

Учреждение образования  
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Н. Н. Пустовалова, А. И. Бракович**

# **ИНФОРМАТИКА И КОМПЬЮТЕРНАЯ ГРАФИКА**

**Курс лекций  
для студентов всех специальностей**

Минск 2010

УДК 004.92(042.4)  
ББК 72я73  
П89

Рассмотрен и рекомендован к изданию редакционно-издательским советом университета

Рецензенты:

кандидат технических наук,  
доцент кафедры информационных технологий  
автоматизированных систем БГУИР *О. В. Герман*;  
кандидат физико-математических наук, доцент кафедры прикладной  
математики и экономической кибернетики БГЭУ *И. В. Кашиникова*

**Пустовалова, Н. Н.**

П89 Информатика и компьютерная графика : курс лекций для студентов всех специальностей / Н. Н. Пустовалова, А. И. Бракович. – Минск : БГТУ, 2010. – 204 с.  
ISBN 978-985-434-949-7.

Курс лекций содержит достаточно обширный материал, который может быть применен студентами в учебной и научно-исследовательской работе. Приведены основные принципы работы с приложениями Microsoft Office и освещены способы расширения возможностей этих приложений путем использования языка Visual Basic for Application. В работе кратко изложены численные методы и их реализация в Excel и Mathcad. Основы работы с такими программными средствами, как Paint, Microsoft Visio, Adobe Photoshop, Macromedia Flash, позволят студентам использовать компьютерную графику, а способы создания web-страниц дадут понятие о принципах организации информации в сети Интернет.

**УДК 004.92(042.4)  
ББК 72я73**

**ISBN 978-985-434-949-7**

© УО «Белорусский государственный  
технологический университет», 2010  
© Пустовалова Н. Н., Бракович А. И., 2010

## ВВЕДЕНИЕ

В инженерной деятельности компьютер используется достаточно широко. Целью курса «Информатика и компьютерная графика», разработанного в соответствии с образовательными стандартами Республики Беларусь для соответствующих специальностей, является изучение методов подготовки и решения задач на современных компьютерах, способов работы с пакетами прикладных программ.

Грамотный специалист должен уметь работать с приложениями Microsoft Office, знать современные технические средства взаимодействия с персональными компьютерами, основы алгоритмизации прикладных задач, методы разработки алгоритмов и составления программ на языках программирования.

Приложения, входящие в пакет Microsoft Office: текстовый редактор Word, электронные таблицы Excel, системы управления базами данных Access, – позволяют программировать задачи на соответствующих версиях языка Visual Basic. Этот язык является системой программирования в операционной среде Windows и предназначен для создания прикладных программ, решающих конкретные задачи в режиме диалога с пользователем. Разработка проекта в Windows складывается из двух последовательных этапов:

- сначала размещаются на экране элементы управления в виде значков, пиктограмм, кнопок, вложенных окон и т. п. для организации необходимого графического пользовательского интерфейса прикладной программы (приложения) на экране компьютера;

- затем для определенных элементов управления интерфейса разрабатывается программный код, по которому будет выполняться работа.

Такое программирование называется *визуально-событийным*. Программа, как правило, связывается с определенным элементом управления графического интерфейса на экране (кнопкой, окном и т. п.) и начинает работать тогда, когда произойдет связанное с этим элементом *событие*: щелчок мышью, движение мышью, нажатие определенной клавиши на клавиатуре и т. п.

В данной работе рассматриваются способы программирования на языке VBA (Visual Basic for Application) для приложения Excel. Если приложение Excel установлено на компьютере в полном объеме, то язык VBA входит в его состав, что позволяет значительно расширить

возможности приложения при использовании его в инженерной практике и научных исследованиях.

Решение задач инженерно-технического и научно-исследовательского характера требует знания различных методов вычислительной математики и современных программных средств при их применении. Выполняя соответствующие задания, студенты изучают основные численные методы, используемые при решении задач: методы вычисления определенных интегралов, приближенного решения уравнений, решения систем алгебраических и дифференциальных уравнений, методы получения математических моделей и оптимизации процессов, а также приобретают навыки решения вышеперечисленных задач в Excel (при необходимости используя элементы программирования на языке VBA) и Mathcad.

В настоящее время компьютерная графика проникла практически во все сферы применения персонального компьютера. Она используется при подготовке печатных документов, для визуализации расчетов, в системах компьютерного моделирования, в мультимедийных программах (электронных учебниках, энциклопедиях, играх), и, конечно, огромное количество компьютерной графики можно встретить на web-страницах в Интернете. Поэтому инженер должен уметь использовать компьютерную графику в своей практической деятельности.

В пособии рассматриваются такие программные средства для работы с компьютерной графикой, как Paint, Microsoft Visio, Adobe Photoshop, Macromedia Flash. И, наконец, в данной работе излагаются основные принципы функционирования сети Интернет и способы создания простых web-страниц.

# 1. АРХИТЕКТУРА И ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ КОМПЬЮТЕРА

## 1.1. Обобщенная схема компьютера

В настоящее время существуют самые разные модели вычислительных машин: маленькие карманные персональные компьютеры, ноутбуки, настольные персональные компьютеры, рабочие станции, суперкомпьютеры и т. д.

Основные функциональные блоки компьютера представлены на обобщенной схеме (рис. 1.1).

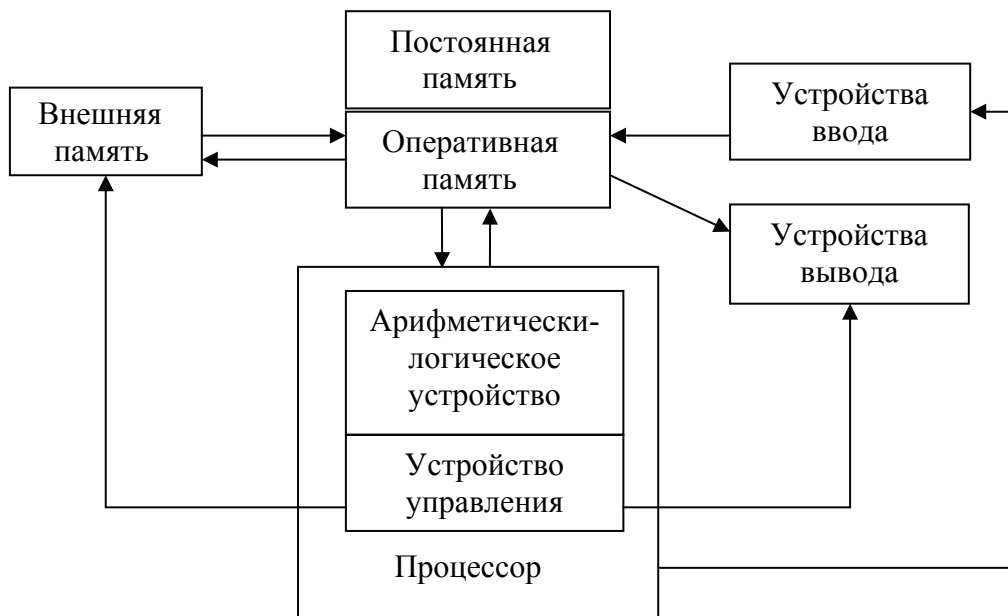


Рис. 1.1. Обобщенная схема компьютера

Оперативная память состоит из байтов, каждый из которых, в свою очередь, состоит из битов. Бит может принимать два значения: 0 и 1. Сочетанием этих цифр в компьютере записывается вся информация. В оперативной памяти информация хранится временно, для более длительного ее хранения служит постоянная память на жестких дисках.

Процессор осуществляет операции над данными и управляет взаимодействием всех устройств компьютера.

Если информацию необходимо перенести на другой компьютер, то можно использовать внешнюю память, в частности, такие носители информации, как дискеты, компакт-диски, USB-флешки.

К устройствам ввода относится клавиатура, различные виды сканеров, микрофоны и др.

Устройства вывода – это мониторы, различные виды принтеров, графопостроителей, музыкальные колонки и пр.

## 1.2. Операционные системы

Операционная система (ОС) – это комплекс взаимосвязанных системных программ, которые организуют взаимодействие пользователя с компьютером и обеспечивают выполнение всех других программ.

В различных моделях компьютеров используют операционные системы с разной архитектурой и возможностями. Для их работы требуются разные ресурсы. Они предоставляют различную степень сервиса для программирования и работы с готовыми программами.

В настоящее время наибольшее распространение получили операционные системы Windows. Основные их достоинства по сравнению с ранее использовавшимися средствами: красивый интерфейс; многозадачность; возможность использования буфера обмена; многообразие шрифтов для написания текста; реализация механизма связи и внедрения объектов.

Различные версии операционных систем семейства Windows используются пользователями в настоящее время: Windows 98, Windows 2000, Windows XP, Windows Vista, Windows 7.

Операционная система Windows предоставляет пользователю удобные средства организации работы на компьютере.

**Работа с мышкой и окнами.** Операция «Щелчок» – одинарный щелчок левой клавишей мыши.

Операция «Двойной щелчок» – двойной щелчок левой клавишей мыши. Для того чтобы запустить программу, надо поместить курсор мыши на ее пиктограмму и сделать двойной щелчок.

Операция «Протащить» – курсор мыши помещается в нужное место, нажимается левая клавиша, и, не отпуская ее, перетаскивается курсор в нужное место. Используется для выделения и переноса.

При нажатии правой клавиши мыши появляется *контекстное меню*, которое относится к тому элементу, на котором находится курсор мыши.

В правом верхнем углу каждого окна имеются кнопки. Кнопка с изображением черты позволяет свернуть окно и разместить его в нижней части экрана на панели задач. Кнопка с квадратом служит для

разворачивания окна до полного экрана, при этом появляются два квадрата, и нажатие на такую кнопку позволяет восстановить первоначальное положение окна. Кнопка с крестиком закрывает окно.

Чтобы изменить размер окна, надо поместить курсор на границу окна, при этом он примет вид двойной стрелки. Затем следует протащить границу на нужное место и отпустить левую клавишу. Чтобы изменить положение окна, надо подвести курсор к заголовку окна и перетащить окно на нужное место.

