Wide Web).
- 15) К службам сети Интернет не относится HTML (язык разметки гипертекста).
- 16) Устройство, обеспечивающее сохранение формы и амплитуды сигнала при передаче его на большее, чем предусмотрено данным типом физической передающей среды, расстояние, называется *повторителем*.
- 17) Телефонный кабель является вариантом *витой пары*.
- 18) Устройство, коммутирующее несколько каналов связи на один канал путем частотного разделения, называется *концентратором*.

- 19) Топология сети определяется способом соединения узлов сети *каналами (кабелями) связи*.
- 20) Физическая передающая среда сети не может быть *оптической*.
- 21) Устройство сопряжения ЭВМ с несколькими каналами связи называется *шлюзом*.
- 22) Для установки истинности отправителя сообщения по сети используется *электронно-цифровая подпись*.
- 23) В локальных вычислительных сетях в качестве передающей среды используются *витая пара проводов, коаксиальный кабель, оптоволоконный кабель*.
- 24) Наиболее защищенными от несанкционированного доступа линиями связи сегодня являются *оптоволоконные*.
- 25) Устройством, соединяющим две сети, использующие одинаковые методы передачи данных, является *мост*.
- 26) Устройство, коммутирующее несколько каналов связи, называется *коммутатором*.
- 27) Электронно-цифровая подпись (ЭЦП) документа позволяет получателю удостовериться только *в истинности отправителя документа*.
- 28) Устройство, выполняющее модуляцию и демодуляцию информационных сигналов при их передаче и при приеме, называется *модемом*.
- 29) Наиболее эффективным способом коммуникации для передачи компьютерного трафика являются *пакеты*.
- 30) Кольцевая, шинная, звездообразная – это типы *сетевых топологий*.
- 31) Компьютерные сети реализуют *распределенную обработку информации*.
- 32) Абонентами сети являются объекты, *генерирующие или потребляющие информацию сети*.
- 33) Топология сети *в виде снежинки* не является базовой.
- 34) Устройством для преобразования цифровых сигналов в аналоговую форму является *модем*.
- 35) Компьютер, имеющий две сетевые карты и предназначенный для соединения сетей, называется *мостом*.
- 36) Модем – это устройство *для модуляции-демодуляции сигналов*.
- 37) Усредненная геометрическая схема соединений узлов сети называется *топологией сети*.

- 38) Линией связи с минимальной задержкой является *волоконно-оптическая линия связи*.
- 39) Абонентами сети не могут быть *конечные пользователи* (если имеются в виду люди...).
- 40) News – является одной из рубрик телеконференций, *выделяющей информацию и новости*.
- 41) FTP-сервер – это компьютер, на котором содержатся файлы, *предназначенные для открытого доступа*.
- 42) Comp – является одной из рубрик телеконференций, выделяющей *темы, связанные с компьютером*.
- 43) Soc – является одной из рубрик телеконференций, выделяющей *социальную тематику*.
- 44) Sci – является одной из рубрик телеконференций, выделяющей *темы из области научных исследований*.
- 45) Дискуссионная группа, входящая в состав Usenet, называется *телеконференцией*.
- 46) Программа The Bat позволяет *загружать и редактировать электронную почту*.
- 47) Протокол SMTP предназначен для *отправки электронной почты*.
- 48) Internet Explorer позволяет загружать веб-страницы по протоколу HTTP и файлы по протоколу FTP.
- 49) Для чтения электронной почты предназначены программы Outlook Express, The Bat.
- 50) Система Usenet используется для *перемещения новостей между компьютерами по всему миру*.
- 51) Сервис Telnet является программой для работы с *удаленным компьютером*.
- 52) Адрес веб-страницы для просмотра в браузере начинается с *http://*.
- 53) Стандарты, определяющие формы представления и способы пересылки сообщений, процедуры их интерпретации, правила совместной работы различного оборудования в сетях, – это *сетевые протоколы*.

54) Наиболее эффективным средством для защиты от сетевых атак является *использование сетевых экранов, или Firewall*.

Примечания.

1) *Топология* (от греч. *topos* – место и ...логия) – раздел математики, изучающий топологические свойства фигур, т. е. свойства, не изменяющиеся при любых деформациях, производимых без разрывов и склеиваний (точнее, при взаимно однозначных и непрерывных отображениях). Примерами топологических свойств фигур являются размерность, число кривых, ограничивающих данную область, и т. д. Так, окружность, эллипс, контур квадрата имеют одни и те же топологические свойства, т. к. эти линии могут быть деформированы одна в другую описанным выше образом; в то же время кольцо и круг обладают различными топологическими свойствами: круг ограничен одним контуром, а кольцо – двумя.

2) *Топологическое пространство* – математическое понятие, обобщающее понятие метрического пространства. Топологическое пространство – множество элементов любой природы, в котором тем или иным способом установлены предельные соотношения.

Глава 7. Основы защиты информации и сведений, составляющих государственную тайну

7.1. Информационная безопасность и ее составляющие

С технологической точки зрения информация является продукцией информационных систем. Как и для всякого продукта, для информации большое значение имеет ее качество, то есть способность удовлетворять определенные информационные потребности.

Качество информации является сложным понятием, его основу составляет базовая система показателей, включающая показатели трех классов:

- класс выдачи (своевременность, актуальность, полнота, доступность и другие);
- класс обработки (достоверность, адекватность и другие);
- класс защищенности (физическая целостность, логическая целостность, безопасность).

Своевременность информации оценивается временем выдачи (получения), в течение которого информация не потеряла свою актуальность.

Актуальность информации – это степень ее соответствия текущему моменту времени. Нередко с актуальностью связывают коммерческую ценность информации. Устаревшая и потерявшая свою актуальность информация может приводить к ошибочным решениям и тем самым терять свою практическую ценность.

Полнота информации определяет достаточность данных для принятия решений или для создания новых данных на основе имеющихся. Чем полнее данные, тем проще подобрать метод, вносящий минимум погрешностей в ход информационного процесса.

Достоверность информации – это степень соответствия между получаемой и исходящей информацией.

Адекватность информации – это степень соответствия реальному объективному состоянию дела. Неадекватная информация может образовываться при создании новой информации на основе неполных или недостаточных данных. Однако и полные, и достоверные данные могут приводить к созданию неадекватной информации в случае применения к ним неадекватных методов.

Доступность информации – мера возможности получить ту или иную информацию. Отсутствие доступа к данным или отсутст-

вие адекватных методов обработки данных приводят к одинаковому результату: информация оказывается недоступной.

Одним из наиболее существенных показателей качества информации является ее *безопасность*. В качестве предмета защиты рассматривается информация, хранящаяся, обрабатываемая и передаваемая в компьютерных системах. Особенности этой информации являются:

- двоичное ее представление внутри системы, независимо от физической сущности носителей исходной информации;
- высокая степень автоматизации обработки и передачи информации;
- концентрация большого количества информации в компьютерных системах (КС).

Понятие КС охватывает следующие системы:

- ЭВМ всех классов и назначений;
- вычислительные комплексы и системы;
- вычислительные сети (локальные, глобальные).

Информация доступна человеку, если она содержится на материальном носителе. Поэтому необходимо защищать материальные носители информации, так как с помощью материальных средств можно защищать только материальные объекты.

Информация имеет ценность, которая определяется степенью ее полезности для владельца. В свою очередь степень полезности информации зависит от ее истинности или достоверности. Истинной информацией является та, которая с достаточной точностью отражает объекты и процессы окружающего мира в определенных временных и пространственных рамках. Если информация искажена, то она является дезинформацией. Если к информации ограничен доступ, то такая информация является конфиденциальной. Такая информация может содержать государственную или коммерческую тайну.

Государственную тайну могут содержать сведения, принадлежащие государству. Сведениям, представляющим ценность для государства, могут быть присвоены следующие степени секретности (гриф):

- особой важности;
- совершенно секретно;
- секретно.

Коммерческую тайну могут содержать сведения, принадлежащие частному лицу, фирме, корпорации и тому подобное. Сведениям, представляющим коммерческую тайну, могут быть присвоены следующие категории:

- коммерческая тайна – строго конфиденциально или строгий учет;
- коммерческая тайна – конфиденциально или строго конфиденциально;
- коммерческая тайна или конфиденциально.

Безопасность (защищенность) информации в КС – это состояние всех компонент компьютерной системы, обеспечивающее на требуемом уровне защиту информации от возможных угроз. Безопасность информации в КС (информационная безопасность) является одним из основных направлений обеспечения безопасности государства, отрасли, ведомства, государственной организации или частной структуры.

Информационная безопасность достигается проведением руководством соответствующего уровня политики информационной безопасности. Определяющим документом, на основе которого проводится политика информационной безопасности, является *программа информационной безопасности*. Этот документ разрабатывается и принимается как официальный руководящий документ высшими органами управления государством, ведомством, организацией. На основе этого документа создается комплексная система защиты информации на уровне соответствующей структуры (государства, отрасли, ведомства, учреждения).

Под системой защиты информации в КС понимается единый комплекс правовых норм, организационных мер, технических, программных и криптографических средств, обеспечивающий защищенность информации в КС в соответствии с принятой политикой безопасности.

7.2. Угрозы безопасности информации в компьютерных системах

Под *угрозой безопасности информации* понимается возможное событие, процесс или явление, которое может привести к уничтожению, утрате целостности, конфиденциальности или доступности информации. Все множество потенциальных угроз безопасности

информации в автоматизированных информационных системах (АИС) или в КС может быть разделено на два класса: случайные угрозы и преднамеренные угрозы.

Угрозы, которые не связаны с преднамеренными действиями злоумышленников и реализуются в случайные моменты времени, называются *случайными* или *непреднамеренными*. К случайным угрозам относятся: стихийные бедствия и аварии, сбои и отказы технических средств, ошибки при разработке АИС или КС, алгоритмические и программные ошибки, ошибки пользователей и обслуживающего персонала. Реализация угроз этого класса приводит к наибольшим потерям информации (по статистическим данным – до 80 % от ущерба, наносимого информационным ресурсам КС любыми угрозами). При этом может происходить уничтожение, нарушение целостности и доступности информации. Реже нарушается конфиденциальность информации, однако при этом создаются предпосылки для злоумышленного воздействия на информацию. Согласно тем же статистическим данным только в результате ошибок пользователей и обслуживающего персонала происходит до 65 % случаев нарушения безопасности информации.

Следует отметить, что механизм реализации случайных угроз изучен достаточно хорошо и накоплен значительный опыт противодействия этим угрозам. Современная технология разработки технических и программных средств, эффективная система эксплуатации автоматизированных информационных систем, включающая обязательное резервирование информации, позволяют значительно снизить потери от реализации угроз этого класса.

Угрозы, которые связаны со злоумышленными действиями людей, а эти действия носят не просто случайный характер, а как правило, являются непредсказуемыми, называются *преднамеренными*. К преднамеренным угрозам относятся традиционный или универсальный шпионаж и диверсии, несанкционированный доступ к информации, электромагнитные излучения и наводки, несанкционированная модификация структур, вредительские программы.

В качестве источников нежелательного воздействия на информационные ресурсы по-прежнему актуальны методы и средства шпионажа и диверсий. К методам шпионажа и диверсий относятся подслушивание, визуальное наблюдение, хищение документов и машинных носителей информации, хищение программ и атрибутов

систем защиты, подкуп и шантаж сотрудников, сбор и анализ отходов машинных носителей информации, поджоги, взрывы, вооруженные нападения диверсионных или террористических групп.

Несанкционированный доступ к информации – это нарушение правил разграничения доступа с использованием штатных средств вычислительной техники или автоматизированных систем. Несанкционированный доступ возможен:

- при отсутствии системы разграничения доступа;
- при сбое или отказе в компьютерных системах;
- при ошибочных действиях пользователей или обслуживающего персонала компьютерных систем;
- при ошибках в системе распределения доступа;
- при фальсификации полномочий.

Процесс обработки и передачи информации техническими средствами компьютерных систем сопровождается электромагнитными излучениями в окружающее пространство и наведением электрических сигналов в линиях связи, сигнализации, заземлении и других проводниках. Все это получило название «побочные электромагнитные излучения и наводки» (ПЭМИН). Электромагнитные излучения и наводки могут быть использованы злоумышленниками как для получения информации, так и для ее уничтожения.

Большую угрозу безопасности информации в компьютерных системах представляет несанкционированная модификация алгоритмической, программной и технической структуры системы.

Одним из основных источников угроз безопасности информации в КС является использование специальных программ, получивших название «вредительские программы». В зависимости от механизма действия вредительские программы делятся на четыре класса:

- логические бомбы;
- черви;
- троянские кони;
- компьютерные вирусы.

Логические бомбы – это программы или их части, постоянно находящиеся в ЭВМ или вычислительных системах и выполняемые только при соблюдении определенных условий. Примерами таких условий могут быть наступление заданной даты, переход КС в оп-

ределенный режим работы, наступление некоторых событий заданное число раз и тому подобное.

Черви – это программы, которые выполняются каждый раз при загрузке системы, обладают способностью перемещаться в вычислительных системах или в сети и самовоспроизводить копии. Лавинообразное размножение программ приводит к перегрузке каналов связи, памяти и блокировке системы.

Троянские кони – это программы, полученные путем явного изменения или добавления команд в пользовательские программы. При последующем выполнении пользовательских программ наряду с заданными функциями выполняются несанкционированные, измененные или какие-то новые функции.

Компьютерный вирус – это достаточно сложная и своеобразная программа, выполняющая несанкционированные пользователем действия, которая пытается тайно записать себя на компьютерные диски. Большинство компьютерных вирусов распространяются одним из двух методов:

- записываются в файлы (записываются в каждый исполняемый или объектный файл, размещенный на компьютере);
- записываются на загрузочные секторы жесткого или флоппи-диска, помещенного в дисковод зараженного компьютера.

Способ функционирования компьютерных вирусов – это такое изменение системных файлов компьютера пользователя, в результате которого компьютерный вирус начинает свою деятельность либо при каждой загрузке, либо в момент, когда происходит некоторое «событие вызова».

Компьютерные вирусы – это одна из наиболее широко известных опасностей для сетей. В настоящее время имеется около тысячи видов вирусов. Учитывая тот факт, что каждый из них существует в нескольких модификациях, это число следует увеличить в 5 ÷ 10 раз. Однако только 5 % всех компьютерных вирусов могут вызвать серьезные неполадки в аппаратном обеспечении или разрушение операционной системы.

По типу поведения вирусы можно классифицировать следующим образом:

- вирусы, поражающие загрузочный сектор, пытаются заменить или инфицировать часть диска, зарезервированную

только для операционной системы и хранения файлов запуска;

- вирусы, инфицирующие файлы, обычно EXE- и COM-исполняемые файлы;
- многофункциональные вирусы, использующие для заражения компьютерной системы как загрузочный сектор, так и метод заражения файлов;
- вирусы-невидимки, использующие ряд методов для маскировки своего присутствия – они фальсифицируют фактические значения контрольных сумм. Контрольная сумма – это результат выполнения математического алгоритма, проверяющего соответствие длины полученного файла длине его первоначальной копии. Проверка контрольных сумм – один из способов выявления и идентификации вируса антивирусными программами;
- системные вирусы, поражающие операционные системы. Основные жертвы этих вирусов – это таблица размещения файлов (оглавление диска), таблицы разделов, драйверы устройств и системные файлы.

Существует несколько различных классов (типов) вирусов. Вирусы каждого из классов используют различные методы воспроизведения. К наиболее известным классам относятся следующие.

- *Троянские кони* – это компьютерные вирусы, скрытые в файлах данных и в исполняемых файлах, то есть эта программа может располагаться и в программных файлах, и в файлах библиотек, пришедших в сжатом виде. Троянский конь – это небезопасная программа, скрывающаяся под видом безобидного приложения типа архиватора, игры или программы обнаружения и уничтожения вирусов.
- *Полиморфные вирусы* – это компьютерные вирусы, которые зашифровывают свое тело. Прежде чем приступить к работе, такой компьютерный вирус расшифровывает себя с помощью специальной процедуры расшифровки, которая превращает зашифрованную информацию в обычную. В процессе расшифровки названная процедура захватывает управление компьютером, которое затем передается расшифрованному компьютерному вирусу. Полиморфные вирусы обнаружить очень трудно, поскольку они генериру-

ют абсолютно новые процедуры расшифровки при каждом новом инфицировании.

- *Стелс-вирусы* – это вирусы, которые скрывают изменения, созданные в инфицированном файле. Эти вирусы либо обладают невидимым размером, либо невидимы для чтения.
- *Медленные вирусы* – это вирусы, инфицирующие только те файлы, которые изменяются или копируются операционной системой. Медленный вирус инфицирует любой исполняемый файл в тот момент, когда выполняются некоторые операции с этим файлом.
- *Ретро-вирус* – это компьютерный вирус, пытающийся обойти или помешать действиям антивирусных программ. Эти вирусы атакуют антивирусное программное обеспечение.
- *Составные вирусы* – это вирусы, инфицирующие как исполняемые файлы, так и загрузочные секторы дисков, то есть инфицируют компьютер различными путями.
- *Вооруженные вирусы* – это вирусы, защищающие себя с помощью специального кода, благодаря которому сильно усложняется отслеживание и обезвреживание компьютерного вируса.
- *Вирусы-компаньоны* – это вирусы, создающие параллельно с инфицированным файлом файл с таким же именем, но с другим расширением имени.
- *Вирус-фаг* – это программа, которая изменяет другие программы или базы данных. Вирусы-фаги обычно замещают текст программ своим собственным кодом. Чаще всего они являются генераторами вирусов-компаньонов. Вирусы-фаги – это наиболее опасный вид компьютерных вирусов. Они не только размножаются и заражают другие программы, но и стремятся уничтожить все инфицированные программы.
- *Вирус-червь (репликатор)* стремится вывести из строя инфицированный компьютер. Эти вирусы копируют себя на другие компьютеры с помощью протоколов и иных подобных систем. Для нормальной работы вирусам-червям необходимы операционные системы, обеспечивающие возможность удаленного выполнения и позволяющие приходящим программам выполняться на компьютере. Современные операционные системы обладают такими

возможностями и поэтому могут поддерживать работу вирусов-червей.

Файловые вирусы и макровирусы – это наиболее опасные типы компьютерных вирусов. Файловым вирусом может быть троянский конь, вооруженный вирус, стелс-вирус и некоторые другие. Файловые вирусы опасны для данных, хранящихся на сервере, одноранговых сетей и, в какой-то степени, для сети Интернет. Известны следующие способы инфицирования сетевого сервера.

- Копирование инфицированных файлов прямо на сервер. После этого компьютерный вирус, расположившийся в файле, начнет инфицировать все остальные файлы.
- Наличие файлового вируса на рабочей станции может инфицировать всю сеть. После своего запуска вирус сможет инфицировать любое приложение, хранимое на сервере. Если же вирус сумеет проникнуть в какой-либо файл, расположенный на сервере, то он сможет инфицировать и все компьютерные системы в сети.
- Наличие резидентного вируса на рабочей станции может вызвать инфицирование всей сети. После своего запуска резидентный вирус может получить информацию о передаваемых данных и скопировать себя на сервер, не обладая при этом прямым доступом к расположенной на сервере информации.

Файловые вирусы не могут размножаться в интернете и инфицировать удаленные компьютеры, поэтому интернет является не «инкубатором» для компьютерных вирусов, а их носителем. Чтобы произошло инфицирование, компьютер должен загрузить зараженный файл из сети и запустить его.

В последнее время возросло количество компьютерных вирусов, распространяемых с помощью интернета. Обычные текстовые сообщения электронной почты не могут содержать компьютерные вирусы. Текстовые сообщения относятся к файлам данных, которые не являются программами и не могут выполняться в компьютерной системе. Компьютерные вирусы могут заражать только исполняемые программы (не считая макровирусов). Однако существуют ситуации, когда файл с данными оказывается исполняемым файлом. Некоторые веб-браузеры и диалоговые служебные утилиты выполняют программы сразу после их запуска.

В большинстве случаев файлы данных текстовых процессоров (документы MS Word) содержат внедренные макрокоманды, с помощью которых можно существенно ускорить работу. Макрокоманда является некоторой разновидностью программ, благодаря этому стало возможным создание макровирусов. *Макровирус* – это макрокоманда, которая воспроизводит себя в каждом новом документе.

7.3. Методы защиты информации

Защита информации в компьютерных сетях основывается на выявлении, оценке и парировании возможных угроз безопасности информации с учетом частоты их проявления и вероятности потери информации, возможности по противодействию и ликвидации последствий проявления этих угроз. Защищенные информационные системы предполагают:

- безотказность;
- безопасность;
- конфиденциальность;
- бизнес-этику.

Безотказность – компьютерная система готова к работе в любой момент, когда в этом возникает необходимость, и функционирует, как и ожидается, на соответствующем уровне.

Безопасность – система устойчива к атакам, секретность и целостность ее данных находится под защитой.

Конфиденциальность – у пользователей есть возможность контролировать данные о самих себе, а те, кто добросовестно использует эти данные, следуют принципам честного использования информации.

Бизнес-этика – компании-разработчики несут ответственность перед клиентами, помогают им найти решения, отвечающие их потребностям, и в отношениях с клиентами действуют открыто.

Защита информации в компьютерных системах обеспечивается созданием комплексной системы защиты. Комплексная система защиты включает:

- правовые методы защиты;
- организационные методы защиты;
- методы защиты от случайных угроз;
- методы защиты от традиционного шпионажа и диверсий;

- методы защиты от электромагнитных излучений и наводок;
- методы защиты от несанкционированного доступа;
- криптографические методы защиты;
- методы защиты от компьютерных вирусов.

Среди методов защиты имеются и универсальные, которые являются базовыми при создании любой системы защиты. Это, прежде всего, правовые методы защиты информации, которые служат основой легитимного построения и использования системы защиты любого назначения. К числу универсальных методов можно отнести и организационные методы, которые используются в любой системе защиты без исключений и, как правило, обеспечивают защиту от нескольких угроз.

Методы защиты от случайных угроз разрабатываются и внедряются на этапах проектирования, создания, внедрения и эксплуатации компьютерных систем. К их числу относятся:

- создание высокой надежности компьютерных систем;
- создание отказоустойчивых компьютерных систем;
- блокировка ошибочных операций;
- оптимизация взаимодействия пользователей и обслуживающего персонала с компьютерной системой;
- минимизация ущерба от аварий и стихийных бедствий;
- дублирование информации.

При защите информации в компьютерных системах от традиционного шпионажа и диверсий используются те же средства и методы защиты, что и для защиты других объектов, на которых не используются компьютерные системы. К их числу относятся:

- создание системы охраны объекта;
- организация работ с конфиденциальными информационными ресурсами;
- противодействие наблюдению и подслушиванию;
- защита от злоумышленных действий персонала.

Все методы защиты от электромагнитных излучений и наводок можно разделить на пассивные и активные. Пассивные методы обеспечивают уменьшение уровня опасного сигнала или снижение информативности сигналов. Активные методы защиты направлены на создание помех в каналах побочных электромагнитных излучений и наводок, затрудняющих прием и выделение полезной информации из перехваченных злоумышленником сигналов. На

электронные блоки и магнитные запоминающие устройства могут воздействовать мощные внешние электромагнитные импульсы и высокочастотные излучения. Эти воздействия могут приводить к неисправности электронных блоков и стирать информацию с магнитных носителей информации. Для блокирования угрозы такого воздействия используется экранирование защищаемых средств.

Для защиты информации от несанкционированного доступа создаются:

- система разграничения доступа к информации;
- система защиты от исследования и копирования программных средств.

Исходной информацией для создания системы разграничения доступа является решение администратора компьютерной системы о допуске пользователей к определенным информационным ресурсам. Так как информация в компьютерных системах хранится, обрабатывается и передается файлами (частями файлов), то доступ к информации регламентируется на уровне файлов. В базах данных доступ может регламентироваться к отдельным ее частям по определенным правилам. При определении полномочий доступа администратор устанавливает операции, которые разрешено выполнять пользователю. Различают следующие операции с файлами:

- чтение (R);
- запись (W);
- выполнение программ (E).

Операции записи имеют две модификации:

- субъекту доступа может быть дано право осуществлять запись с изменением содержимого файла (W);
- разрешение дописывания в файл без изменения старого содержимого (A).

Система защиты от исследования и копирования программных средств включает следующие методы:

- методы, затрудняющие считывание скопированной информации;
- методы, препятствующие использованию информации.

Под криптографической защитой информации понимается такое преобразование исходной информации, в результате которого она становится недоступной для ознакомления и использования лицами, не имеющими на это полномочий. По виду воздействия на

исходную информацию методы криптографического преобразования разделяются на следующие группы:

- шифрование;
- стенография;
- кодирование;
- сжатие.

Вредительские программы и, прежде всего, вирусы представляют очень серьезную опасность для информации в компьютерных системах. Знание механизмов действия вирусов, методов и средств борьбы с ними позволяет эффективно организовать противодействие вирусам, свести к минимуму вероятность заражения и потерь от их воздействия.

Компьютерные вирусы – это небольшие исполняемые или интерпретируемые программы, обладающие свойством распространения и самовоспроизведения в компьютерных системах. Вирусы могут выполнять изменение или уничтожение программного обеспечения или данных, хранящихся в компьютерных системах. В процессе распространения вирусы могут себя модифицировать. Все компьютерные вирусы классифицируются по следующим признакам:

- по среде обитания;
- по способу заражения;
- по степени опасности вредительских воздействий;
- по алгоритму функционирования.

По среде обитания компьютерные вирусы подразделяются на:

- сетевые;
- файловые;
- загрузочные;
- комбинированные.