Для занесения содержимого экрана в буфер обмена для дальнейшего использования служит клавиша **<Print Screen>** (**<PrtSc>**). Если требуется поместить в буфер обмена содержимое активного окна, то необходимо одновременно нажать на клавиши **<Alt>** и **<PrtSc>**. Для вставки содержимого буфера в документ, программу или рисунок в приложениях используется команда **Вставить** из пункта меню **Правка**.

**Создание, открытие и сохранение файлов.** Для создания нового файла надо в приложении выполнить команду **Файл/Создать**. Чтобы открыть уже имеющийся файл, следует выполнить команду **Файл/Открыть** и в появившемся окне выбрать нужный файл. Для сохранения файла первый раз можно использовать кнопку **Сохранить** на панели инструментов (если таковая имеется в приложении) либо команду **Сохранить** или **Сохранить как...** из пункта меню **Файл**. В появившемся диалоговом окне выбирается устройство (диск), папка, задается имя файла и нажимается **<ОК>**. Для повторного сохранения файла под прежним именем используется кнопка или команда **Сохранить** из пункта меню **Файл**, а для сохранения файла под другим именем или в другом месте – команда **Сохранить как...**

В приложениях Microsoft Office 2007 вышеперечисленные команды (**Создать**, **Открыть**, **Сохранить**, **Сохранить как...**) содержатся в окне, появляющемся после нажатия кнопки **Office**.

**Копирование, перемещение, вставка.** Во всех приложениях операции копирования, перемещения и вставки могут выполняться несколькими способами. Чтобы *скопировать информацию*, надо ее выделить и выбрать один из способов копирования:

- выполнить команду **Правка/Копировать**;
- нажать кнопку **Копировать** на панели инструментов;
- выполнить команду **Копировать** из контекстного меню;
- нажать «горячие» клавиши **<Ctrl> + <C>**.

Затем следует поместить курсор в место вставки информации и выполнить команду **Правка/Вставить** либо команду **Вставить** из контекстного меню либо нажать кнопку **Вставить** на панели

инструментов. Можно также использовать «горячие» клавиши **<Ctrl> + <V>**.

Чтобы *переместить информацию*, ее надо выделить и вырезать, используя одну из следующих команд: **Правка/Вырезать**, кнопку **Вырезать** на панели инструментов, команду **Вырезать** из контекстного меню или «горячие» клавиши **<Ctrl> + <X>**. Затем информацию следует вставить в нужное место.

Операцию перемещения можно выполнить и с помощью мыши. Для этого надо выделить информацию, поместить на выделенный участок мышь, нажать левую клавишу мыши и, не отпуская ее, протащить курсор в нужное место. Если при этом нажать клавишу **<Ctrl>**, то будет выполнена операция копирования.

**Поиск файла.** Для поиска файла или папки средствами Windows надо нажать кнопку **Пуск**, выбрать **Найти**, затем **Файлы и папки**. В появившемся окне ввести имя или часть имени. При этом можно использовать символ «\*», который заменяет несколько неизвестных символов, или символ «?», который заменяет один неизвестный символ. Можно сузить границы поиска, указав дополнительные критерии, например, по дате создания, размеру файла и т. д.

### 1.3. Программы группы «Стандартные»

Запуск программ начинается с нажатия кнопки **Пуск** в левом нижнем углу экрана. Затем выбирается пункт меню **Программы** и далее соответствующая программа или группа программ.

В группе «Стандартные» содержатся некоторые популярные приложения: **Проводник**, **Калькулятор**, **Блокнот**, **Paint** и др.

**Создание рисунков в графическом редакторе Paint.** Рисунок создается на экране в поле, размеры которого можно изменять, зацепив мышкой за маркеры на границах. В левой части экрана находится панель инструментов: две кнопки для выделения изображения; кнопка **A** для создания надписей; кнопки для заливки и стирания; для рисования карандашом, кистью; для рисования геометрических фигур. Для детальной проработки рисунка можно воспользоваться кнопкой **Масштаб**.

Если рисовать левой кнопкой мыши, то используется основной цвет, а если правой – фоновый. Чтобы нарисовать горизонтальную или вертикальную линию или линию с наклоном 45°, надо нажать клавишу **<Shift>** и рисовать, не отпуская ее.

Ширина и цвет линий задается в нижней части экрана.



Графические объекты можно наклонять, растягивать, поворачивать и т. д., выполняя соответствующие команды из пункта меню **Рисунок**. Данный пункт меню активируется после выделения графического объекта.

Команды из пункта меню **Правка** (вырезать, копировать и т. д.) становятся активными после выделения соответствующих графических объектов. Команда **Отменить** последовательно отменяет выполненные операции.

Для создания другого рисунка необходимо выбрать команду **Создать** из пункта меню **Файл**.

**Приложение Проводник**. Это приложение может быть установлено на компьютере отдельно или может входить в группу «Стандартные». Оно предназначено для выполнения операций с дисками, папками и файлами, для связи с другими компьютерами в сети.

Команды из пункта меню **Вид** позволяют отсортировать имена папок и файлов в нужном порядке, представить их в удобном виде.

Каждый файл на диске имеет обозначение, которое состоит из двух частей: имени и расширения. В имени файла может быть от 1 до 256 символов. Расширение указывает на тип файла, оно записывается после точки и состоит из 3 или 4 символов. Например, **doc** – документ Word (версии 1997–2003 гг.); **exe** – выполнимый файл; **txt** – текстовый файл; **bmp** – точечный рисунок.

Для создания папки надо поместить курсор на имя диска или папки, внутри которой следует ее создать, выполнить команду **Файл/Создать/Папка**, ввести имя папки и нажать клавишу ввода.

При работе с файлами удобно использовать контекстное меню, команды которого отображают допустимые операции с файлами. Среди них надо отметить команды сжатия файлов, которые позволяют значительно сократить объем файлов.

Для перемещения файлов можно применять описанные выше способы, а также мышку, щелкнув по имени файла и удерживая ее, можно переместить файл в нужное место. Если при этом держать нажатой клавишу **<Ctrl>**, то файл копируется. Эти операции можно осуществлять над группой файлов, предварительно их выделив.

Для выделения смежных файлов надо щелкнуть мышкой по первому файлу и при нажатой клавише **<Shift>** щелкнуть по последнему. Для выделения несмежных файлов следует щелкнуть мышкой по первому файлу и при нажатой клавише **<Ctrl>** щелкать по другим файлам.

## 1.4. Группа программ Microsoft Office

**Microsoft Word** – текстовый процессор, который позволяет создавать документы любой сложности и оформлять их с использованием различных шрифтов, имеет удобный графический интерфейс и средства автоматизации оформления документов.

**Microsoft Excel** – программа представления и обработки данных в виде электронных таблиц. Excel имеет мощный аппарат математических инструментов для решения задач линейного программирования, оптимизации, статистического моделирования.

**Microsoft PowerPoint** – программа создания презентаций и слайд-фильмов, которая предоставляет пользователю возможности по оформлению текста, рисованию, построению диаграмм, а также широкий набор стандартных иллюстраций, стилевых шаблонов и возможности использования звука и видео.

**Microsoft Access** – программа для создания и редактирования баз данных.

**Microsoft FrontPage** – программа для создания и дизайна страниц Интернет.

**Microsoft Visio** – пакет деловой графики.

Все программы Microsoft Office содержат встроенные средства программирования, основанные на языке программирования VBA (Visual Basic for Application).

В данном пособии представлены способы работы в приложениях Microsoft Office 2003, а также даны краткие пояснения по основным приемам работы в Microsoft Office 2007.

Для повышения эффективности работы в приложениях Microsoft Office 2007 разработан новый интерфейс. Меню и панели инструментов заменены так называемой лентой. Она облегчает доступ к командам и состоит из вкладок, связанных с определенными целями или объектами. Каждая вкладка, в свою очередь, имеет в своем составе несколько групп взаимосвязанных элементов управления. По сравнению с прежним меню и панелями инструментов лента вмещает значительно больше содержимого – кнопок, коллекций, элементов диалоговых окон и т. д. Основные команды работы с документами (создание, сохранение, печать и др.) открываются нажатием кнопки **Office**. Масштаб просмотра документа можно установить, используя **Масштаб** на панели **Вид**. На этой же панели определяется вид просмотра документа.

Если поиск требуемых команд при выполнении различных действий вызывает затруднение, то можно воспользоваться справкой, где описано расположение всех команд на имеющихся лентах.

## 1.5. Задание для выполнения на компьютере

1. Ознакомиться с учебником по Windows, открыв **Пуск/Справка**.
2. Поработать с приложением **Калькулятор**. В пункте меню **Вид** выбрать команду **Инженерный**. Выполнить несколько операций вычисления. Нажав клавиши **<Alt> + <PrtSc>**, занести в буфер обмена вид окна.
3. Запустить приложение **Проводник**, просмотреть содержимое пунктов меню и справку.
4. В папке **Мои документы** создать дерево папок (рис. 1.2).



Рис. 1.2. Дерево папок

5. Запустить графический редактор **Paint**, разместить на экране рисунок калькулятора из буфера обмена и сохранить его в папке **Мои документы**.
6. В приложении **Paint** создать рисунок, используя кнопки на панели инструментов. Разместить на рисунке текст. Раскрасить рисунок в различные цвета.
7. Опробовать операции копирования и вставки части рисунка. Поработать с командами **Отразить/Повернуть** и **Растянуть/Наклонить**. Сохранить рисунок в папке **Графика**.
8. В редакторе **Блокнот** ввести текст и сохранить файл в папке **Текст**.
9. Найти на компьютере созданный ранее рисунок, используя **Пуск/Найти**, и скопировать его в папку **Фамилия**, переименовать.
10. Найти на диске **С:** файлы с расширением **.bmp** и скопировать некоторые из них в папку **Графика**.
11. На диске **С:** найти текстовые файлы, созданные 3 сентября, и скопировать некоторые в папку **Текст**.
12. Определить объем свободного места на диске **С:** и поместить информацию об этом в один из текстовых файлов.