Средой обитания *сетевых* вирусов являются элементы компьютерных сетей. *Файловые* вирусы размещаются в исполняемых файлах. *Загрузочные* вирусы находятся в загрузочных секторах внешних запоминающих устройств. *Комбинированные* вирусы размещаются в нескольких средах обитания. Например, загрузочно-файловые вирусы.

По способу заражения среды обитания компьютерные вирусы делятся на:

- резидентные;

- нерезидентные.

Резидентные вирусы после их активизации полностью или частично перемещаются из среды обитания в оперативную память компьютера. Эти вирусы, используя, как правило, привилегированные режимы работы, разрешенные только операционной системе, заражают среду обитания и при выполнении определенных условий реализуют вредительскую функцию.

Нерезидентные вирусы попадают в оперативную память компьютера только на время их активности, в течение которого выполняют вредительскую функцию и функцию заражения. Затем они полностью покидают оперативную память, оставаясь в среде обитания.

По степени опасности для информационных ресурсов пользователя вирусы разделяются на:

- безвредные;
- опасные;
- очень опасные.

Безвредные вирусы создаются авторами, которые не ставят себе цели нанести какой-либо ущерб ресурсам компьютерной системы. Однако такие вирусы все-таки наносят определенный ущерб:

- расходуют ресурсы компьютерной системы;
- могут содержать ошибки, вызывающие опасные последствия для информационных ресурсов;
- вирусы, созданные ранее, могут приводить к нарушениям штатного алгоритма работы системы при модернизации операционной системы или аппаратных средств.

Опасные вирусы вызывают существенное снижение эффективности компьютерной системы, но не приводят к нарушению целостности и конфиденциальности информации, хранящейся в запоминающих устройствах.

Очень опасные вирусы имеют следующие вредительские воздействия:

- вызывают нарушение конфиденциальности информации;
- уничтожают информацию;
- вызывают необратимую модификацию (в том числе и шифрование) информации;
- блокируют доступ к информации;

- приводят к отказу аппаратных средств;
- наносят ущерб здоровью пользователей.

По алгоритму функционирования вирусы подразделяются на:

- не изменяющие среду обитания при их распространении;
- изменяющие среду обитания при их распространении.

Для борьбы с компьютерными вирусами используются специальные антивирусные средства и методы их применения. Антивирусные средства выполняют следующие задачи:

- обнаружение вирусов в компьютерных системах;
- блокирование работы программ-вирусов;
- устранение последствий воздействия вирусов.

Обнаружение вирусов и блокирование работы программ-вирусов осуществляется следующими методами:

- сканирование;
- обнаружение изменений;
- эвристический анализ;
- использование резидентных сторожей;
- вакцинирование программ;
- аппаратно-программная защита.

Устранение последствий воздействия вирусов реализуется следующими методами:

- восстановление системы после воздействия известных вирусов;
- восстановление системы после воздействия неизвестных вирусов.

7.3.1. Профилактика заражения вирусами компьютерных систем

Главным условием безопасной работы в компьютерных системах является соблюдение правил, которые апробированы на практике и показали свою высокую эффективность.

Правило первое. Обязательное использование программных продуктов, полученных законным путем, так как в пиратских копиях вероятность наличия вирусов во много раз выше, чем в официально полученном программном обеспечении.

Правило второе. Дублирование информации, то есть создание копий рабочих файлов на съемных носителях информации (дискеты, компакт-диски и другие) с защитой от записи.

Правило третье. Регулярно использовать антивирусные средства, то есть перед началом работы выполнять программы-сканеры и программы-ревизоры (Aidstest и Adinf). Эти антивирусные средства необходимо регулярно обновлять.

Правило четвертое. Проявлять особую осторожность при использовании новых съемных носителей информации и новых файлов. Новые дискеты и компакт-диски необходимо проверять на отсутствие загрузочных и файловых вирусов, а полученные файлы – на наличие файловых вирусов. Проверка осуществляется программами-сканерами и программами, осуществляющими эвристический анализ (Aidstest, Doctor Web, AntiVirus). При первом выполнении исполняемого файла используются резидентные сторожа. При работе с полученными документами и таблицами нужно запретить выполнение макрокоманд встроенными средствами текстовых и табличных редакторов (MS Word, MS Excel) до завершения полной проверки этих файлов на наличие вирусов.

Правило пятое. При работе в системах коллективного пользования необходимо новые сменные носители информации и вводимые в систему файлы проверять на специально выделенных для этой цели ЭВМ. Это должен выполнять администратор системы или лицо, отвечающее за безопасность информации. Только после всесторонней антивирусной проверки дисков и файлов они могут передаваться пользователям системы.

Правило шестое. Если не предполагается осуществлять запись информации на носитель, то необходимо заблокировать выполнение этой операции.

Постоянное выполнение изложенных правил позволяет значительно уменьшить вероятность заражения программными вирусами и обеспечить защиту пользователя от безвозвратных потерь информации.

В особо ответственных системах для борьбы с вирусами используются аппаратно-программные средства (например, Sheriff).

7.3.2. Порядок действий пользователя при обнаружении заражения вирусами компьютерной системы

Несмотря на строгое выполнение всех правил профилактики заражения вирусами компьютерной системы, нельзя полностью исключить возможность их заражения. Однако, если придерживаться

определенной последовательности действий при заражении вирусами, то последствия пребывания вирусов в компьютерной системе можно свести к минимуму.

О наличии вирусов можно судить по следующим событиям:

- появление сообщений антивирусных средств о заражении или о предполагаемом заражении;
- явные проявления присутствия вирусов (сообщения, выдаваемые на монитор или принтер, звуковые эффекты, уничтожение файлов и другие);
- неявные проявления заражения, которые могут быть вызваны сбоями или отказами аппаратных и программных средств, «зависаниями» системы, замедлением выполнения определенных действий, нарушением адресации, сбоями устройств и другими проявлениями.

При получении информации о предполагаемом заражении пользователь должен убедиться в этом. Решить такую задачу можно с помощью всего комплекса антивирусных средств. Если заражение действительно произошло, тогда пользователю следует выполнить последовательность действий.

- Выключить ЭВМ для уничтожения резидентных вирусов.
- Осуществить загрузку эталонной операционной системы со сменного носителя информации, на котором отсутствуют вирусы.
- Сохранить на сменных носителях информации важные файлы, которые не имеют резидентных копий.
- Использовать антивирусные средства для удаления вирусов и восстановления файлов, областей памяти. Если работоспособность компьютерной системы восстановлена, то завершить восстановление информации всесторонней проверкой компьютерной системы с помощью всех имеющихся в распоряжении пользователя антивирусных средств. В противном случае необходимо продолжить выполнение антивирусных действий.
- Осуществить полное стирание и разметку (форматирование) несъемных внешних запоминающих устройств. В персональных компьютерах для этого могут быть использованы программы MS-DOS FDISK и FORMAT. Программа форматирования FORMAT не удаляет главную загрузочную

запись на жестком диске, в которой может находиться загрузочный вирус. Поэтому необходимо воспользоваться программой FDISK с недокументированным параметром MBR, создать с помощью этой же программы разделы и логические диски на жестком диске. Затем выполняется программа FORMAT для всех логических дисков.

- Восстановить операционную систему, другие программные системы и файлы с резервных копий, созданных до заражения.
- Тщательно проверить файлы, сохраненные после обнаружения заражения, и, при необходимости, удалить вирусы и восстановить файлы.
- Завершить восстановление информации всесторонней проверкой компьютерной системы с помощью всех имеющихся в распоряжении пользователя антивирусных средств.

7.3.3. Особенности защиты информации в базах данных

Базы данных рассматриваются как надежное хранилище структурированных данных, снабженное специальным механизмом для их эффективного использования в интересах пользователей (процессов). Таким механизмом является система управления базами данных (СУБД). Под системой управления базами данных понимается совокупность программных и (или) аппаратно-программных средств, реализующих функции управления данными, такие как добавление, просмотр, модификация (редактирование), удаление, поиск, сортировка, выборка, выполнение операций определения статистических характеристик и другие. Базы данных могут размещаться:

- на компьютерной системе пользователя;
- на специально выделенной ЭВМ (сервере).

На компьютерной системе пользователя, как правило, размещаются личные или персональные базы данных, которые обслуживают процессы одного пользователя.

На серверах базы данных размещаются в локальных и корпоративных компьютерных сетях, которые используются, как правило, централизованно. Общедоступные глобальные компьютерные сети имеют распределенные базы данных. В таких сетях серверы размещаются на различных объектах сети. Серверы – это

специализированные ЭВМ, приспособленные для хранения больших объемов данных и обеспечивающие сохранность и доступность информации, а также оперативность обработки поступающих запросов. В централизованных базах данных решаются проще проблемы защиты информации от преднамеренных угроз, поддержания актуальности и непротиворечивости данных. Достоинством распределенных баз данных является их высокая защищенность от стихийных бедствий, аварий, сбоев технических средств и диверсий, если осуществляется дублирование (резервное копирование) этих данных.

Особенности защиты информации в базах данных:

- необходимость учета функционирования СУБД при выборе механизмов защиты;
- разграничение доступа к информации реализуется не на уровне файлов, а на уровне частей баз данных.

При создании средств защиты информации в базах данных необходимо учитывать взаимодействие этих средств не только с операционной системой, но с СУБД. При этом возможно встраивание механизмов защиты в СУБД или использование их в виде отдельных компонент. Для большинства СУБД придание им дополнительных функций возможно только на этапе их разработки. В эксплуатируемые системы управления базами данных дополнительные компоненты могут быть внесены путем расширения или модификации языка управления.

7.3.4. Программно-технические средства защиты информации

Требования к программно-техническим средствам защиты информации сформулированы в руководящих документах Государственной технической комиссии (ГТК) при Президенте РФ. Основой всего набора таких документов является «Концепция защиты средств вычислительной техники и автоматизированных систем от несанкционированного доступа (НСД) к информации». Этим документом вводится понятие «штатные средства», под которыми понимается совокупность программного, микропрограммного и технического обеспечения средств вычислительной техники и компьютерных сетей. Главным средством защиты считается *система разграничения доступа* (СРД) субъектов к объектам доступа.

Подключение компьютерных сетей к другим информацион-

ным системам и сетям производится через межсетевые экраны не ниже 3 класса защищенности, сертифицированные по требованиям руководящего документа Гостехкомиссии России «Средства вычислительной техники. Межсетевые экраны. Показатели защищенности от несанкционированного доступа к информации».

Объектам компьютерных сетей присваивается различная степень защищенности для определения требуемых организационно-технических мероприятий по защите информации в зависимости от ее важности, реальных условий размещения элементов компьютерных сетей, возможных каналов утечки информации, а также для минимизации затрат на защиту информации. Допускается присваивать различные степени защищенности по отдельным элементам компьютерных сетей при обработке ими информации различной степени конфиденциальности. Установлены следующие степени (категории) защищенности объектов компьютерных сетей в соответствии со степенью конфиденциальности обрабатываемой информации:

- 1 степень – объекты, связанные с обработкой строго конфиденциальной информации;
- 2 степень – объекты, связанные с обработкой конфиденциальной информации;
- 3 степень – объекты, связанные с обработкой служебной информации.

Наиболее совершенные *информационные системы ведения реестра (ИСВР)* и *информационные депозитарные системы (ИДС)* предусматривают функции, обеспечивающие многоуровневую защиту данных, которая включает в себя:

- защиту данных от несанкционированного доступа, то есть использование процедуры аутентификации пользователя по идентификатору и паролю при входе в систему;
- наличие процедур кодирования и шифровки информации;
- многоуровневый доступ к данным – наличие нескольких типов пользователей в соответствии с их полномочиями;
- защиту информации от случайного уничтожения при сбое системы или неправильных действиях персонала;
- наличие средств контроля достоверности и непротиворечивости данных;
- возможность резервного сохранения и последующего восстановления данных.

7.3.4.1. Защита компьютерных сетей с помощью брандмауэров

Брандмауэр – это совокупность аппаратных средств и программного обеспечения, которая связывает две и больше сетей и одновременно разделяет защищенную и незащищенную сети или защищенную область сети от незащищенной области той же сети. Является центральным пунктом управления безопасностью. Обеспечивает контроль взаимного доступа сетей друг к другу и защиту этого соединения сетей от несанкционированного вмешательства.

Брандмауэры могут реализовываться как программно, так и аппаратно-программно. Аппаратные брандмауэры являются специализированными компьютерами. При присоединении сети к сети Интернет основой безопасности этого соединения должен стать брандмауэр. Он обеспечивает возможность центрального управления безопасностью сети. Обычно брандмауэр строится на основе компьютера-бастиона.

Компьютер-бастион – это компьютер, специально установленный для защиты от атак на локальную сеть.

Обычно брандмауэр устанавливается между локальной сетью и интернетом. Он препятствует проникновению пользователей всего мира в локальную сеть и контролирует доступ к данным, хранящимся в ней.

В дополнение к традиционному аппаратному и программному обеспечению брандмауэров для большей безопасности используют экранирующий маршрутизатор.

Экранирующий маршрутизатор (screening router) – это специальный компьютер (созданный, как правило, на базе персонального компьютера) или электронное устройство, экранирующее (фильтрующее) пакеты данных, основываясь на заданных администратором критериях. Критерии для выбора пакетов устанавливаются с помощью специального программного обеспечения или аппаратно.

Для защиты различных N отделов внутри локальной сети от вторжений друг к другу применяется конфигурация брандмауэра, в которой используется один компьютер-бастион с N сетевыми картами.

Для защиты локальной сети от внешних пользователей (сети Интернет) и обеспечения достаточно простой связи с интернетом тем пользователям локальной сети, которым это требуется, приме-

няется сетевая конфигурация, в которой используется сервер (компьютер-бастион) с двумя сетевыми картами. В этом случае сетевые карты подсоединяются к компьютеру через разные, изолированные друг от друга порты.

Чтобы удовлетворить требования широкого диапазона пользователей, существует три типа брандмауэров:

- сетевого уровня;
- уровня приложений;
- уровня соединения.

Брандмауэр сетевого уровня – это обычно экранирующий маршрутизатор или специальный компьютер, который исследует адреса пакетов, чтобы определить, передать пакет в локальную сеть или из нее, либо отклонить его. Пакеты наряду с другой информацией содержат IP-адреса отправителя и получателя, а также массу другой информации, которую брандмауэр использует для управления доступом к пакету. Настройка экранирующего маршрутизатора на блокирование определенных пакетов происходит с помощью файла, который содержит IP-адреса внешних сайтов (мест назначения), чьи пакеты следует блокировать. Брандмауэр (или маршрутизатор) должен блокировать пакеты, в которых эти адреса фигурируют как адреса отправителя или получателя. Подобное блокирование конкретных узлов называют методом черного списка (blacklisting).

Брандмауэр уровня приложения – это главный компьютер сети, использующий программное обеспечение *сервера-посредника*. *Сервер-посредник* – это программа, управляющая трафиком и контролирующая передачу данных между двумя сетями, обеспечивая связь между пользователями локальной сети и серверами внешней сети. Брандмауэры уровня приложения позволяют контролировать тип и объем трафика (передач данных), поступающего на сайт. Эти брандмауэры обеспечивают определенное физическое разделение между локальной сетью и интернетом и тем самым отвечают самым высоким требованиям безопасности.

Брандмауэр уровня соединения подобен брандмауэру уровня приложения – оба они являются серверами-посредниками. Однако для брандмауэра уровня соединения не нужно использовать специальные приложения, поддерживающие режим связи между сервером-посредником и клиентом. Здесь клиент и сервер сообща-

ются через брандмауэр уровня соединения без сообщения с самим брандмауэром.

Наилучшей защитой для сети является использование в архитектуре брандмауэра как маршрутизатора, так и сервера-посредника. Существует три наиболее популярных типа архитектуры брандмауэра:

- двусторонний главный брандмауэр;
- фильтрующий главный брандмауэр;
- фильтрующий брандмауэр подсети.

Двусторонний главный брандмауэр использует две отдельные сетевые карты. Это простая, но обеспечивающая очень высокую степень безопасности конфигурация, в которой один главный компьютер играет роль разделительной линии между локальной сетью и интернетом. Такой брандмауэр использует набор серверов-посредников уровня приложений или уровня соединения.

Фильтрующий главный брандмауэр использует комбинацию маршрутизаторов и серверов-посредников. Он может обеспечивать более высокую степень безопасности, чем двусторонний главный брандмауэр. Здесь для пользователей локальной сети имеется прямой доступ в интернет, в то время как сервер этой сети будет ограничивать доступ внешних пользователей в локальную сеть.

Фильтрующий брандмауэр подсети использует также комбинацию маршрутизаторов и серверов-посредников. Он еще более изолирует локальную сеть от интернета, включая между ними периферийную промежуточную сеть. В этом брандмауэре главный компьютер помещается на периферийной сети, к которой пользователи имеют доступ через два отдельных маршрутизатора. Сервер-посредник помещается в сеть, состоящую из двух экранирующих маршрутизаторов. Один из них контролирует трафик (передачу данных) по локальной сети, а второй контролирует входящие и выходящие из интернета сообщения. Фильтрующий брандмауэр подсети обеспечивает наиболее эффективную защиту от возможных атак через интернет. В дополнение к этому маршрутизатор, находящийся в локальной сети, обеспечивает защиту от внутреннего доступа к серверу.

7.4. Законодательные акты РФ, регулирующие правовые отношения в сфере информационной безопасности и защиты государственной тайны

В государстве должна проводиться единая политика в области безопасности информационных технологий. Это требование нашло отражение в «Концепции национальной безопасности Российской Федерации», утвержденной Указом Президента РФ № 1300 от 17 декабря 1997 года. В этом документе отмечается, что в современных условиях всеобщей информатизации и развития информационных технологий резко возрастает значение обеспечения национальной безопасности РФ в информационной сфере. Значимость обеспечения безопасности государства в информационной сфере подчеркнута и в принятой в сентябре 2000 года «Доктрине информационной безопасности Российской Федерации». В этих документах определены важнейшие задачи государства в области информационной безопасности:

- установление необходимого баланса между потребностью в свободном обмене информацией и допустимыми ограничениями ее распространения;
- совершенствование информационной структуры, ускорение развития новых информационных технологий и их широкое внедрение, унификация средств поиска, сбора, хранения и анализа информации с учетом вхождения России в глобальную информационную инфраструктуру;
- разработка соответствующей нормативно-правовой базы в интересах обеспечения информационной безопасности;
- координация деятельности органов государственной власти и других органов, решающих задачи обеспечения информационной безопасности;
- развитие отечественной индустрии телекоммуникационных и информационных средств, их приоритетное по сравнению с зарубежными аналогами распространение на внутреннем рынке;
- защита государственного информационного ресурса и, прежде всего, в федеральных органах власти и на предприятиях оборонного комплекса.

25 февраля 1995 года Государственной Думой принят Федеральный закон «Об информации, информатизации и защите

информации». В законе даны определения основных терминов: информация, информатизация, информационные системы, информационные ресурсы, конфиденциальная информация, собственник и владелец информационных ресурсов, пользователь информации. Государство гарантирует права владельца информации, независимо от форм собственности, распоряжаться ею в пределах, установленных законом. Владелец информации имеет право защищать свои информационные ресурсы, устанавливать режим доступа к ним. В этом законе определены цели и режимы защиты информации, а также порядок защиты прав субъектов в сфере информационных процессов и информатизации.

Другим важным правовым документом, регламентирующим вопросы защиты информации в КС, является закон РФ «О государственной тайне», принятый 21 июля 1993 года. Закон определяет уровни секретности государственной информации и соответствующую степень важности информации.

Статья 2. Основные понятия, используемые в настоящем Законе.

В настоящем Законе используются следующие основные понятия: государственная тайна – защищаемые государством сведения в области его военной, внешнеполитической, экономической, разведывательной, контрразведывательной и оперативно-розыскной деятельности, распространение которых может нанести ущерб безопасности Российской Федерации.

Носители сведений, составляющих государственную тайну, – материальные объекты, в том числе физические поля, в которых сведения, составляющие государственную тайну, находят свое отображение в виде символов, образов, сигналов, технических решений и процессов.

Система защиты государственной тайны – совокупность органов защиты государственной тайны, используемых ими средств и методов защиты сведений, составляющих государственную тайну, и их носителей, а также мероприятий, проводимых в этих целях.

Допуск к государственной тайне – процедура оформления права граждан на доступ к сведениям, составляющим государственную тайну, а предприятий, учреждений и организаций – на проведение работ с использованием таких сведений.

Доступ к сведениям, составляющим государственную тайну, – санкционированное полномочным должностным лицом ознакомле-

ние конкретного лица со сведениями, составляющими государственную тайну.

Гриф секретности – реквизиты, свидетельствующие о степени секретности сведений, содержащихся в их носителе, проставляемые на самом носителе и (или) в сопроводительной документации на него.

Средства защиты информации – технические, программные, криптографические и другие средства, предназначенные для защиты сведений, составляющих государственную тайну, средства, в которых они реализованы, а также средства контроля эффективности защиты информации.

К государственной тайне могут быть отнесены следующие сведения.

1) Сведения в военной области:

- о содержании стратегических и оперативных планов, документов боевого управления по подготовке и проведению операций, стратегическому, оперативному и мобилизационному развертыванию войск, об их боеспособности и мобилизационной готовности, о создании и использовании мобилизационных ресурсов; о направлениях развития вооружения и военной техники, содержании и результатах выполнения целевых программ, научно-исследовательских и опытно-конструкторских работ по созданию и модернизации образцов вооружения и военной техники; о количестве, устройстве и технологии производства ядерного и специального оружия, технических средствах и методах его защиты от несанкционированного применения;
- о тактико-технических характеристиках и возможностях боевого применения образцов вооружения и военной техники, свойствах, рецептурах или технологиях производства новых видов ракетного топлива или взрывчатых веществ военного назначения;
- о дислокации, назначении, степени готовности и защищенности режимных и особо важных объектов, об их проектировании и строительстве, а также об отводе земель, недр и акваторий для этих объектов;
- о дислокации, организационной структуре, действительных наименованиях, вооружении и численности объединений, соединений и частей Вооруженных Сил Российской Федерации.

2) Сведения в области экономики, науки и техники:

- о содержании планов подготовки Российской Федерации и ее отдельных регионов к возможным военным действиям, мобилизационных мощностях промышленности по изготовлению вооружения и военной техники, об объемах поставок и о запасах стратегических видов сырья и материалов, а также о размещении и фактических размерах государственных материальных резервов;
- об использовании инфраструктуры Российской Федерации в интересах обеспечения ее обороноспособности и безопасности;
- о силах и средствах гражданской обороны, дислокации, предназначении и степени защищенности объектов административного управления, обеспечения безопасности населения, о функционировании промышленности, транспорта и связи в целом по Российской Федерации;
- об объемах, планах (заданиях) государственного оборонного заказа, выпуске и поставках (в денежном или натуральном выражении) вооружения, военной техники и другой оборонной продукции, о наличии и наращивании мощностей по их выпуску, связях предприятий по кооперации, разработчиках или изготовителях указанных вооружения, военной техники и другой оборонной продукции;
- о научно-исследовательских, опытно-конструкторских и проектных работах, технологиях, имеющих важное оборонное или экономическое значение, влияющих на безопасность Российской Федерации;
- о государственных запасах драгоценных металлов и драгоценных камней Российской Федерации, ее финансах и бюджетной политике (кроме обобщенных показателей, характеризующих общее состояние экономики и финансов).

3) Сведения в области внешней политики и экономики:

- о внешнеполитической и внешнеэкономической (торговой, кредитной и валютной) деятельности Российской Федерации, преждевременное распространение которых может нанести ущерб ее интересам.

4) Сведения в области разведывательной, контрразведывательной и оперативно-розыскной деятельности:

- о силах, средствах, источниках, методах, планах и результатах разведывательной, контрразведывательной и оперативно-розыскной деятельности, а также данные о финансировании этой деятельности, если эти данные раскрывают перечисленные сведения;
- о лицах, сотрудничающих или сотрудничавших на конфиденциальной основе с органами, осуществляющими разведывательную, контрразведывательную и оперативно-розыскную деятельность;
- о системе правительственной и об иных видах специальной связи, о государственных шифрах, методах и средствах их анализа;
- о методах и средствах защиты секретной информации; о государственных программах и мероприятиях в области защиты государственной тайны.

Статья 8. Степени секретности сведений и грифы секретности носителей этих сведений.

Степень секретности сведений, составляющих государственную тайну, должна соответствовать степени тяжести ущерба, который может быть нанесен безопасности Российской Федерации вследствие распространения указанных сведений.

Устанавливаются три степени секретности сведений, составляющих государственную тайну, и соответствующие этим степеням грифы секретности для носителей указанных сведений: «особой важности», «совершенно секретно» и «секретно».

Порядок определения размеров ущерба, который может быть нанесен безопасности Российской Федерации вследствие распространения сведений, составляющих государственную тайну, и правила отнесения указанных сведений к той или иной степени секретности устанавливаются правительством Российской Федерации.

Использование перечисленных грифов секретности для засекречивания сведений, не отнесенных к государственной тайне, не допускается.

Статья 9. Порядок отнесения сведений к государственной тайне.

Отнесение сведений к государственной тайне осуществляется в соответствии с их отраслевой, ведомственной или программно-целевой принадлежностью.

Отношения, связанные с созданием программ и баз данных, регулируются Федеральным законом Российской Федерации от 20 июля 2004 № 72-ФЗ «О внесении изменений в Закон Российской Федерации «Об авторском праве и смежных правах».

Важной составляющей правового регулирования в области информационных технологий является установление ответственности граждан за противоправные действия при работе с КС. Преступления, совершенные с использованием КС или причинившие ущерб владельцам КС, получили название компьютерных преступлений.