## 2. ПРИЛОЖЕНИЕ MICROSOFT WORD

В приложении Word 2003 можно раскрыть несколько панелей пиктографических меню. Они устанавливаются с помощью команды **Панели инструментов** из пункта меню **Вид**. Обычно загружаются две панели: **Стандартная** и **Форматирование**. Панель можно зацепить мышкой и переместить на нужное место. В современных программах около кнопок появляются надписи, поясняющие функции кнопок, если на них навести курсор.

### 2.1. Основные операции в приложении Word

**Создание документа.** Чтобы создать новый документ, следует выполнить команду **Создать** из пункта меню **Файл** либо щелкнуть кнопку **Создать** на панели **Стандартная**. Word позволяет работать с несколькими документами одновременно, переход между которыми осуществляется из пункта меню **Окно**.

Можно установить масштаб просмотра документа, используя раскрывающийся список **Масштаб** на панели **Стандартная**.

**Набор текста.** Ввод текста осуществляется путем набора символов на клавиатуре. По достижении правого края страницы текст автоматически переносится на новую строку. Клавишу **<Enter>** нужно нажимать тогда, когда следует начать новый абзац.

Чтобы вводимый текст замещал, а не сдвигал текст, имевшийся ранее, следует включить режим замены. Переключение режима замены осуществляется нажатием клавиши **<Insert>** на клавиатуре или двойным щелчком на индикаторе **Зам** в строке состояния, располагающейся в нижней части экрана.

**Режимы просмотра документа.** Один и тот же документ может иметь различный вид в зависимости от установленного режима просмотра документа на экране. Режим можно установить с помощью команд из пункта меню **Вид**, а также с помощью четырех кнопок, расположенных слева от горизонтальной полосы прокрутки.

Первая левая кнопка устанавливает *обычный режим*. Этот режим используется при простом вводе и редактировании текста. В этом режиме не отображаются специальные элементы страницы, рисунки и столбцы текста.

*Режим web-документа* позволяет посмотреть, как будет выглядеть страница в сети Интернет.

В *режиме разметки страницы* документ представляется на экране точно так, как он будет выглядеть при печати на бумаге.

Следующая кнопка открывает документ в *режиме структуры*. Этот режим удобен для работы над планом документа (составление, просмотр, редактирование).

Используя **Вид/Режим чтения**, можно разместить на экране две страницы целиком и читать документ как книгу, перемещаясь по нему с помощью значков под вертикальной линейкой прокрутки.

Команда **Вид/Схема документа** располагает документ в двух различных окнах. Слева находятся заголовки, которые позволяют перемещаться по тексту в правой части.

**Выделение участков текста.** Чтобы выделить часть строки, надо подвести курсор к началу участка выделения и протащить мышью к концу. Для выделения целой строки следует щелкнуть мышкой слева от нее. Для выделения нескольких строк нужно слева от них протащить мышку. Выделить весь документ можно командой **Выделить все** из пункта меню **Правка**. Чтобы убрать выделение, надо щелкнуть мышкой вне выделенного текста.

**Копирование, вырезание, вставка.** Для перемещения текста можно использовать описанные ранее способы, а также мышку, зацепив которой за выделенный текст, можно переместить его в нужное место. Если при этом держать нажатой клавишу **<Ctrl>**, то текст копируется.

Для отмены операции служит команда **Отменить** из пункта меню **Правка** либо кнопка **Отменить** на панели **Стандартная**.

**Форматирование текста.** Для всего документа или выделенного участка можно установить формат, выполнив команду **Абзац** из пункта меню **Формат** и сделав соответствующие настройки: определить отступы, интервал между строчками, тип выравнивания.

Над окном документа можно расположить координатную линейку с помощью команды **Линейка** из пункта меню **Вид**. Линейка позволяет своими указателями установить ширину текста и красную строку.

Для установки размера шрифта, его типа, цвета и т. д. можно использовать кнопки на панели инструментов **Форматирование** или возможности диалогового окна **Шрифт**, открываемого командой **Формат/Шрифт**. В появившемся окне на вкладке **Шрифт** можно осуществить нестандартные варианты подчеркивания текста (например, двойной чертой или пунктиром), можно оформить верхний и нижний индексы, зачеркнутый текст, текст с тенью, большие и малые прописные символы.

Элементы управления вкладки **Интервал** позволяют изменить интервал между символами, благодаря этому текст может быть уплотнен или разрежен. Средства вкладки **Анимация** применяются для динамического оформления текста, но только в электронных документах.

Для выравнивания текста на странице используются следующие кнопки на панели инструментов **Форматирование**: выравнивание по левому краю, выравнивание по центру, выравнивание по правому краю, выравнивание по ширине.

Кнопки **Уменьшить отступ** и **Увеличить отступ** позволяют расположить текст на странице нужным образом.

В Word 2007 команда **Абзац** находится на ленте в разделе **Главная**. В этом же разделе можно изменить шрифт (тип, начертание, размер и т. д.) и стиль текста.

**Списки.** Для преобразования текста в список надо его выделить и щелкнуть по кнопке **Нумерация** (на панели **Форматирование**) для преобразования текста в нумерованный список или щелкнуть по кнопке **Маркеры** – для преобразования текста в маркированный список. Создание списка заканчивается двукратным нажатием на клавишу <Enter>. Для создания многоуровневых списков удобно использовать клавишу <Tab> (перейти на строку подпункта и нажать <Tab> или кнопку **Увеличить отступ** на панели **Форматирование**).

Чтобы изменить или настроить формат списка, необходимо выполнить команду **Формат/Список** и в открывшемся диалоговом окне выбрать подходящий вид списка.

В Word 2007 текст для преобразования его в список надо выделить и выбрать на ленте в разделе **Главная** соответствующую пиктограмму для нужного типа списка. Тип списка можно изменить с помощью стрелки, расположенной справа.

**Вставка файла в текст.** В текст можно вставить файл с другим текстом. Для этого надо зафиксировать курсор в месте вставки файла, выполнить команду **Файл** из пункта меню **Вставка**. В появившемся окне выбрать нужную папку, нужный файл и нажать клавишу ввода.

**Вставка рисунка в текст.** В текст можно вставить рисунок, подготовленный самостоятельно или имеющийся в библиотеке рисунков Word. Для этого надо:

- зафиксировать курсор в месте вставки рисунка;
- выполнить команду **Рисунок** из пункта меню **Вставка**;
- в появившемся окне выбрать нужный каталог и нужный файл, включить режим **Просмотр** (открывается при нажатии кнопки **Представление**), просмотреть рисунок и нажать клавишу <Enter>.

(В Word 2007 рисунки из коллекции Word открываются при нажатии пиктограммы **Клип**.)

Чтобы выполнить обтекание рисунка текстом, надо рисунок выделить, в контекстном меню выбрать команду **Формат рисунка** и на вкладке **Положение** задать вид обтекания.

Вставленный рисунок можно откорректировать с помощью кнопок на панели **Рисование**. Как правило, она расположена в нижней части окна программы. В случае отсутствия панели необходимо выполнить команду **Вид/Панели инструментов/Рисование**.

Рисунок можно составить с помощью графических примитивов (линии, прямоугольника, окружности), а можно воспользоваться коллекцией автофигур, которая содержит фигурные стрелки, элементы блок-схем, различные выноски и прочие фигуры. Коллекция автофигур открывается списком **Автофигуры** на панели **Рисование**.

Фигуру для редактирования необходимо выделить, щелкнув по ней мышью. Затем можно менять расположение и размеры, используя маркеры. Для выделения нескольких фигур следует нажать кнопку **Выбор объектов** на панели **Рисование** и выделить прямоугольную область с несколькими фигурами. Можно выделять фигуры, щелкая по ним мышью с нажатой клавишей **<Shift>**.

Помимо редактирования фигур, с ними можно производить различные действия, нажав кнопку **Рисование** на панели **Рисование**. Команда **Рисование/Порядок** позволяет поменять порядок фигур на рисунке. Команда **Рисование/Повернуть/Отразить** предоставляет возможность повернуть фигуру или получить ее зеркальное отображение, отразив вертикально или горизонтально. При вставке рисунка в текст можно воспользоваться командой **Рисование/Обтекание текстом**. Несколько фигур для выравнивания их относительно краев или центра рисунка необходимо выделить и выбрать команду **Рисование/Выровнять/Распределить**.

Чтобы разместить текст на рисунке, следует воспользоваться кнопкой **Надпись** на панели **Рисование**. Для размещения текста внутри фигуры нужно в контекстном меню фигуры выбрать команду **Вставить текст**.

После того, как рисунок скомпонован из различных фигур, его можно сгруппировать. Для этого следует выделить все фигуры (кнопка **Выбор объектов**) и выполнить команду **Рисование/Группировать**. При необходимости рисунок опять может быть разгруппирован.

**Вставка текущих времени и даты.** Для вставки в документ текущих времени и даты необходимо установить текстовый курсор в то

место, куда должны быть вставлены дата и время, и выполнить команду **Вставка/Дата и время**. В открывшемся окне следует выбрать подходящий формат. Если нужно, чтобы дата и время обновлялись автоматически при каждом новом открытии документа, то следует установить флажок **Обновлять автоматически**.

**Вставка надписи.** Для вставки текста, рисунка в рамке надо:

- зафиксировать курсор в месте вставки надписи;
- выполнить команду **Надпись** из пункта меню **Вставка**, курсором мыши в виде крестика обозначить размеры рамки;
- поместить внутри рамки текст или рисунок.

Чтобы убрать линию вокруг рамки, следует установить курсор на границу надписи и в контекстном меню выбрать команду **Формат надписи**, затем на вкладке **Цвета и линии** определить тип линии – «Нет линии». На других вкладках можно задать виды обтекания надписи текстом, размеры надписи и др.

## 2.2. Таблицы и диаграммы

**Создание таблиц.** Для создания в тексте таблицы надо выполнить команду **Таблица/Вставить/Таблица** и ввести количество строк и столбцов или нажать кнопку **Вставить таблицу** на панели **Стандартная** и протяжкой мыши задать нужное количество столбцов и строк.

При создании шапки таблицы можно выполнять следующее:

- зацепив вертикальную границу столбца, корректировать его ширину;
- объединять ячейки, предварительно выделив их и выполнив команду **Объединить ячейки** из пункта меню **Таблица**;
- менять направление текста в ячейке, выбрав команду **Направление текста** в контекстном меню;
- задавать обводки границ, выделяя соответствующие зоны ячеек с помощью кнопки **Внешние границы** на панели **Форматирование**.

Таблицу можно создать на основе уже набранного текста путем преобразования его в табличную форму. Для этого нужно определить, какой элемент в тексте используется в качестве разделителей столбцов. Обычно применяется знак табуляции: каждая строка таблицы набирается в один абзац, а столбец от столбца отделяется знаком табуляции. Если предполагается пустая ячейка, то на этом месте ставятся два знака табуляции. Выбрав разделитель, нужно ввести текст (или



отформатировать существующий), затем выделить его и выполнить операцию **Таблица/Преобразовать в таблицу**. В открывшемся окне при необходимости уточняются параметры преобразования (тип разделителя, количество столбцов и т. д.).

В Word 2007 таблица вставляется с помощью команды **Вставка/Таблица**.

**Создание простых диаграмм.** В Word наиболее часто используются гистограммы, линейчатые диаграммы, графики, круговые диаграммы, кольцевые диаграммы и др.

Для создания диаграммы необходимо выделить всю таблицу или ее часть, выполнить команду **Вставка/Рисунок/Диаграмма**. Будет создана диаграмма, основанная на содержимом выделенной таблицы (части таблицы), в строку меню Word вставятся дополнительные меню, в стандартную панель инструментов добавятся кнопки, относящиеся к диаграммам.

При необходимости можно менять данные в таблице данных, что автоматически приведет к изменению диаграммы. Если таблицы данных на экране нет, то вызвать ее можно двойным щелчком левой кнопки мыши по диаграмме.

В Word 2007 для работы с диаграммами используется команда **Вставка/Диаграмма**.

**Создание организационных диаграмм.** Для создания новой организационной диаграммы следует установить курсор в то место, где будет находиться диаграмма, выполнить команду **Вставка/Рисунок/Организационная диаграмма**. Word создаст базовую диаграмму, в которую можно ввести текст и добавить необходимые фигуры. Для ввода текста в фигуры надо установить текстовый курсор внутри фигуры и набрать текст.

Для изменения вида диаграммы (добавление и удаление фигур, изменение связей) следует выделить фигуру, к которой нужно добавить соединение, щелкнуть по стрелке списка **Добавить фигуру** на панели инструментов **Организационная диаграмма**. В открывшемся меню необходимо выбрать вид связи: **Подчиненный** – добавляет фигуру уровнем ниже выделенной, **Коллега** – добавляет фигуру того же уровня, **Помощник** – добавляет фигуру уровнем ниже и в стороне от основной ветви.

Для изменения структуры диаграммы следует щелкнуть по кнопке **Макет** на панели инструментов и выбрать подходящую структуру. Изменение стиля диаграммы (**По умолчанию**, **Контур**, **Тень**, **Огонь** и т. д.) осуществляется с использованием кнопки **Автоформат** на панели

инструментов. Для выбора вида схемы диаграммы надо щелкнуть по кнопке **Добавить диаграмму или организационную диаграмму** на панели **Рисование**. На экране появится окно, в котором можно выбрать вид схемы.

В Word 2007 создание организационной диаграммы осуществляется с помощью команды **Вставка/Таблица**.

### 2.3. Задание для выполнения на компьютере

Набрать текст на двух страницах. Вид первой страницы представлен на рис. 2.1, а второй – на рис. 2.2.

На первой странице надо расположить текст так, чтобы он занимал всю страницу. Обозначить в строке **Выполнил:** свою фамилию и номер группы. Основной текст должен иметь шрифт Times New Roman, размер 12 пт, выравнивание по ширине, абзацный отступ 1,25 см, одинарный междустрочный интервал.

Учреждение образования «Белорусский государственный технологический университет»
Кафедра информационных систем и технологий (рисунок)
Расчетно-графическая работа
<b>Программирование циклических процессов</b>
Выполнил: Проверил:
Минск 2010

Рис. 2.1. Вид первой страницы задания

Текст на следующей странице должен иметь шрифт Times New Roman, размер 14 пт, выравнивание по ширине, абзацный отступ 1,25 см, одинарный междустрочный интервал. Описание схемы алгоритма программы следует расположить в надписи.

## Введение

В инженерной деятельности компьютер используется достаточно широко. Грамотный специалист должен:

- уметь работать с приложениями Microsoft Office;
- понимать основы алгоритмизации прикладных задач;
- уметь составлять программы на языке программирования.

### 1. Циклические программы

В циклических программах повторяются операторы, с помощью которых производятся вычисления. При этом меняется содержимое переменной, называемой параметром цикла.

#### 1.1. Постановка задачи

Формула	Исходные данные
$y = \cos x$	$x = 3(0.1)4$

Здесь запись  $x = 3(0.1)4$  означает, что  $x$  меняется от 3 до 4 с шагом 0.1.

#### 1.2. Схема алгоритма программы

Пусть требуется вычислить  $y = x^2$ , где  $x = 3(0.1)4$ , при этом  $x$  меняется от 3 до 4 с шагом 0.1. На рисунке представлена схема алгоритма циклической программы.

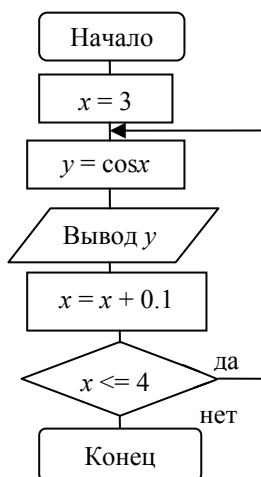


Рисунок. Схема алгоритма

Первоначально значение переменной  $x$  равно 3. Вычисляется значение  $y$  при  $x = 3$  и полученный результат выводится. Затем значение  $x$  увеличивается на 0.1. Если  $x$  меньше или равно 4, то вновь рассчитывается  $y$ , в противном случае вычисления прекращаются.

Таким образом, определяется одиннадцать значений  $y$ .

Параметром цикла в данной задаче является переменная  $x$ .

Данный алгоритм можно реализовать при помощи операторов цикла **Do...Loop** или **For...Next**.

Рис. 2.2. Вид второй страницы задания

Необходимо установить следующие параметры страницы: верхнее поле – 2 см, левое – 3 см, нижнее – 2 см, правое – 1 см.

## 3. ОФОРМЛЕНИЕ ДОКУМЕНТОВ В ПРИЛОЖЕНИИ WORD

### 3.1. Текстовые эффекты и ввод формул

**Буквицы.** Первую букву абзаца часто оформляют в виде буквицы. Для этого надо выделить первую букву, в пункте меню **Формат** выполнить команду **Буквица**, а в ее диалоговом окне задать желаемый режим оформления буквицы (эффект **Буквицы** виден в режиме разметки страницы).

В Word 2007 для создания буквицы используется команда **Вставка/Буквица**.

**Колонки текста.** Можно набранный обычным образом текст разместить в виде колонок. Для этого следует его выделить, нажать кнопку **Колонки** на панели **Стандартная** и при появлении рамки из 4 колонок протяжкой мыши задать необходимое количество колонок. Результат следует смотреть в режиме разметки страницы.

В Word 2007 для работы с колонками служит команда **Разметка страницы/Колонки**.

**Создание сложных текстовых эффектов.** Для создания красивых надписей применяется **Мастер текста (WordArt)**. Для работы с ним надо выполнить команду **Вставка/Рисунок/Объект WordArt**, в появившемся меню выбрать вид оформления текста и в раскрывшемся окне набрать текст. Заканчивается работа с объектом **WordArt** щелчком мыши вне рамки надписи.

В Word 2007 для создания сложных текстовых эффектов выполняется команда **Вставка/WordArt**.

При необходимости откорректировать такой текст надо сделать на нем двойной щелчок и, используя кнопки панели **WordArt**, произвести соответствующую коррекцию.

**Набор математических формул.** При написании простых математических формул различные надстрочные и подстрочные символы устанавливаются с помощью команды **Формат/Шрифт**.

Такие символы, как знак интеграла, суммы, греческие буквы и т. п., можно вставить в текст, выполнив команду **Вставка/Символ**.

Если нужно в тексте набрать более сложные формулы, используется приложение **Microsoft Equation**. Для его загрузки следует выполнить команду **Объект** из пункта меню **Вставка** и в появившемся списке выбрать соответствующую строчку.

В Word 2007 для этих же целей выполняется команда **Вставка/Объект/Microsoft Equation 3.0**.

После загрузки приложения и появления пиктографического меню можно вводить формулу в появившейся рамке. Заканчивается запись формулы щелчком мыши вне рамки. Размер шрифта формулы изменяется с помощью команды **Размер/Определить**. Также можно изменить шрифты и интервалы, используя соответственно команды **Стиль/Определить** и **Формат/Интервал**.

Чтобы отредактировать формулу, следует дважды щелкнуть мышью в поле формулы и внести нужные изменения.

В текстовом редакторе Word 2007 для вставки объекта **Microsoft Equation** на вкладке **Вставка** служит кнопка **Объект** группы **Текст**. Кроме этого, добавить формулу можно, используя кнопку **Формула**.

**Вычисления в таблицах.** Для выполнения вычислений в таблице нужно установить курсор в ячейку, где должен быть результат, и вставить формулу, выбрав команду **Таблица/Формула**.

В качестве формулы может быть использована готовая математическая функция, которая выбирается из списка **Вставить функцию**, или собственная формула, написанная с использованием математических операторов и ссылок на ячейки. Ссылки на ячейки можно записать в виде **A1, A2, B1, B2**, где латинская буква указывает на столбец (**A** – первый столбец, **B** – второй и т. д.), а число – на строку.

В Word 2007 после расположения курсора внутри таблицы появляются дополнительные разделы **Работа с таблицами (Конструктор и Макет)**. Для вставки функции служит команда **Макет/Формула**.