В Уголовный кодекс РФ, принятый 1 января 1997 года, включена глава № 28, в которой определена уголовная ответственность за преступления в области компьютерных технологий.

В статье 272 предусмотрены наказания за неправомерный доступ к компьютерной информации. Это правонарушение может наказываться от штрафа в размере 200 минимальных зарплат до лишения свободы на срок до 5 лет.

Статья 273 устанавливает ответственность за создание, использование и распространение вредоносных программ для ЭВМ. Это правонарушение может наказываться от штрафа до лишения свободы на срок до 7 лет.

В статье 274 определена ответственность за нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети. Если такое деяние причинило существенный вред, то виновные наказываются лишением права занимать определенные должности или заниматься определенной деятельностью на срок до 5 лет. Если те же деяния повлекли тяжкие последствия, то предусмотрено лишение свободы на срок до 4 лет.

7.5. Контрольные вопросы

- 1) Перечислите классы потенциальных угроз безопасности информации.
- 2) Назовите классы вредительских программ.
- 3) Приведите классификацию вирусов по степени опасности для информационных ресурсов пользователя.
- 4) Как называется несанкционированный перенос информации к злоумышленнику?
- 5) Как называется криптографическое преобразование информации?

- 6) Чем является носитель, в загрузочном секторе которого находится вирус?
- 7) Что должна обеспечивать защита информации?
- а) целостность;
 - б) конфиденциальность;
 - в) ограничение доступности;
 - г) скрытность.
- 8) Что не включает в себя комплексная система защиты информации?
- а) деловые методы;
 - б) методы защиты от случайных угроз;
 - в) методы защиты от несанкционированного доступа;
 - г) методы защиты от компьютерных вирусов.
- 9) Перечислите основные пути проникновения вирусов в компьютер.
- 10) Что является основной антивирусной защитой?
- а) периодическая проверка списка загруженных программ;
 - б) периодическая проверка списка автоматически загружаемых программ;
 - в) периодическая проверка компьютера средствами антивирусной защиты.
- 11) Какими могут быть вирусы?
- а) загрузочными; б) мутантами; в) невидимыми; г) логическими.
- 12) Какой гриф не может иметь информация, составляющая государственную тайну?
- а) особой важности; б) совершенно секретно; в) секретно;
 - г) строго конфиденциально.
- 13) Какая информация подлежит защите?
- а) информация о состоянии операционной системы;
 - б) сведения об окружающем мире;
 - в) информация, приносящая выгоду;
 - г) информация о кадровом составе учреждения.
- 14) Что является преднамеренной угрозой безопасности информации?
- а) ошибка персонала; б) наводнение; в) землетрясение; г) кража.

Заключение

В современном мире количество компьютеров удваивается в среднем каждые три года. При этом в среднем один раз в полтора года удваиваются основные технические параметры аппаратных средств. Один раз в два-три года меняются поколения программного обеспечения, один раз в пять-семь лет меняется база стандартов, интерфейсов и протоколов. Предметная область дисциплины «Информатика» изменяется чрезвычайно динамично, что является кардинальным отличием ее от других технических дисциплин. Поэтому для эффективного использования вычислительной техники от специалистов (пользователей) требуется достаточно высокий уровень базовых знаний и практических навыков, неоценимую помощь в приобретении которых (по крайней мере, авторы питают на этот счет неумиряющую надежду) и может оказать предлагаемое учебное пособие.

Литература

1. Информатика. Т. 1 : Концептуальные основы : учебник / под общ. ред. В. А. Минаева. – 2-е изд., расшир. и доп. – М. : Маросейка, 2008. – 464 с.
2. Информатика. Т. 2 : Средства и системы обработки данных : учебник / под общ. ред. В. А. Минаева. – 2-е изд., расшир. и доп. – М. : Маросейка, 2008. – 544 с.
3. Острейковский, В. А. Информатика : учебник для студентов техн. направлений и специальностей вузов. – 4-е изд., стер. – М. : Высш. шк., 2007. – 511 с.
4. Информатика: Базовый курс : учеб. пособие для студентов вузов / под ред. С. В. Симоновича. – 2-е изд. – СПб. : Питер, 2008. – 640 с.
5. Каймин, В. А. Информатика : учеб. для студентов вузов, обучающихся по естественно-науч. направлениям и специальностям. – 5-е изд. – М. : ИНФРА-М, 2006. – 285 с.
6. Сырецкий, Г. А. Информатика. Фундаментальный курс : Т. 2 : Информационные технологии и системы : учеб. для студентов вузов. – СПб. : БХВ-Петербург, 2007. – 848 с.
7. Меняев, М. Ф. MS Office XP: Эффективный самоучитель : учеб. пособие для студентов вузов. – М. : Омега-Л, 2006. – 432 с.
8. Васильев, А. Н. Excel 2007 на примерах. – СПб. : БХВ-Петербург, 2007. – 656 с.
9. Лавренов, С. М. Excel: сборник примеров и задач. – М. : Финансы и статистика, 2006. – 336 с.
10. Попов, А. А. Excel: практическое руководство : учеб. пособие для вузов. – М. : Десс, 2004. – 301 с.
11. Уокенбах, Дж. Excel 2003. Библия пользователя. – М. : Диалектика, 2004. – 768 с.
12. Программирование в пакетах MS Office : учеб. пособие для студентов вузов / под ред. С. В. Назарова. – М. : Финансы и статистика, 2007. – 656 с.
13. Уокенбах, Дж. Профессиональное программирование на VBA в Excel 2003. – М. : Вильямс, 2005. – 800 с.
14. Роман, С. Использование макросов в Excel. – 2-е изд. – СПб. : Питер, 2004. – 507 с.
15. Кренке, Д. М. Теория и практика построения баз данных. – СПб. : Питер, 2005. – 859 с.

16. Кузнецов, С. Д. Основы баз данных : курс лекций. – М. : Интернет-Университет Информационных Технологий, 2005. – 488 с.
17. Диго, С. М. Базы данных: проектирование и использование : учеб. для студентов вузов. – М. : Финансы и статистика, 2005. – 488 с.
18. Кошелев, В. Е. Access 2007. – М. : Бином-Пресс, 2008. – 592 с.
19. Блюттман К. Анализ данных в Access: сборник рецептов / К. Блюттман, У. Фриз ; пер. с англ. Н. Вильчинского. – СПб. : Питер, 2008. – 350 с.
20. Мак-Дональд, М. Access 2007. Недостающее руководство. – М. : Русская редакция, 2007. – 784 с.
21. Мельников, В. П. Информационная безопасность и защита информации : учеб. пособие для студентов вузов / В. П. Мельников, С. А. Клейменов, А. М. Петраков ; под ред. С. А. Клейменова. – М. : Академия, 2006. – 336 с.
22. Правовое обеспечение информационной безопасности : учебник / под ред. В. А. Минаева. – 2-е изд., расшир. и доп. – М. : Маросейка, 2008. – 368 с.
23. Леонтьев, В. П. Новейшая энциклопедия персонального компьютера 2009. – М. : ОЛМА Медиа Групп, 2008. – 928 с.

Словарь основных терминов и сокращений

Глава 1

Адекватность информации – это степень ее соответствия реальному, объективному состоянию дела.

Актуальность информации – это степень ее соответствия текущему моменту времени.

Безопасность (защищенность) информации в компьютерной системе – это состояние всех компонент компьютерной системы, обеспечивающее на требуемом уровне защиту информации от возможных угроз.

Данные – это факты (зарегистрированные сигналы), характеризующие объекты, процессы, явления.

Дизъюнкция (логическое сложение) – объединение нескольких высказываний в одно с помощью союза «ИЛИ».

Достоверность информации – это степень соответствия между получаемой и исходящей информацией.

Доступность информации – мера возможности получить ту или иную информацию.

Импликация (логическое следование) образуется соединением двух высказываний в одно с помощью оборота речи «если..., то...».

Инверсия (логическое отрицание) – присоединение частицы «НЕ» к высказыванию.

Информатика – это прикладная техническая наука, систематизирующая приемы создания, хранения, воспроизведения, обработки и передачи данных (информации) средствами вычислительной техники, а также принципы функционирования этих средств и методы управления ими.

Информация – это сведения об окружающем нас мире, выраженные с помощью знаков, сигналов, жестов и т. п. или мера устранения неопределенности в отношении интересующего нас события или объекта.

Конъюнкция (логическое умножение) – объединение нескольких высказываний в одно с помощью союза «И».

Полнота информации – это степень достаточности данных для принятия решений или для создания новых данных на основе имеющихся.

Своевременность информации оценивается временем выдачи (получения), в течение которого информация не потеряла свою актуальность.

Эквивалентность (логическое равенство) образуется соединением двух высказываний в одно с помощью оборота речи «... тогда и только тогда, когда...».

Глава 2

Адаптер – это специальный блок, через который осуществляется подключение периферийных устройств.

Виртуальная машина – создаваемая операционной системой абстрактная машина, скрывающая от пользователей и его приложений реальные аппаратные и другие ресурсы.

Виртуальная память – механизм управления памятью, суть которого заключается в том, что при нехватке места в оперативной памяти на диск перемещаются только части образов процессов. В современных операционных системах используется как основной, в качестве дополнительного механизма используется свопинг.

Внутренняя память персонального компьютера включает кэш-память, постоянное запоминающее устройство, оперативное запоминающее устройство.

Дигитайзер – это устройство поточечного координатного ввода графических изображений.

DIMM (Dual In-line Memory Module) – это модуль микросхемы памяти с двухрядным расположением выводов.

Интероперабельность – свойство ресурсов (информационных, программных, аппаратных), заключающееся в возможности их совместного использования в рамках единой системы.

Интерфейс – 1) совокупность средств (программных, языковых, технических) и правил для обеспечения взаимодействия между различными программными системами, между техническими устройствами или между пользователем и системой; 2) способ взаимодействия пользователя с компьютером, определяемый комплексом спецификаций.

Компьютер – это электронный прибор, предназначенный для автоматизации создания, хранения, обработки и транспортировки данных (информации).

Контроллер – это микросхема, управляющая работой конкретного типа внутренних устройств и предназначенная для обеспечения их связи с системной платой.

Конфигурация вычислительной системы – это ее состав, включающий аппаратные и программные средства, которые представляют собой отдельно аппаратную и программную конфигурации.

Модем – это устройство для преобразования данных из цифрового формата в аналоговый и обратно.

Плоттер – это устройство, позволяющее вычерчивать сложные графические изображения.

Принципы фон Неймана работы компьютера – принцип программного управления, принцип однородности памяти, принцип адресности.

Протокол – это совокупность технических условий, обеспечивающих взаимное согласование различных устройств при их совместной работе.

Разрядность процессора – это количество двоичных разрядов, которое может обрабатываться процессором за один такт.

Разрешающей способностью (разрешением) монитора является количество точек (пикселей) на 1 см^2 .

Системная шина – это основная интерфейсная система компьютера, обеспечивающая сопряжение и связь всех его устройств между собой.

Стример – это устройство (накопитель большой емкости) для записи и воспроизведения цифровой информации (содержимого жесткого диска) на кассету с магнитной лентой.

Глава 3

Архитектура операционной системы классическая – архитектура, обеспечивающая операционной системе выполнение основных функций в привилегированном режиме, то есть код ядра имеет доступ к областям памяти всех приложений, но сам полностью от них защищен.

Архитектура операционной системы микроядерная – архитектура, позволяющая в привилегированном режиме работать только небольшой части операционной системы, называемой микроядром.

База данных (БД) – это совокупность относящихся к определенной области знания (теме, проблеме) взаимосвязанных данных, представленных в определенном формате на машинном носителе.

BIOS (Basic Input/Output System) – базовая система ввода/вывода, часть операционной системы.

Безопасность программного обеспечения – это состояние, в котором программы не могут быть использованы, просмотрены и модифицированы неавторизованными лицами, компьютерами или другими программами.

Графический редактор – вид прикладного программного обеспечения общего назначения, предназначенный для создания, редактирования и отображения изображений компьютерной графики.

Драйвер – это программа, обеспечивающая взаимодействие операционной системы с определенным периферийным устройством компьютера.

DrvSpace – это программа уплотнения диска.

Качество программного обеспечения – степень соответствия присущих характеристик (свойств) программного обеспечения установленным или предполагаемым требованиям (потребностям).

Ключ – атрибут или совокупность атрибутов, однозначно идентифицирующих запись БД.

Коммуникационная система – вид прикладного программного обеспечения общего назначения, предназначенный для организации взаимодействия пользователя с удаленными абонентами или информационными ресурсами вычислительной сети.

Конструктор меню – вид инструментальных средств пользователя, предназначенный для выполнения настройки необходимых режимов и команд меню пакета прикладных программ.

Конструктор экранных форм – вид инструментальных средств пользователя, предназначенный для создания новых и изменения существующих экранных форм в пакетах прикладных программ (расположение на экране реквизитов, соответствующих полям базы данных; использование текста подсказок; цветовое оформление и др.).

LINUX – это операционная система (некоммерческая версия UNIX для ПК).