**Колонтитулы.** Это текст, который печатается внизу или вверху каждой страницы документа. Обычно в область колонтитула вносят номер страницы, текущую дату, название документа и т. д. Можно разместить в колонтитуле рисунок, что создаст эффект водяного знака. Оформить колонтитул можно с помощью панели инструментов, которая появляется при выполнении команды **Вид/Колонтитулы**. Для завершения работы с колонтитулом необходимо нажать кнопку **Заккрыть** на панели **Колонтитулы**. В Word 2007 для создания колонтитула используется команда **Вставка/Колонтитулы**.

**Фон.** Для определения фона документа выполняется команда **Формат/Фон**. Если в появившемся окне в качестве подложки установить рисунок, то он будет выполнять роль водяного знака.

## 3.2. Оформление документа, его редактирование и печать

**Границы и заливка.** Весь текст или выделенную часть можно заключить в рамку, выполнив команду **Границы и заливка** из пункта меню **Формат** или используя соответствующие кнопки на панели **Форматирование**.

В появившемся окне на вкладке **Границы** задаются три параметра: тип линии, цвет и ширина. Четыре кнопки, находящиеся в правой части окна, включают/отключают соответствующую границу (верхнюю, нижнюю, правую, левую). На вкладке **Страница** в списке **Рисунок** выбирается графический или декоративный стиль рамки для страницы. На вкладке **Заливка** задается цвет фона, на котором располагается текст.

В Word 2007 для этих целей применяется команда **Разметка страницы/Границы страниц**.

**Поиск и замена.** Имеются специальные средства поиска и замены, которые позволяют найти в тексте фрагмент, заданный в виде текстовой строки, и заменить его новым текстом. Для этого необходимо командой **Правка/Найти** открыть диалоговое окно **Найти и заменить** и на вкладке **Найти** в поле **Найти** ввести фрагмент разыскиваемого текста (для задания дополнительных параметров поиска надо щелкнуть на кнопке **Больше**).

В Word 2007 для поиска и замены используется команда **Главная/Найти**.

Дополнительные кнопки **Формат** и **Специальный** позволяют разыскивать текст, отформатированный указанным образом, и специальные непечатаемые символы. Поиск начинается после щелчка на кнопке **Найти далее**.

Для замены найденного текста используют вкладку **Заменить**. Заменяющая строка вводится в поле **Заменить на...** По щелчку на кнопке **Найти далее** разыскивается очередное место, где заданная строка встречается в документе, затем щелчком на кнопке **Заменить** выполняется замена, если она необходима. Если заранее известно, что замену следует произвести по всему документу и во всех случаях, можно сразу щелкнуть на кнопке **Заменить все**.

Вкладка **Перейти** используется для перехода к специфическому тексту или объекту, например к заданной странице, сноске и т. д.

**Стилевое оформление.** Стиль – это набор параметров форматирования, которые применяются к тексту документа. Имеется два типа

стилей: стиль абзаца и стиль символов. Стиль абзаца представляет собой группу форматов, которые влияют на весь абзац в целом: шрифт абзаца, расположение абзаца на полосе (влево, вправо, отступы и т. д.), табуляция и др. Стиль символов включает в себя оформление, которое может быть применено к символам командами окна **Шрифт**.

Доступные стили перечислены в раскрывающемся списке **Стиль**, расположенном на панели инструментов **Форматирование**. При выборе стиля из списка изменяется формат текущего абзаца или формат выделенного фрагмента.

В Word 2007 для задания стиля служит команда **Главная/Изменить стили**.

**Формат по образцу.** Это копирование параметров форматирования выделенного объекта или текста, которое может быть применено к выбранному объекту или тексту. Чтобы выполнить форматирование по образцу, надо установить курсор на абзац, имеющий нужный метод форматирования, и щелкнуть на кнопке **Формат по образцу** на панели **Стандартная**. Затем следует выделить текст, формат которого требуется изменить.

Если нужно изменить формат нескольких абзацев, следует дважды щелкнуть на кнопке **Формат по образцу**. После внесения всех необходимых изменений надо еще раз щелкнуть на кнопке **Формат по образцу** или нажать клавишу <Esc>. Форматировать по образцу можно не только абзац, но и отдельные слова, символы или объекты.

В Word 2007 для изменения формата абзацев применяется команда **Главная/Формат по образцу**.

**Отображение специальных символов.** Приложение Word предусматривает возможность отображения таких непечатаемых символов, как символы конца абзаца, табуляции, пробела, мягкого переноса. Отображение этих специальных символов удобно включать, чтобы видеть все приемы, с помощью которых оформлен текст. Можно включить/выключить их отображение с помощью кнопки **Непечатаемые знаки** на панели **Стандартная**.

В Word 2007 для отображения специальных символов используется команда **Главная/подраздел Абзац**.

**Разрыв страницы.** Иногда необходимо создать разрыв страницы в нужном месте. Тогда следует расположить курсор в месте разрыва и выполнить команду **Вставка/Разрыв**. В открывшемся диалоговом окне необходимо выбрать требуемое действие.

В Word 2007 для вставки разрыва страницы служит команда **Вставка/Разрыв страницы**.

**Правописание и перенос.** Правописание в приложении Word проверяется автоматически, при этом неверно написанные слова подчеркиваются красной волнистой линией. Если выполнить команду **Правописание** из пункта меню **Сервис**, то для каждого ошибочно записанного текста компьютер предлагает свои варианты замены. Используя кнопки **Пропустить**, **Заменить**, **Добавить**, можно осуществлять коррекцию текста.

В Word 2007 для проверки правописания используется команда **Рецензирование/Правописание**.

Чтобы текст документа выглядел более компактным, в нем необходимо расставить переносы. Для этого надо выполнить команду **Сервис/Язык/Расстановка переносов**. В появившемся окне следует установить флажок **Автоматическая расстановка переносов**.

В Word 2007 для расстановки переносов применяется команда **Разметка страницы/Расстановка переносов**.

**Нумерация страниц.** Чтобы расставить номера страниц в документе, необходимо выполнить команду **Вставка/Номера страниц** (в Word 2007 – аналогично). В открывшемся диалоговом окне следует установить нужные параметры. Использование кнопки **Формат** в этом окне позволяет указать, какими символами будет производиться нумерация (арабские, римские цифры, буквы) и с какого числа начнется нумерация.

**Создание оглавления.** Страницы документа должны быть пронумерованы. К участкам текста, являющимся заголовками, надо применить стили заголовков (**Заголовок 1**, **Заголовок 2** и т. д.), используя **Стиль** на панели **Форматирование**.

В Word 2007 для этих целей служит команда **Главная/Стили**.

Затем надо выполнить команду **Вставка/Ссылка/Оглавление и указатели**, при этом курсор должен быть установлен в том месте документа, где предполагается разместить оглавление. Далее в диалоговом окне **Оглавление и указатели** следует выбрать вкладку **Оглавление** и задать внешний вид оглавления и его параметры: число уровней, внешний вид заполнителя. Для изменения оглавления надо внести соответствующие коррективы в документ и вновь выполнить команду **Вставка/Ссылка/Оглавление и указатели**. На вопрос о замене оглавления ответить **<ОК>**.

В Word 2007 для вставки оглавления выбирается команда **Ссылки/Оглавление**.

**Сноски.** Сноска – это примечание к тексту, которое может находиться или в нижней части документа или в конце всего документа.



Она отделяется от основного текста чертой и отмечается специальным знаком сноски.

Сноски бывают нумерованные (1, 2, 3 или I, II, III или A, B, C и т. д.) или маркированные (# или ® или & и т. д.). Для вставки сносок необходимо расположить курсор в том месте, где будет сноска, выполнить команду **Вставка/Ссылка/Сноска**, заполнить диалоговое окно и нажать кнопку **Вставить**. В тексте появится знак сноски, а в нижней части страницы (или документа) откроется область сносок, в которую нужно ввести требуемый текст.

В Word 2007 для вставки сноски выполняется команда **Ссылки/Сноски**.

**Предварительный просмотр.** Чтобы убедиться в том, что текст на бумаге будет выглядеть так, как надо, используется специальный режим предварительного просмотра.

Для перехода в этот режим служит кнопка **Предварительный просмотр** или команда **Файл/Предварительный просмотр**. При этом можно задать просмотр сразу нескольких страниц. Для выхода из просмотра надо нажать кнопку **Заккрыть**.

В Word 2007 для предварительного просмотра выбирается команда **Вид/Режим чтения**.

**Параметры страницы документа.** Параметры страницы документа задаются командой **Файл/Параметры страницы**. В появившемся окне на вкладке **Поля** можно установить поля документа, определить ориентацию документа (книжная или альбомная). На вкладке **Размер бумаги** можно изменить формат бумаги.

В Word 2007 для задания параметров страницы используется команда **Разметка страницы/Параметры страницы**.

**Печать.** Команда **Печать** из пункта меню **Файл** или кнопка **Печать** на панели **Стандартная** вызывает появление диалогового окна, в котором надо установить параметры печати. Печатать можно весь документ, отдельные страницы или выделенный фрагмент.

В Word 2007 для этих целей следует нажать кнопку **Офис**, затем **Печать**.

### 3.3. Задание для выполнения на компьютере

1. Открыть ранее созданный документ из двух страниц и добавить третью, вид которой представлен на рисунке.

## 2. Текст программы

```
Private Sub CommandButton1_Click()
Dim x As Single, y As Single
For x = 3 To 4 Step 0.1
y = Cos(x)
MsgBox y
Next
End Sub
```

## 3. Вычисление суммы

Вычислить  $S = \sum_{i=1}^{10} \left( \frac{a_i + 1}{i + 10} \right)^2$ , где  $a = \{2; 0.1; -4; 5.7; -3; 11; 0.2; 8; -6; 21\}$ .