Novell Netware – это операционная система, используемая для управления сетью с выделенным сервером.

Оболочка операционной системы – программа, выполняющая функции удобного и наглядного интерфейса между пользователем и операционной системой.

Общее программное обеспечение – программное обеспечение, ориентированное на пользователей и предназначенное для решения типовых задач обработки информации: управления ресурсами компьютера, создания прикладного программного обеспечения. В общесистемном программном обеспечении можно выделить следующие классы программных средств: операционные системы, вспомогательные программы, инструментарий технологии программирования, прикладное программное обеспечение общего назначения.

Ограничение целостности базы данных – ограничение на значения определенного набора объектов данных и/или их связей, специфицированное в схеме базы данных. Проверка выполнения ограничений целостности осуществляется механизмами СУБД на стадии выполнения. При нарушении ограничений СУБД отвергает вызвавшую его операцию.

Операционная система – вид общего программного обеспечения, предназначенный для управления процессами и ресурсами компьютера; обеспечения запуска всех остальных программ; организации интерфейса с пользователем.

Организер – это программа, позволяющая планировать работу.

OLE Automation – OLE-автоматизация (технология на основе COM, позволяющая прикладной программе-серверу предоставлять свои сервисы в распоряжение других программ-клиентов через диспетчерские интерфейсы (dispinterface – диспетчерский интерфейс)).

Программные средства СУБД – комплекс средств, обеспечивающих выполнение операций с базой данных. Среди программных компонентов СУБД можно выделить ядро СУБД, обеспечивающее создание базы данных, организацию ввода, обработки и хранения данных, т. е. управление данными. Среди других компонент можно назвать средства настройки системы; средства тестирования; утилиты, обеспечивающие выполнение вспомогательных функций (например, восстановление БД). Важным компонентом СУБД являются трансляторы для используемых ею языковых средств.

Программа презентационной графики – вид прикладного программного обеспечения общего назначения, используемый для

создания изображений и показа их на экране; подготовки слайд-фильмов, видеофильмов, мультфильмов, их редактирования и определения порядка их следования.

Программа проверки орфографии – вид прикладного программного обеспечения общего назначения, используемый для проверки орфографии и стилевой правки текстов документов.

Программа распознавания текста – вид прикладного программного обеспечения общего назначения, используемый для распознавания считанной сканером информации и преобразования ее в текстовое представление.

Программа системная обрабатывающая – вид вспомогательных модулей операционной системы, выполняющих функции текстовых или графических редакторов, компиляторов, отладчиков и др.

Программа-переводчик – вид прикладного программного обеспечения общего назначения, используемый для перевода текстов с русского языка на иностранный (английский, немецкий, французский и др.) и обратно.

Реестр – это специальная база данных, в которой хранятся все сведения о конфигурации ОС Windows.

RGB (Red-Green-Blue) – красный-зеленый-синий – основная цветовая палитра, используемая в программировании и компьютерной графике.

RAM (Random-Access Memory) – это запоминающее устройство с произвольной выборкой или оперативная память.

ROM (Read Only Memory) – это память, доступная только для чтения, или постоянное запоминающее устройство. Содержит программы тестирования компьютера, часть операционной системы (BIOS) и т. д.

CASE (Compute-Aided Software/System Engineering) – общее название группы технологий, методов и средств проектирования программного обеспечения, поддерживаемых соответствующими средствами автоматизации этапов анализа, проектирования, разработки и сопровождения систем.

Системный монитор – это программа, анализирующая пиковую загрузку процессора и других ресурсов.

Системные программы – это программы, управляющие работой аппаратных средств и обеспечивающие услуги пользователю.

Служебная программа ОС Windows «Очистка диска» служит для удаления временных файлов интернета, установленных компонентов и программ, которые больше не используются, и очистки корзины.

СМΥК (Cyan-Magenta-Yellow-Black) – голубой-пурпурный-желтый-черный – это цветовая палитра или система для представления и печати цветных изображений.

Система управления базами данных (СУБД) – это комплекс языковых и программных средств, предназначенных для создания баз данных, поддержания их в актуальном состоянии, организации доступа к ним различных пользователей.

COM (Component Object Model) – модель компонентных объектов Microsoft (стандартный механизм, включающий интерфейсы, с помощью которых одни объекты предоставляют свои сервисы другим, – является основой многих объектных технологий, в том числе OLE и ActiveX).

Стандарт OLE (Object Linking and Embedding) – стандарт межпрограммного интерфейса, обеспечивающий обмен объектами между приложениями путем встраивания или связывания, а также их совместное использование в формировании новых составных ресурсов.

Стандарт SQL (Structured Query Language) – международный стандарт языка структурированных запросов в реляционных системах баз данных. Признанием доминирующей роли языка в реляционных системах баз данных было утверждение ANSI и ISO/IEC целого ряда стандартов SQL. В настоящее время продолжают разработки незавершенных компонентов языка и новых его элементов, которые значительно расширят функциональность SQL.

Стандарт UML (Unified Modelling Language) – стандарт унифицированного языка моделирования, использующего графическую нотацию и предназначенный для спецификации, визуализации, конструирования и документирования систем программного обеспечения на основе объектно-ориентированных методов. Язык UML не зависит от конкретных языков программирования, используемых при реализации разрабатываемых систем. Он не ориентируется на какой-либо конкретный процесс разработки и может быть адаптирован к различным процессам. Язык UML стал

играть ведущую роль среди используемых в CASE-системах объектных языков моделирования.

Стандарт XML (eXtensible Markup Language) – стандарт языка разметки для веб-документов, разработанный консорциумом W3C для представления информационных ресурсов WWW второго поколения. Язык XML стал основой активно развивающейся более совершенной и перспективной технологической платформы для среды WWW.

Стандарт GUI (Graphical User Interface) – стандарты на графический пользовательский интерфейс, обеспечивающие единообразное управление программным обеспечением, навигацию и реакцию программного обеспечения на действия пользователя. В соответствии со стандартами GUI пользовательский интерфейс характеризуется использованием многооконной технологии, пиктограмм, меню, кнопок, курсоров и других визуальных элементов, а также различных устройств-манипуляторов, необходимых для работы пользователя в такой среде.

Стандарт ODBC (Open Database Connectivity) – стандарт открытого интерфейса прикладного программирования для SQL-серверов, обеспечивающий доступ к реляционным базам данных.

Табличный процессор – вид прикладного программного обеспечения общего назначения, позволяющий обрабатывать большие массивы чисел (информации, представимой в виде таблиц). Включает средства для вычислений, средства деловой графики, встроенные функции, средства для работы с базами данных, средства анализа табличных данных.

Текстовый процессор – вид прикладного программного обеспечения общего назначения, позволяющий создавать текстовые документы различной степени сложности – с автоматическим форматированием документов, вставкой рисованных объектов и графики, составлением оглавлений и указателей, проверкой орфографии, шрифтовым оформлением, подготовкой шаблонов документов и т. д.

Транзакция – совокупность операций манипулирования данными (вставки, удаления, выборки, обновления) в системе баз данных, которая переводит базу данных из одного целостного состояния в другое. В соответствии с классической моделью транзакция как единое целое: либо все операции выполняются, и результаты фикс-

сируются в базе данных, либо транзакции полностью аннулируются в базе данных (откат транзакции). В обоих случаях логическая целостность в базе данных не нарушается.

Утилиты – это вспомогательные программы, обеспечивающие сервис пользователю при работе на ЭВМ. Служебные программы, которые предоставляют пользователю ряд дополнительных услуг.

FAT (File Allocation Table) – это таблица размещения файлов (структура данных на диске, поддерживаемая операционной системой для отслеживания состояния различных сегментов дискового пространства).

Ядро операционной системы – совокупность модулей, выполняющих базовые функции операционной системы: управление процессами, памятью, устройствами ввода-вывода и т. д. Одним из определяющих свойств ядра является работа в привилегированном режиме.

Язык запросов (ЯЗ) – языковые средства СУБД, ориентированные на доступ к базе данных конечного пользователя.

Языковые средства СУБД – комплекс средств, обеспечивающих интерфейс пользователей разных категорий с базой данных. Набор языковых средств СУБД можно классифицировать по разным признакам. Например, по функциональному назначению выделяют языки определения данных, языки манипулирования данными, языки запросов.

Глава 4

Вербальная модель – это представление информационной модели средствами естественного разговорного языка.

Знаковые (символические) модели – это модели, в которых используются алфавиты формальных языков (условные знаки, специальные символы, буквы, цифры) и сформулированы правила оперирования с этими знаками. Примеры: специальные языковые системы, физические или химические формулы, математические выражения и формулы. К знаковым относятся математические и логические модели, программные тексты.

Информационная модель – это модель, с помощью которой описываются и изучаются свойства и состояние объекта, процесса или явления, а также их связи и отношения с окружающим миром.

Материальная модель – это материальная копия объекта моделирования.

Математическая модель – это способ представления информационной модели, отображающий связь различных параметров объекта через математические формулы и понятия.

Компьютерная модель – это созданный за счет ресурсов компьютера виртуальный образ, качественно и количественно отражающий внутренние свойства и связи моделируемого объекта, иногда передающий и его внешние характеристики.

Моделирование – исследование каких-либо явлений, процессов или систем объектов путем построения и изучения их моделей. Модели используются для определения или уточнения характеристик и рационализации способов построения вновь конструируемых объектов. Моделирование – одна из основных категорий теории познания: на идее моделирования по существу базируется любой метод научного исследования – как теоретический (при котором используются различного рода знаковые, абстрактные модели), так и экспериментальный (использующий предметные модели).

Модель – это 1) упрощенный образ оригинала, неразрывно с ним связанный, отражающий существенные свойства, связи и отношения оригинала; 2) мысленно представляемая или материально реализуемая система, исследование которой дает новую или подтверждает имеющуюся информацию об оригинале.

Мысленная (интуитивная) модель – это мысленное представление об объекте.

Модель жизненного цикла – структура, состоящая из процессов, работ и задач, включающих в себя разработку, эксплуатацию и сопровождение программного продукта, охватывающая жизнь системы от установления требований к ней до прекращения ее использования.

Модель представления интерфейса – это описание внешнего вида интерфейса с точки зрения конечного пользователя системы. Это может быть документ, показывающий внешний вид экрана или структуру отчета, или экран (отчет), созданный с помощью одного из средств визуальной разработки приложений – так называемых языков четвертого поколения (4GL – Fourth Generation Languages). Так как большинство языков 4GL позволяют быстро создавать работающие прототипы приложений, пользователь имеет

возможность увидеть работающий прототип системы на ранних стадиях проектирования.

Образная модель – это выражение свойств оригинала с помощью образов.

Образно-знаковые модели – это модели, построенные с использованием знаковых образов какого-либо вида: схем, чертежей, графов, графиков, планов, карт.

Объект – это то, что имеет название, является чем-то целым и обладает некоторыми свойствами или характеристиками.

Свойства объекта – это присущие ему качества, характеристики, признаки.

Система – это совокупность взаимодействующих между собой составных частей (элементов), составляющих единое целое, выделенных из окружающей среды и функционирующих с определенной целью.

Формализация – процесс построения информационных моделей с помощью формальных языков.

Эксперимент – это опыт, который производится с объектом или его моделью.

Глава 5

Алгоритм – это конечная последовательность точно определенных действий, приводящих от изменяемых исходных данных к решению поставленной задачи (конкретному результату).

Библиотека подпрограмм – это часть системы программирования, предназначенная для хранения часто используемых в процессе программирования стандартных подпрограмм.

Библиотека процедур – это вид вспомогательных модулей операционной системы, упрощающих разработку приложений, например, библиотека математических функций.

Библиотека функций, процедур – это вид инструментальных средств пользователя, представляющий совокупность подпрограмм, в результате выполнения которых формируются некоторые значения и вызов которых может быть использован многократно в качестве операндов в выражениях (в функциях) или по имени процедуры.

Дискретность алгоритма – это его свойство, проявляющееся в разбиении процесса, предусмотренного алгоритмом, на отдельные этапы (элементарные операции).

Массовость алгоритма – это его способность обеспечить решение любого, сколько угодно большого количества однотипных задач с различными исходными данными.

Определенность алгоритма – это его способность быть однозначным, исключая произвольность толкования любого из предписаний и заданного порядка исполнения.

Результативность алгоритма – это его способность выдать ожидаемые результаты или сообщение о невозможности решения задачи.

Запуск программ – процесс загрузки в основную память команд и данных, инициализации устройств ввода-вывода и файлов, а также подготовки других ресурсов, выполняемый операционной системой.

Инструментарий технологии программирования – 1) вид общего программного обеспечения, сопровождающий технологию разработки и внедрения программных средств; 2) совокупность программных средств, обеспечивающих технологию разработки и внедрения программных систем.

Макрокоманда – предложение языка, идентифицирующее набор простейших команд. Макрокоманды заменяются макрорасширениями (последовательностью команд) в результате процесса макрогенерации.

Макропрограммирование – вид инструментальных средств пользователя, предназначенный для разработки программ с использованием макрокоманд.