Текст программы	Контрольный расчет	
<pre>Private Sub CommandButon1_Click() Dim a As Single, s As Single Dim i As Integer s = 0 For i = 1 To 10 a = Cells(i, 1) s = s + a Next MsgBox s End Sub</pre>	Значения массива $a$	2
		0.1
		-4
		5.7
		-3
		11
		0.2
		8
		-6
		21
Результат	44	

## Заключение

Разного рода вычисления используются в инженерных расчетах достаточно широко, поэтому грамотный специалист должен уметь их выполнять.

## Оглавление

Введение .....	2
1. Циклические программы .....	2
1.1. Постановка задачи .....	2
1.2. Схема алгоритма программы .....	2
2. Текст программы .....	3
3. Вычисление суммы .....	3
Заключение .....	3

Вид третьей страницы задания

На этой странице формулу со знаком суммы создать с помощью **Microsoft Equation**. Символы фигурных скобок вставить с использованием команды **Символ** из пункта меню **Вставка**.

В таблице с контрольным расчетом для оформления применить объединение ячеек, а для подсчета суммы – команду **Формула** из пункта меню **Таблица**. Поворот текста осуществить с использованием контекстного меню. Оглавление создать автоматически.

2. В получившемся документе из трех страниц оформить название с помощью **Мастера текста**. В нижнем колонтитуле записать свою фамилию. Проверить орфографию текста, осуществить перенос слов. Опробовать операции поиска и замены.

3. Просмотреть документ в различных режимах просмотра (обычный, структура, разметка страницы).

Нажимая кнопку **Непечатаемые знаки**, ознакомиться с тем, как отображаются на экране символы пробела и конца абзаца. В режиме предварительного просмотра расположить вместе три страницы.

## 4. ПРИЛОЖЕНИЕ MICROSOFT EXCEL

Приложение Excel предназначено для работы с электронными таблицами. Документ, созданный в Excel, называется книгой (**Book**) и при открытии имеет имя **Книга1**. В состав книги входят рабочие листы (**WorkSheets**), называемые **Лист1**, **Лист2** и т. д. Листы можно переименовывать, удалять и добавлять новые с помощью контекстного меню либо используя соответствующую команду: **Правка/Удалить лист**, **Вставка/Лист**, **Формат/Лист/Переименовать**.

Переход от листа к листу удобно осуществлять, нажимая кнопки со стрелками, находящиеся слева от названия листов, так как не все названия листов одновременно видны на экране.

Для получения сведений по отдельным вопросам работы приложения служит пункт меню **Справка**.

### 4.1. Основные операции в приложении Excel

*Создание документа, его открытие и сохранение.* Для создания документа надо выполнить команду **Создать** из пункта меню **Файл** или нажать кнопку **Создать** в пиктографическом меню **Стандартная**. Если документ предварительно создан, то его можно открыть с помощью команды **Открыть** из того же пункта меню или нажать кнопку **Открыть**. Можно загружать несколько документов, имена которых записаны в ниспадающем меню пункта **Окно**. Активный документ помечен галочкой.

Если нужно сохранить документ первый раз, то следует выполнить команду **Сохранить как...** или **Сохранить** из пункта меню **Файл**. Для записи документа на прежнее место необходимо использовать команду **Сохранить** из того же пункта меню или кнопку **Сохранить**.

В Excel 2007 для создания нового документа следует нажать кнопку **Office** и выполнить команду **Создать**. Необходимо обратить внимание на то, что, если в документе присутствуют макросы, созданные пользователем, то следует выбрать **Сохранить как.../Книга Excel с поддержкой макросов**. Если планируется в дальнейшем открывать созданный файл в старых версиях **Office**, то необходимо выбрать **Сохранить как.../Книга Excel 97–2003**.

*Имена ячеек.* Ячейки именуются буквой столбца и номером строки, например **A1**, **G5**.

Можно ячейке присвоить собственное имя. Для этого надо активировать ячейку и выполнить команду **Имя** из пункта меню **Вставка**, затем выбрать пункт **Присвоить** и в появившемся окне задать имя.

**Ввод и редактирование информации.** Для ввода информации в ячейку следует подвести к ней курсор, ввести нужный текст, число, формулу и нажать клавишу **<Enter>**.

В Excel есть вспомогательные средства для заполнения ячеек, облегчающие процесс ввода. Например, для ввода названий месяцев надо в первой ячейке написать название первого месяца, затем зацепить курсором мыши за маркер в правом нижнем углу и протянуть курсор на столько месяцев, сколько необходимо.

Создать другую последовательность данных можно подобным же образом: записать число, зацепить курсором мыши за маркер в правом нижнем углу, протянуть курсор на столько ячеек, сколько необходимо, и выполнить команду **Правка/Заполнить/Прогрессия**. В появившемся окне выбрать нужные параметры.

В Excel 2007 для создания последовательности данных служит команда **Главная/Редактирование/Заполнить/Прогрессия**.

Для записи информации различными форматами удобнее всего использовать контекстное меню. Если, например, в ячейке отображается дата вместо числа, то нужно в контекстном меню выполнить команду **Формат ячеек**, на вкладке **Число** выбрать формат **Числовые** и установить нужные параметры формата.

Чтобы отредактировать содержимое ячейки, надо сделать двойной щелчок в поле ячейки или нажать клавишу **<F2>** и внести изменения.

Выделение ячейки выполняется щелчком мыши на ней. Для выделения нескольких несмежных групп ячеек следует выделить одну группу, нажать клавишу **<Ctrl>** и, не отпуская ее, выделить другие ячейки. Чтобы выделить целый столбец или строку таблицы, необходимо щелкнуть мышью в поле имени. Для выделения рабочего листа надо щелкнуть на левой верхней (пустой) ячейке пересечения названий столбцов и строк. Для выделения нескольких листов необходимо нажать клавишу **<Ctrl>** и, не отпуская ее, щелкать на ярлыках листов.

**Операции со строками и столбцами.** Для вставки столбца или строки надо поместить курсор в нужное место, правой клавишей мыши вызвать контекстное меню, выполнить команду **Добавить ячейки** и в появившемся окне выбрать режим **Столбец** или **Строка**.

Удалять можно как содержимое строк и столбцов, так и целиком строки и столбцы. Для уничтожения содержимого выделенных ячеек

проще всего использовать клавишу **<Del>**. Для удаления столбцов, строк или ячеек надо их выделить и выполнить команду **Правка/Удалить** с последующим выбором нужной опции.

Для изменения ширины столбца или высоты строки следует указатель мыши поместить на границу в поле названий и перетащить границу нужным образом. Автоматическая настройка ширины столбца или высоты строки на самый длинный текст осуществляется двойным щелчком мыши на границе в поле названий или выбором команды **Формат/Строка (Столбец)/Автоподбор высоты (Автоподбор ширины)**.

**Копирование, вырезание, вставка.** Эти операции выполняются над выделенными участками с помощью команд меню **Правка** или кнопок на панели **Стандартная**.

Содержимое ячейки или выделенного интервала можно копировать и с помощью мыши. Для этого надо зацепить курсор мыши за правый нижний угол и, удерживая клавишу **<Ctrl>**, скопировать ячейку или интервал ячеек в нужное место. При переносе информации клавиша **<Ctrl>** не используется. При копировании формул таким способом автоматически меняются названия ячеек формул.

В Excel 2007 операции вырезания, копирования и вставки информации выполняются над выделенными участками с помощью кнопок на панели **Главная**.

**Оформление таблицы.** Для изменения цвета фона и цвета линий используются кнопки на панели **Форматирование**, команды **Ячейки** из пункта меню **Формат** или контекстное меню.

Кнопка **Границы** на панели **Форматирование** позволяет создать рамки в таблице.

Использование команды **Формат/Условное форматирование** предоставляет возможность настроить цвет фона выделенных ячеек и цвет шрифта в них в зависимости от некоторых условий. Например, при вводе положительных чисел один цвет чисел, при вводе чисел, превышающих некоторое значение, – другой цвет и т. д.

В Excel 2007 для изменения цвета фона и цвета линий используются кнопки на панели **Главная** или контекстное меню. Весьма эффективным и наглядным является применение стилей для таблиц с помощью команды **Главная/Стили**. Можно быстро форматировать данные таблиц на листе, используя темы и конкретные стили. Темы могут совместно применяться другими приложениями (например, Microsoft Word и Microsoft PowerPoint), а стили предназначены для изменения формата элементов только в приложении Excel.

**Фиксирование заголовков.** При работе с большими таблицами удобно фиксировать заголовки. Для этого следует поместить курсор в нужное место с учетом того, что закрепляется область левее и выше расположения курсора. Затем выполнить команду **Закрепить области** из пункта меню **Окно**.

В Excel 2007 закрепить области можно с помощью команды **Закрепить области** на вкладке **Вид**.

## 4.2. Вычисления по формулам

**Ввод формул.** При вводе формул надо соблюдать следующие правила:

- все формулы начинаются со знака «=»;
- при адресации ячеек используются только латинские буквы;
- арифметические операции обозначаются символами: \* (умножение), / (деление), + (сложение), - (вычитание), ^ (возведение в степень);
- для отделения целой части от дробной применяется запятая (или точка в зависимости от настройки Windows);
- аргументы используемых в формулах функций разделяются между собой символом «;»;
- интервалы ячеек записываются указанием первой и последней ячейки интервала со знаком двоеточия между ними. Например, **=A2\*2,2+МИН(S1:S10)**.

Формулы можно копировать и перемещать обычным способом, при этом автоматически меняются адреса ячеек.

**Использование встроенных функций.** Мастер функций (кнопка *fx*) дает возможность использовать готовые встроенные функции: математические, тригонометрические, финансовые, текстовые, статистические, функции даты и времени и т. д. Для того чтобы вставить функцию, необходимо:

- активизировать ячейку, нажать **fx (Мастер функций)**;
- в появившемся окне выбрать категорию функции и имя;
- в следующем окне задать аргументы функции.

Можно создать примечания к ячейкам. Для чего надо выполнить команду **Примечание** из пункта меню **Вставка**. Ячейка с примечанием помечается красным треугольником в правом верхнем углу.

Одна из часто применяемых в Excel функций – это функция автоматического суммирования. Кнопка **Автосумма** находится на панели **Стандартная**.