Отладчик – часть системы программирования, представляющая собой обрабатывающую программу, предназначенную для анализа поведения другой программы, обеспечивающая ее трассировку (отслеживание выполнения программы в пооператорном варианте), останов в указанных точках или при выполнении указанных условий (например, для идентификации места и вида ошибок), просмотр и изменение ячеек памяти, регистров процессора и команд программы.

Программа – упорядоченная последовательность команд, которые должен выполнить компьютер в соответствии с алгоритмом решения конкретной задачи (один из способов описания алгоритмов).

Редактор связей – это часть системы программирования, представляющая собой обрабатываемую программу, которая используется для построения одного загрузочного модуля из одного или более объектных или загрузочных модулей, то есть модулей, готовых для выполнения на ЭВМ.

Система программирования – это вид локальных средств разработки программ, представленных комплексом инструментов, предназначенных для автоматизации процессов программирования; включает в себя: язык программирования, транслятор, интегрированную среду разработки, набор вспомогательных средств для подготовки программ к выполнению (редактор связей, отладчик, библиотеки подпрограмм, справочную систему и др.).

Справочная система – это часть системы программирования (любой программной системы), предназначенная для представления справочной информации о ее возможностях, конструкциях операторов языка программирования, режимах работы, выходах из ошибочных ситуаций и т. д.

Средства автоматизированного создания информационных систем, CASE-технология (Compute-Aided Software/System Engineering) – вид инструментария технологии программирования, представляющий собой методы анализа, проектирования и создания программных систем и предназначенный для автоматизации процессов разработки и реализации информационных систем.

Транслятор – часть системы программирования, представляющая собой обрабатываемую программу, которая выполняет преобразование текста исходной программы (исходного модуля) с языка программирования в машинные коды (объектный модуль). Трансляторы бывают двух видов – компиляторы и интерпретаторы.

Язык программирования – часть системы программирования, представляющая собой формализованный язык для написания программ и записи алгоритмов решения задач на ЭВМ. В языках программирования строго определены алфавит, синтаксис и семантика.

Языки программирования высокого уровня – это языки, которые ориентированы на человека (похожи на естественный), более близки и понятны ему.

Языки программирования низкого уровня (ассемблеры) – это языки, которые ориентированы на конкретный тип процессора и учитывают его особенности.

Язык четвертого поколения (4GL – Forth Generation Language) – интерактивное средство программирования, используемое, в частности, для взаимодействия пользователя с СУБД или генераторами приложений.

Глава 6

Брандмауэр – это совокупность аппаратных средств и программного обеспечения, которая связывает две и больше сетей и одновременно разделяет защищенную и незащищенную сети или защищенную область сети от незащищенной области той же сети.

Брандмауэр сетевого уровня – это обычно экранирующий маршрутизатор или специальный компьютер, который исследует адреса пакетов, чтобы определить, передать пакет в локальную сеть или из нее, либо отклонить его.

Брандмауэр уровня приложения – это главный компьютер сети, использующий программное обеспечение сервера-посредника.

Брандмауэр уровня соединения подобен брандмауэру уровня приложения – оба они являются серверами-посредниками.

СОМР – это одна из рубрик телеконференций, в рамках которой выделяются темы, связанные с компьютером.

Двусторонний главный брандмауэр использует две отдельные сетевые карты. Это простая, но обеспечивающая очень высокую степень безопасности конфигурация, в которой один главный компьютер играет роль разделительной линии между локальной сетью и интернетом. Такой брандмауэр использует набор серверов-посредников уровня приложений или уровня соединения.

HTML (HyperText Markup Language) – гипертекстовый язык разметки документов для публикации в WWW. Стандарт языка HTML разработан и поддерживается консорциумом W3C. Язык HTML служит не для разметки содержания, а для форматной разметки документов. Разметка осуществляется с помощью предусмотренного в языке набора специальных меток, называемых тегами, каждый тег

определяет некоторые характеристики разметки и область их действия в документе. Важную роль в языке играют гиперссылки – бинарные связи между документами и/или их составными частями. Язык HTML позволяет интегрировать в HTML-документы информационные структуры нетекстовой природы – графику, аудио, видео и др.

Internet Explorer – это программа-браузер, позволяющая загружать веб-страницы по протоколу HTTP и файлы по протоколу FTP.

Протокол ICQ (произносится как «I Seek You») – это набор правил, по которому предоставляется услуга, обеспечивающая ведение личной переписки через сеть Интернет посредством собственной службы мгновенного обмена сообщениями, компьютерного клиента для этой службы, а также интернет-портала.

Интернет-служба E-mail (Electronic MAIL) – это электронная почта, обеспечивающая пересылку писем.

Интернет-служба FTP (File Transfer Protocol) – это служба передачи файлов, обеспечивающая их пересылку.

Интернет-служба DNS (Domain Name System) обеспечивает назначение доменных имен и устанавливает их соответствие числовым IP-адресам.

Коммутатор – это устройство, коммутирующее несколько каналов связи.

Концентратор – это устройство, коммутирующее несколько каналов связи на один канал путем частотного разделения.

Модем – это устройство, выполняющее модуляцию и демодуляцию информационных сигналов при их передаче и приеме.

Мост – это 1) устройство, соединяющее две сети, использующие одинаковые методы передачи данных; 2) компьютер, имеющий две сетевые карты и предназначенный для соединения сетей или сегментов локальной сети.

NEWS – это одна из рубрик телеконференций, в рамках которой рассматриваются и обсуждаются новости.

Операционная система сетевая – это операционная система, выполняющая все функции локальной операционной системы, а также обладающая дополнительными средствами, позволяющими ей взаимодействовать по сети с операционными системами других компьютеров. Основными функциональными компонентами сете-

вой операционной системы являются средства управления локальными ресурсами компьютера и сетевые средства.

Пакеты – это наиболее эффективный способ коммуникации для передачи компьютерного трафика.

Протоколы – это специальные стандарты (правила) для обеспечения необходимой совместимости в компьютерной сети.

Повторитель – это устройство, обеспечивающее сохранение формы и амплитуды сигнала при передаче его на большее, чем предусмотрено данным типом физической передающей среды, расстояние.

Сетевые средства – компонент сетевой операционной системы, реализующий ее сетевые функции.

Сервер баз данных – вид прикладного программного обеспечения общего назначения, используемый для создания и доступа при работе в сети к интегрированным базам данных в архитектуре клиент-сервер. Сервер баз данных всю обработку данных выполняет самостоятельно, одновременно обеспечивая данными большое число пользователей сети (Oracle, Microsoft SQL Server и др.).

Сервер – это компьютер, предоставляющий свои ресурсы другим компьютерам (клиентам) при совместной работе в сети.

Сервер сети – это компьютер, предоставляющий доступ к сетевым ресурсам.

Сервер-посредник – это программа, управляющая трафиком и контролирующая передачу данных между двумя сетями, обеспечивая связь между пользователями локальной сети и серверами внешней сети.

Сетевые экраны (Firewall) – это наиболее эффективные средства для защиты от сетевых атак.

SKYPE – бесплатное проприетарное (т. е. являющееся частной собственностью) программное обеспечение с закрытым кодом, обеспечивающее шифрованную голосовую связь через интернет между компьютерами.

SOC – это одна из рубрик телеконференций, для рассмотрения социальной тематики.

SCI – это одна из рубрик телеконференций, для обсуждения различных тем из области научных исследований.

SMTP (Simple Mail Transfer Protocol) – это основной сетевой протокол, предназначенный для передачи электронной почты в сетях TCP/IP.

Топология сети – это усредненная геометрическая схема соединений узлов сети.

The Bat – это программа (почтовый клиент), позволяющая загружать и редактировать электронную почту.

Telnet – это программа (сетевой теледоступ) для работы с удаленным компьютером.

Топология (от греч. topos – место и ...логия) – это раздел математики, изучающий топологические свойства фигур, т. е. свойства, не изменяющиеся при любых деформациях, производимых без разрывов и склеиваний (точнее, при взаимно однозначных и непрерывных отображениях).

UseNet (USEr NETwork) – это система обмена текстовой информацией, позволяющей всем пользователям интернета участвовать в групповых дискуссиях.

Фильтрующий главный брандмауэр использует комбинацию маршрутизаторов и серверов-посредников.

FTP-сервер – это компьютер, на котором хранятся файлы, предназначенные для открытого доступа.

Шлюз (gateway) – аппаратный маршрутизатор или программное обеспечение для сопряжения компьютерных сетей, использующих разные протоколы (например, локальной и глобальной).

Электронно-цифровая подпись – это реквизит электронного документа, позволяющий установить истинность его отправителя.

Глава 7

Межсетевой экран (брандмауэр) – метод защиты сети от угроз безопасности, исходящих от других систем и сетей, с помощью централизации доступа к сети и контроля за ним аппаратно-программными средствами.

Компьютер-бастион – это компьютер, специально установленный для защиты от атак на локальную сеть.

Компьютерный вирус – программа, способная к саморазмножению, к самомодификации в работающей вычислительной среде и вызывающая нежелательные для пользователя действия: нарушение работы программ; вывод на экран монитора посторонних

сообщений, символов, изображений; порчу и/или невозможность прочтения записей, как отдельных файлов, так и дисков в целом; замедление работы компьютера и т. д.

Логические бомбы – это программы или их части, постоянно находящиеся в ЭВМ или вычислительных системах и выполняющиеся только при соблюдении определенных условий.

Троянские кони – это программы, полученные путем явного изменения или добавления команд в пользовательские программы.

Угроза безопасности – потенциально возможное событие, процесс или явление, которые могут привести к нарушению основных качественных характеристик программного обеспечения, таких как функциональные возможности, надежность, эффективность.

Угроза безопасности информации – это возможное событие, процесс или явление, которое может привести к уничтожению, утрате целостности, конфиденциальности или доступности информации.

Черви – это программы, которые выполняются каждый раз при загрузке системы, обладают способностью перемещаться в вычислительных системах или в сети и самовоспроизводить свои копии.

Экранирующий маршрутизатор (screening router) – это специальный компьютер (созданный, как правило, на базе персонального компьютера) или электронное устройство, экранирующее (фильтрующее) пакеты данных, основываясь на заданных администратором критериях.

Оглавление

| | |
|---|-----------|
| Предисловие | 3 |
| Введение | 4 |
| Глава 1. Понятие информации, общая характеристика процессов сбора, передачи, обработки и накопления информации | 6 |
| 1.1. Понятие данных и информации | 6 |
| 1.2. Общая характеристика процессов сбора, передачи, обработки и накопления информации | 6 |
| 1.3. Виды и типы данных | 7 |
| 1.4. Алгебра логики | 8 |
| 1.4.1. Основные логические операции | 9 |
| 1.4.2. Свойства операций | 10 |
| 1.4.3. равносильные логические выражения | 11 |
| 1.4.4. Основные законы логики | 14 |
| 1.5. Позиционные системы счисления. Основные определения | 14 |
| 1.5.1. Некоторые понятия, вопросы и ответы | 17 |
| 1.5.2. Алгоритм получения дополнительного k -разрядного кода отрицательного числа | 18 |
| 1.6. Кодирование данных двоичным кодом | 21 |
| 1.6.1. Кодирование целых и вещественных чисел | 21 |
| 1.6.2. Кодирование текстовых данных | 22 |
| 1.6.3. Кодирование графических данных | 23 |
| 1.6.4. Кодирование звука | 24 |
| 1.7. Основные структуры данных | 24 |
| 1.8. Единицы представления, измерения, хранения и передачи данных | 26 |
| 1.9. Контрольные вопросы | 27 |
| Глава 2. Технические средства реализации информационных процессов | 29 |
| 2.1. Аппаратная конфигурация вычислительной системы | 29 |
| 2.2. Базовая аппаратная конфигурация компьютера | 30 |
| 2.3. Контрольные вопросы | 34 |
| Глава 3. Программные средства реализации информационных процессов | 35 |
| 3.1. Программная конфигурация вычислительной системы | 35 |
| 3.2. Операционные системы персональных компьютеров | 39 |