**Функция ЕСЛИ.** Функция **ЕСЛИ** используется для выбора направления вычислений. Например, **=ЕСЛИ(Е3>2;0,5\*Д3;0)**. Здесь, если условие **Е3>2** выполняется, то содержимое ячейки, в которой приведена эта формула, равно **0,5\*Д3**. Если условие не выполняется, то содержимое ячейки равно **0**.

**Адресация и ссылки.** В формулах различают абсолютную, относительную и смешанную адресацию ячеек. *Абсолютная адресация* использует для указания на ячейку ее фиксированное положение на листе. В абсолютной адресации перед именем столбца и номером строки ставится знак **\$**. Например, ячейка **\$A\$1** находится в столбце **A** строки **1**. При копировании имя ячейки остается прежним.

*Относительная адресация* указывает на ячейку, основываясь на ее положении относительно ячейки, в которой находится формула, и обозначается привычным образом, например ячейка **B3**. При копировании формул ячейки автоматически меняют имя.

В *смешанном адресе* ячейки знак **\$** ставится перед абсолютной частью. Например, **\$A5** – абсолютная ссылка на столбец **A**.

Чтобы *обратиться к ячейке другого листа*, следует указать имя листа и имя ячейки: **Лист1!F8**. *Объемные ссылки* – это ссылки на ячейки диапазона листов в книге. Например, пусть в файле Excel имеется 6 листов. Необходимо определить сумму значений из диапазонов (**B1:B20**), расположенных на всех этих листах. Формула будет иметь вид: **=СУММ(Лист1:Лист6!B1:B20)**.

При работе с объемными ссылками можно использовать до 11 функций в одной формуле.

Если открыты два файла Excel, то обратиться из первого файла к ячейке **\$A\$3** пятого листа второго файла надо следующим образом: **[Имя книги.xls]Лист5!\$A\$3**. Если нужно обратиться к ячейке не открытого файла, то в ссылке необходимо указать полный путь доступа к папке, где хранится книга: **"C:\Имя папки\[Имя книги.xls]Лист5"!\$A\$3**.

**Сообщения об ошибках.** Если формула в ячейке не может быть правильно вычислена, Excel выводит сообщение об ошибке.

**#ИМЯ?** – Excel не смог распознать имя, использованное в формуле;

**#ДЕЛ/0!** – в формуле делается попытка деления на нуль;

**#ЗНАЧ!** – использован недопустимый тип аргумента;

**#Н/Д** – такое сообщение может появиться, если в качестве аргумента задана ссылка на пустую ячейку;

**#ПУСТО!** – неверно указано пересечение двух областей, которые не имеют общих ячеек;

**#ССЫЛКА!** – в формуле задана ссылка на несуществующую ячейку;



**#ЧИСЛО!** – нарушены правила задания операторов, принятые в математике.

**Копирование формул.** В том случае, если надо скопировать только формулу или только значение, полученное по формуле, используется специальная вставка. Надо поместить подсветку на нужную ячейку, скопировать информацию, поместить курсор в место вставки и выполнить команду **Специальная вставка** из пункта меню **Правка**. В появившемся окне выбрать то, что следует копировать.

### 4.3. Построение диаграмм, обработка данных и печать

**Создание диаграмм.** Чтобы построить диаграмму, надо:

- выделить зону информации, для которой будет строиться диаграмма;
- нажать кнопку **Мастер диаграмм** на панели **Стандартная**, выбрать вид диаграммы и нажать кнопку **Далее**;

- построение графика осуществляется в несколько шагов. На каждом шаге появляется свое окно, в котором нужно выбирать соответствующие параметры и нажимать кнопку **Далее**. Можно определить диапазон данных, задать надписи на осях координат. На втором шаге на вкладке **Ряд** можно задать так называемую легенду, в которой обозначаются имена рядов. Нажатие кнопки **Готово** вызывает построение диаграммы.

Для редактирования диаграммы надо сделать по ней двойной щелчок. Можно задавать различные надписи на диаграмме, использовать различные шрифты и окраску. Объемная диаграмма может подвергаться поворотам, менять перспективу, высоту от основания. Удобно для редактирования использовать и контекстное меню.

Для построения диаграммы в Excel 2007 надо выделить зону информации, для которой будет строиться диаграмма, выбрать вкладку **Вставка** и нажать на кнопку с соответствующим типом диаграммы на панели.

**Создание сложных текстовых эффектов.** Для создания красивых надписей используется **Мастер текста (WordArt)**. Для работы с ним надо выбрать команду **Вставка/Рисунок/Объект WordArt**, в появившемся меню задать вид оформления текста и в раскрывшемся окне набрать текст. Заканчивается работа с объектом **WordArt** щелчком мыши вне рамки надписи.

При необходимости откорректировать текст надписи надо сделать на нем двойной щелчок и, используя кнопки панели **WordArt**, выполнить соответствующую коррекцию.

В Excel 2007 для создания красивых надписей используется **Вставка/Объект WordArt**.

**Сортировка и фильтрация данных.** Команда **Данные/Сортировка** позволяет отсортировать исходную информацию различным образом. При выполнении команды **Данные/Фильтр/Автофильтр** в верхнем ряду таблицы появляются небольшие кнопки, нажав на которые можно задать условия выборки данных. При записи условий выборки для текста надо учитывать, что один неизвестный символ обозначается знаком вопроса, а несколько – звездочкой.

В Excel 2007 **Автофильтр** в таблице по умолчанию включен, что позволяет использовать в формулах имена заголовков столбцов таблицы вместо ссылок на ячейки.

**Печать документа.** Перед печатью следует предварительно просмотреть подготовленный документ с помощью команды **Предварительный просмотр** из пункта меню **Файл**. Увеличенное изображение таблицы появляется, когда курсор находится на странице и превращен в изображение лупы. Для изменения масштаба печати таблицы в режиме предварительного просмотра нужно нажать кнопку **Поля** и с помощью мыши изменить размеры левого и правого полей. Если таблица широка, то надо нажать кнопку **Страница** и в поле диалогового окна выбрать нужный масштаб. Здесь же можно задать текст для верхнего и нижнего колонтитулов страницы.

Печать таблицы осуществляется по команде **Печать** из пункта меню **Файл** или при нажатии кнопки **Печать** на панели **Стандартная**.

#### 4.4. Задание для выполнения на компьютере

1. Создать таблицу заданного вида (рисунок).

**Заработная плата работников за январь**

Номер	Фамилия	Должность	Стаж	Оклад	Премия	Надбавка за стаж	Итого	Налоги	Получить	Доля
1										
2										
3										
4										
5										
Сумма										
Среднее										

Вид таблицы задания

2. Заполнить произвольно поля таблицы **Фамилия** и **Должность**. Стаж – в диапазоне 0–30 лет. Оклад – 200 000–800 000 руб.
3. Премия составляет 50% от оклада.
4. Надбавку за стаж вычислить на основании стажа. Если стаж больше 10 лет, тогда надбавка равна 20% от оклада, иначе – 0. Здесь используется встроенная функция **ЕСЛИ**.
5. Налоги составляют 20% от значения поля **Итого**.
6. Вычислить сумму и среднее значение для каждого столбца.
7. Значение поля **Доля** рассчитать как значение поля **Получить** для каждого работника, деленное на значение общей суммы всех значений поля **Получить**.
8. Построить столбчатую гистограмму для столбцов **Фамилия** и **Получить**. В легенде отобразить фамилии работников.
9. На 2-м и 3-м листах создать аналогичные таблицы за февраль и март.
10. Используя объемные ссылки, увеличить оклад за февраль на 10%, а за март – на 5% по сравнению с февралем.
11. На 4-м листе создать таблицу с полями **Месяц** и **Всего получить** для директора (или другого специалиста).
12. Построить график заработной платы директора фирмы за январь, февраль и март.

## 5. ЯЗЫК VBA. ЛИНЕЙНЫЕ И ЦИКЛИЧЕСКИЕ ПРОГРАММЫ

### 5.1. Общие сведения о принципах программирования в Excel

Создание проекта программы на языке VBA (Visual Basic for Application) состоит из двух взаимосвязанных процессов: размещение элементов управления на рабочем листе Excel или на форме (формах) пользователя и написание текста программы в соответствии с алгоритмом решаемой задачи.

1. Для размещения элемента управления на рабочем листе необходимо:

- вызвать панель инструментов **Элементы управления**;
- активизировать на панели нужный элемент и поместить его в соответствующем месте рабочего листа (установить указатель мыши в нужное место, нажать левую кнопку мыши и, не отпуская ее, очертить размеры элемента управлением движением курсора мыши по диагонали).

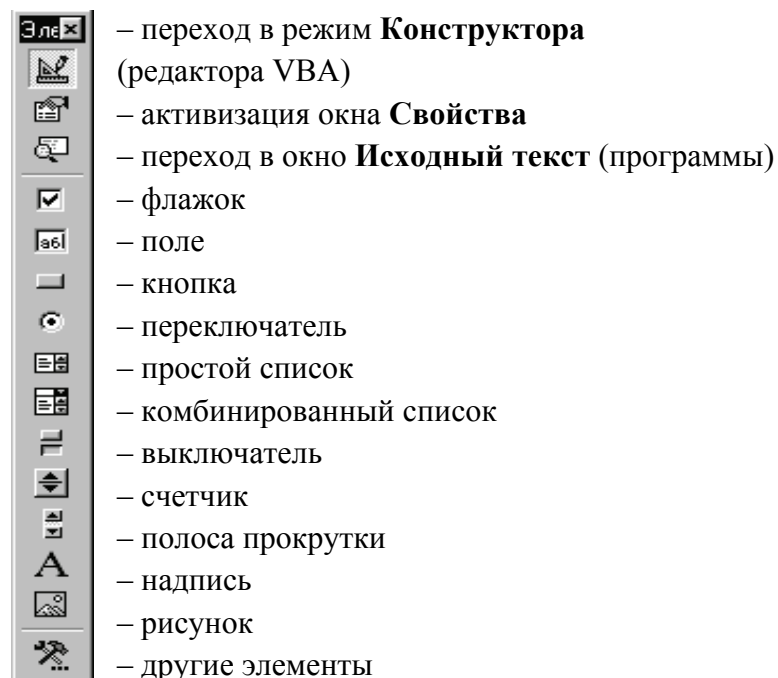


Рис. 5.1. Панель Элементы управления

Надпись на элементе управления, шрифт и цвет надписи и т. д. можно изменить в окне свойств, нажав на кнопку **Свойства** на панели инструментов **Элементы управления**. В левом столбце написаны названия свойств, в правом – их значения.