| | |
|---|----|
| 3.2.1. Общие сведения об операционных системах | 39 |
| 3.2.2. Файловая структура операционных систем | 40 |
| 3.2.3. Базовые функции операционных систем | 42 |
| 3.2.4. Прочие функции операционных систем | 44 |
| 3.2.5. Контрольные вопросы | 45 |
| 3.3. Электронные таблицы MS Excel | 45 |
| 3.3.1. Назначение и возможности электронных таблиц | 45 |
| 3.3.2. Главное окно MS Excel | 47 |
| 3.3.3. Настройки и параметры MS Excel | 49 |
| 3.3.4. Общие принципы работы с книгами | 51 |
| 3.3.4.1. Создание и сохранение книг | 51 |
| 3.3.5. Работа с листами | 53 |
| 3.3.6. Работа с фрагментами данных | 55 |
| 3.3.7. Ввод данных в ячейки | 58 |
| 3.3.8. Автозаполнение | 59 |
| 3.3.8.1. Заполнение числами | 59 |
| 3.3.8.2. Заполнение датами | 61 |
| 3.3.8.3. Списки данных | 61 |
| 3.3.9. Работа с формулами | 62 |
| 3.3.9.1. Простые выражения | 62 |
| 3.3.9.2. Копирование формул | 63 |
| 3.3.10. Ссылки | 64 |
| 3.3.11. Встроенные функции | 67 |
| 3.3.12. Форматирование рабочих листов | 69 |
| 3.3.12.1. Форматирование числовых значений | 71 |
| 3.3.12.2. Форматирование символов в ячейках | 73 |
| 3.3.12.3. Перенос строк и ориентация текста | 73 |
| 3.3.12.4. Использование границ и рамок | 74 |
| 3.3.13. Диаграммы | 75 |
| 3.3.13.1. Гистограммы | 76 |
| 3.3.13.2. Круговые диаграммы | 76 |
| 3.3.13.3. Графики | 77 |
| 3.3.14. Печать содержимого книги | 79 |
| 3.3.15. Коды возможных ошибок при работе с формулами в MS Excel | 81 |
| 3.3.16. Примеры | 82 |
| 3.3.17. Контрольные вопросы | 85 |
| 3.4. Текстовый процессор MS Word | 85 |

| | |
|---|-----|
| 3.4.1. Главное окно MS Word | 86 |
| 3.4.2. Настройки и параметры MS Word | 87 |
| 3.4.3. Создание и сохранение документов | 92 |
| 3.4.4. Параметры страницы | 95 |
| 3.4.5. Общие принципы работы с документами | 96 |
| 3.4.6. Средства поиска и замены | 98 |
| 3.4.7. Изменение параметров шрифта | 100 |
| 3.4.8. Настройки абзаца | 102 |
| 3.4.9. Вставка символов | 104 |
| 3.4.10. Колонки | 106 |
| 3.4.11. Списки | 108 |
| 3.4.11.1. Маркированный список | 108 |
| 3.4.11.2. Нумерованный список | 108 |
| 3.4.11.3. Многоуровневый список | 109 |
| 3.4.12. Вставка и редактирование формул | 110 |
| 3.4.13. Создание и редактирование таблиц | 112 |
| 3.4.14. Графика в документах MS Word | 116 |
| 3.4.14.1. Создание рисунков | 119 |
| 3.4.14.2. Перемещение и копирование рисунков | 120 |
| 3.4.15. Диаграммы | 121 |
| 3.4.15.1. Построение диаграмм | 122 |
| 3.4.16. Форматирование текста с помощью стилей | 122 |
| 3.4.17. Колонтитулы и нумерация страниц | 124 |
| 3.4.18. Оглавление | 125 |
| 3.4.19. Печать документов | 126 |
| 3.4.20. Контрольные вопросы | 127 |
| 3.5. Средства создания и использования презентаций MS PowerPoint | 128 |
| 3.5.1. Основные возможности приложения MS PowerPoint | 129 |
| 3.5.1.1. Главное окно | 129 |
| 3.5.1.2. Обычный режим | 130 |
| 3.5.1.3. Режим сортировщика слайдов | 131 |
| 3.5.1.4. Показ слайдов | 131 |
| 3.5.2. Настройки и параметры MS PowerPoint | 131 |
| 3.5.3. Общие принципы работы с презентациями | 132 |
| 3.5.3.1. Создание и сохранение презентаций | 132 |
| 3.5.4. Оформление слайдов | 133 |
| 3.5.5. Работа со слайдами | 135 |

| | |
|--|-----|
| 3.5.5.1. Основной показ | 135 |
| 3.5.5.2. Произвольные показы | 136 |
| 3.5.5.3. Демонстрация произвольного показа | 138 |
| 3.5.6. Добавление текста | 138 |
| 3.5.7. Вставка таблиц | 140 |
| 3.5.8. Добавление клипов | 142 |
| 3.5.9. Диаграммы | 144 |
| 3.5.10. Анимация объектов презентации | 147 |
| 3.5.11. Показ презентаций | 150 |
| 3.5.11.1. Типы презентаций | 150 |
| 3.5.11.1.1. Презентации на экране | 150 |
| 3.5.11.1.2. Интерактивные презентации | 151 |
| 3.5.11.1.3. Смена слайдов презентации | 152 |
| 3.5.11.1.4. Показ слайдов на втором мониторе | 154 |
| 3.5.12. Упаковка презентаций для переноса | 154 |
| 3.5.13. Вывод слайдов презентации на печать | 155 |
| 3.5.13.1. Цветная печать, черно-белая печать и печать с оттенками серого | 155 |
| 3.5.13.2. Слайды и прозрачки | 156 |
| 3.5.13.3. Структуры страниц заметок и раздаточного материала | 156 |
| 3.5.14. Технология создания и использования презентаций | 157 |
| 3.5.15. Контрольные вопросы | 160 |
| 3.6. Средства создания и обработки компьютерной графики | 161 |
| 3.6.1. Виды компьютерной графики | 161 |
| 3.6.2. Средства для работы с компьютерной графикой | 163 |
| 3.6.2.1. Средства для работы с растровой графикой | 163 |
| 3.6.2.2. Средства создания и обработки векторной графики | 164 |
| 3.6.2.3. Программные средства обработки трехмерной графики | 165 |
| 3.6.3. Представление графических данных | 166 |
| 3.6.4. Контрольные вопросы | 167 |
| 3.7. Базы данных | 168 |
| 3.7.1. Основные свойства полей таблиц | 169 |
| 3.7.2. Типы данных | 170 |
| 3.7.3. Основные типы объектов | 171 |
| 3.7.4. Работа с СУБД MS Access | 172 |
| 3.7.4.1. Создание таблиц | 172 |

| | |
|--|-----|
| 3.7.4.2. Создание межтабличных связей | 174 |
| 3.7.4.3. Создание запросов на выборку | 177 |
| 3.7.4.4. Создание запросов «с параметром» | 178 |
| 3.7.5. Контрольные вопросы | 179 |
| 3.8. Справочная система MS Office | 182 |
| Глава 4. Модели решения функциональных и вычислительных задач | 187 |
| 4.1. Понятие модели и объекта | 187 |
| 4.2. Классификация моделей | 188 |
| 4.3. Этапы компьютерного моделирования | 191 |
| 4.4. Объекты моделирования. Свойства объектов | 193 |
| 4.5. Системы объектов | 194 |
| 4.6. Некоторые разновидности моделей | 195 |
| 4.7. Некоторые ключевые понятия и примеры | 196 |
| Глава 5. Основы алгоритмизации и программирования | 198 |
| 5.1. Понятие алгоритма. Свойства и способы описания | 198 |
| 5.1.1. Графический способ описания | 199 |
| 5.1.2. Базовые конструкции алгоритмов | 200 |
| 5.1.3. Контрольные вопросы | 204 |
| 5.2. Программное обеспечение и технологии программирования | 206 |
| 5.2.1. Средства создания программ | 206 |
| 5.2.2. Алгоритмическое программирование | 207 |
| 5.2.3. Структурное программирование | 208 |
| 5.2.4. Объектно-ориентированное программирование | 209 |
| 5.2.5. Контрольные вопросы | 211 |
| 5.3. Языки программирования | 212 |
| 5.3.1. Языки программирования низкого уровня | 212 |
| 5.3.2. Высокоуровневые языки программирования | 213 |
| 5.3.3. Программирование баз данных | 215 |
| 5.3.4. Программирование для сети Интернет | 216 |
| 5.3.5. Контрольные вопросы | 217 |
| 5.4. Язык программирования Visual Basic for Applications | 217 |
| 5.4.1. Алфавит VBA | 218 |
| 5.4.2. Переменные и константы | 219 |
| 5.4.2.1. Область действия | 220 |
| 5.4.2.2. Объявление | 220 |
| 5.4.2.3. Время жизни | 221 |

| | |
|---|-----|
| 5.4.2.4. Типы переменных и констант | 222 |
| 5.4.3. Управляющие конструкции | 223 |
| 5.4.3.1. Операнды, операции, выражения | 223 |
| 5.4.3.2. Приоритет выполнения операций | 225 |
| 5.4.3.3. Встроенные функции | 225 |
| 5.4.3.4. Операторы альтернативы (ветвления) | 229 |
| 5.4.3.4.1. Условный оператор | 229 |
| 5.4.3.4.2. Оператор выбора | 231 |
| 5.4.3.5. Операторы циклов | 232 |
| 5.4.3.5.1. Циклы с предусловием | 232 |
| 5.4.3.5.2. Циклы с постусловием | 233 |
| 5.4.3.5.3. Цикл по счетчику | 234 |
| 5.4.3.5.4. Вложенные циклы | 235 |
| 5.4.3.6. Подпрограммы-процедуры и подпрограммы-функции | 236 |
| 5.4.3.7. Ввод-вывод с помощью диалоговых окон | 238 |
| 5.4.3.8. Массивы | 239 |
| 5.4.3.8.1. Статические массивы | 239 |
| 5.4.3.8.2. Динамические массивы | 242 |
| 5.4.4. Основные объекты MS Excel | 244 |
| 5.4.4.1. Объект Workbook и семейство Workbooks | 244 |
| 5.4.4.1.1. Свойства | 245 |
| 5.4.4.1.2. Методы | 246 |
| 5.4.4.1.3. События | 247 |
| 5.4.4.2. Объект Worksheet и семейство Worksheets | 247 |
| 5.4.4.2.1. Свойства | 247 |
| 5.4.4.2.2. Методы | 249 |
| 5.4.4.3. Объект Range | 249 |
| 5.4.4.3.1. Свойства | 251 |
| 5.4.4.3.2. Методы | 253 |
| 5.4.4.4. Форматирование рабочего листа | 254 |
| 5.4.4.4.1. Оформление текста в ячейке | 254 |
| 5.4.4.4.2. Оформление границ ячейки | 258 |
| 5.4.4.4.3. Заливка ячейки | 260 |
| 5.4.5. Контрольные вопросы | 261 |
| Глава 6. Локальные и глобальные компьютерные сети | 265 |
| Некоторые понятия и терминология | 268 |
| Глава 7. Основы защиты информации и сведений, составляющих государственную тайну | 272 |

| | |
|--|-----|
| 7.1. Информационная безопасность и ее составляющие | 272 |
| 7.2. Угрозы безопасности информации в компьютерных системах | 274 |
| 7.3. Методы защиты информации | 281 |
| 7.3.1. Профилактика заражения вирусами компьютерных систем | 286 |
| 7.3.2. Порядок действий пользователя при обнаружении заражения вирусами компьютерной системы | 287 |
| 7.3.3. Особенности защиты информации в базах данных | 289 |
| 7.3.4. Программно-технические средства защиты информации | 290 |
| 7.3.4.1. Защита компьютерных сетей с помощью брандмауэров | 292 |
| 7.4. Законодательные акты РФ, регулирующие правовые отношения в сфере информационной безопасности и защиты государственной тайны | 295 |
| 7.5. Контрольные вопросы | 300 |
| Заключение | 302 |
| Литература | 303 |
| Словарь основных терминов и сокращений | 305 |
| Оглавление | 323 |

Таганов Леонид Степанович
Пимонов Александр Григорьевич

Информатика

Учебное пособие

Редактор З. М. Савина

Подписано в печать 03.11.2010. Формат 60×84/16.

Бумага офсетная. Отпечатано на ризографе.

Тираж 250 экз. Уч.-изд. л. 20,6. Заказ № 45.

Кузбасский государственный технический университет,
650000, Кемерово, ул. Весенняя, 28.

Типография Кузбасского государственного технического университета,
650000, Кемерово, ул. Д. Бедного, 4А.



Пимонов
Александр Григорьевич

(<http://vtit.kuzstu.ru/staff.php?id=2>)
– родился 23 ноября 1959 года в селе Чапаево Хакасской автономной области, доктор технических наук, профессор, заведующий кафедрой вычислительной техники и информационных технологий. В 1981 г. с отличием закончил факультет прикладной математики и кибернетики Томского государственного университета. В Кузбасском государственном техническом университете работает с 1985 г. Подготовил около 120 научных работ и учебно-методических разработок. Ведет занятия по дисциплинам «Информатика», «Теория систем и системный анализ», «Информационные системы в экономике», «Информационные технологии управления», «Имитационное моделирование экономических систем».



Таганов
Леонид Степанович

(<http://vtit.kuzstu.ru/staff.php?id=12>)
– родился 8 июля 1936 года в деревне Семёновская Пучежского района Ивановской области, кандидат военных наук, полковник в отставке, доцент кафедры вычислительной техники и информационных технологий (<http://vtit.kuzstu.ru/news.php>). В 1961 г. с отличием и занесением на Доску почета закончил Киевское военное училище связи, в 1970 г. – с отличием Военную академию связи, в 1976 г. – адъюнктуру при Военной академии связи. Более 17 лет преподает информатику в Кузбасском государственном техническом университете (<http://www.kuzstu.ru>).