Для изменения надписи на элементе управления используется свойство **Caption**, для изменения шрифта надписи – **Font**, для изменения цвета букв – **ForeColor**, для изменения цвета кнопки – **BackColor** и т. д. Значение свойства **Name**, определяющее имя элемента, задается только латинскими буквами.

При работе в Excel 2007 необходимо придерживаться следующего порядка действия. Вначале надо поместить на ленту вкладку **Разработчик**. Для этого следует нажать кнопку **Office** в правом верхнем углу приложения, в появившемся окне нажать кнопку **Параметры Excel** и в следующем раскрывшемся окне поставить галочку у надписи **Показывать вкладку «Разработчик» на ленте**. Затем следует активизировать данную вкладку на ленте, щелкнуть по кнопке **Вставить элементы управления** и на панели **Элементы ActiveX** выбрать нужный элемент управления.

2. Для записи программы, которая связывается с некоторым элементом управления, необходимо:

– удостовериться, что установлен режим конструктора, т. е. нажата кнопка **Режим конструктора** на панели инструментов **Элементы управления**;

– активизировать элемент управления, для которого нужно написать процедуру обработки события, и перейти в окно программного кода, щелкнув по кнопке **Исходный текст** на панели инструментов **Элементы управления** (при этом откроется окно редактора VBA). Перейти в окно программ также можно, сделав двойной щелчок по элементу управления;

– в окне программного кода выбрать из предлагаемого списка событие, при совершении которого должна выполняться программа (по умолчанию компьютер выбирает событие **Click** – щелчок).

Рассмотрим некоторые понятия визуально-событийного, объектно-ориентированного программирования. Объектами являются элементы управления, формы, рабочие листы и т. п. Объекты можно создавать и программным путем. В приложении Excel объектами являются рабочие книги (**Workbooks**), листы (**Worksheets**), ячейки (**Cells**), области (**Range**), элементы управления на рабочих листах.

Программа, выполняющаяся при совершении события, записывается в процедуру – блоке программного кода, который имеет имя,

связанное с соответствующим событием. Когда происходит то или иное событие, то выполняется процедура обработки события (программа), которая заключена между ключевыми словами (процедурными скобками) **Sub...End Sub**.

*Пример 5.1.* Разработать процедуру, связанную с элементом управления **Кнопка**, которая будет выполняться при совершении события «Щелчок по командной кнопке». Программа выводит в окно сообщения слово «Привет».

На панели **Элементы управления** надо нажать элемент управления **Кнопка** и разместить его на рабочем листе (появится кнопка с надписью **CommandButton1**). Затем следует дважды щелкнуть по кнопке и откроется редактор VBA. Текст программы:

```
Sub CommandButton1_Click()  
MsgBox ("Привет")  
End Sub
```

Здесь функция **MsgBox ()** активизирует диалоговое окно с сообщением, записанным в кавычках.

Для запуска программы надо перейти на лист рабочей книги Excel, выйти из режима конструктора, отжав кнопку **Режим конструктора** на панели инструментов **Элементы управления**, нажать на кнопку, для которой была написана процедура обработки события, и просмотреть результат.

Редактор VBA можно иначе открыть командой **Сервис/Макрос/Редактор Visual Basic**.

## 5.2. Основные конструкции VBA

При вводе текста процедур могут применяться символы как строчные, так и прописные. В любом месте программы может стоять комментарий, который начинается с апострофа «'». Длинный оператор можно разбить на несколько строк, используя пробел и символ подчеркивания «\_». В одной строке можно помещать несколько операторов, разделяя их двоеточием «:».

Имена переменных на языке VBA задаются латинскими/русскими буквами, для них нельзя использовать ключевые слова и имена стандартных объектов. Различают следующие типы переменных:

– **Integer** – для целых чисел (диапазон значений от –32 768 до 32 767), требует объем памяти в 2 байта;

– **Single** – для вещественных (дробных) чисел одинарной точности (диапазон значений от  $3.402823E + 38$  до  $1.401298E - 45$ ), требует объем памяти в 4 байта. Большие по модулю и близкие к нулю числа записываются с порядком при округлении их до 7-значащих цифр и определении их порядка в виде символа **E** с числом, что эквивалентно умножению на 10 в указанной после символа **E** степени (например,  $4.2E-3$  – это 0.0042);

– **String** – для хранения символьных (строковых) значений, каждый символ требует объем памяти в 1 байт. Длина строки символов от 1 до 64 килобайтов;

– **Variant** – для хранения любых данных, требует объем памяти в 8 байтов. Этот тип присваивается переменной по умолчанию.

Обычно перед использованием переменной производится ее объявление в операторе **Dim** со следующим синтаксисом:

**Dim** <имя переменной> **As** <тип переменной>

Допускается в одном операторе **Dim** определять несколько переменных и различные типы переменных. Например:

**Dim** i **As** Integer, x **As** Single, y **As** Single

Объявление переменной означает резервирование для нее места в памяти при выполнении программы. Допускается не объявлять переменные, в этом случае им по умолчанию присваивается тип **Variant**.

В арифметических выражениях на языке VBA записываются константы, переменные, встроенные функции, соединенные знаками арифметических операций. Арифметические операции задаются следующими символами: + (сложение), – (вычитание), \* (умножение), / (деление), ^ (возведение в степень), а сами выражения записываются в одну строку.

В табл. 5.1 приведена запись некоторых встроенных математических функций (полный перечень функций языка VBA можно найти в справочной системе).

Таблица 5.1

**Запись математических функций на языке VBA**

Математическая запись	$\sin x$	$\cos x$	$e^x$	$ x $	$\operatorname{tg} x$	$\operatorname{arctg} x$	$\ln x$	$\sqrt{x}$
Запись на языке VBA	Sin(x)	Cos(x)	Exp(x)	Abs(x)	Tan(x)	Atn(x)	Log(x)	Sqr(x)

Например, запись арифметического выражения:

$$\sin 2x + \frac{\ln(x-1) + e^{x+3}}{\sqrt[3]{x^4 + 5x^2 - 3} + \operatorname{tg}^3 x^2}$$

на языке VBA будет выглядеть следующим образом:

**Sin(2 \* x) + (Log(x - 1) + Exp(x + 3)) / ((x ^ 4 + 5 \* x ^ 2 - 3) ^ (1 / 3) + (Tan(x ^ 2)) ^ 3**

Оператор присваивания имеет следующий синтаксис:

<имя переменной> = <арифметическое выражение>

Пример записи оператора присваивания:

**x = 2 \* a + Log(a + 0.5)**

*Ввод значений переменных* может производиться в диалоговом режиме или из ячеек таблицы.

1. Диалоговый ввод осуществляется с помощью встроенной функции **InputBox**, которая при работе программы выводит на экран свое окно. Например, при выполнении строки программного кода **a = InputBox("Введите фамилию")** на экране появится диалоговое окно, в котором будет записан текст, заключенный в кавычки, а в полосе ввода числа будет мигать курсор (рис. 5.2). Нужно ввести запрашиваемое в окне значение и нажать клавишу ввода или щелкнуть мышью по кнопке **<OK>**.

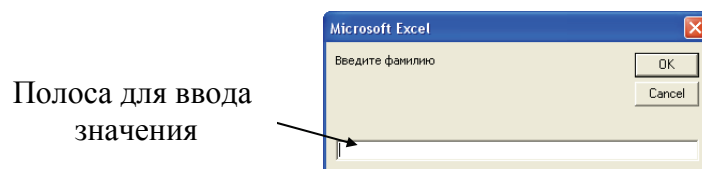


Рис. 5.2. Окно диалогового ввода функции **InputBox**

При этом возвращаемое функцией **InputBox** значение имеет тип **String**. Поэтому для ввода значения числовой переменной в программе следует дополнительно преобразовать получаемое значение к числовому типу функцией **Val**. Например, для ввода значения числа 5.25 в переменную *b* необходимо записать следующую строку программного кода:

**b = Val(InputBox("Введите значение b"))**

При вводе вещественного числа целая часть отделяется от дробной точкой. Число в степенной форме, например  $1.2 \cdot 10^{-6}$ , записывается при вводе как 1.2E-6.



2. Можно ввести значения с листа Excel. Например, чтобы поместить значение в переменную  $x$  из ячейки **A2**, можно записать:

**$x = \text{Worksheets("Лист1").Range("A2")}$**

или

**$x = \text{Worksheets("Лист1").Cells(2, 1)}$**

где **Worksheets("Лист1")** – рабочий лист Excel с именем "Лист1"; **Range("A2")** – ячейка с именем **A2**; **Cells(2, 1)** – ячейка с номером строки 2 и номером столбца 1 (т. е. ячейка **A2**).

*Вывод результатов* может осуществляться в окна сообщений и в ячейки таблицы.

1. Вывести значение переменной в окно сообщений можно с помощью функции **MsgBox**. Ее использование позволяет выводить и числа и произвольные текстовые сообщения на экран, при этом выводимое сообщение должно быть заключено в кавычки. Например, в результате выполнения строки **MsgBox ("Деление на нуль")** на экране появится окно с текстом, приведенным в кавычках.

Число, выводимое в окно, можно форматировать (указывать количество знаков до и после запятой). Для этого используется функция **Format(x, "##.##")**, где  $x$  – переменная; **##.##** – формат числа.

Текст, выводимый в окно сообщений, заключается в кавычки. Для вывода нескольких значений в одно текстовое окно используется операция конкатенации **&**, например:

**MsgBox ("Значение  $x$ " & Format(x, "##.##"))**

2. Можно вывести результаты в ячейки листа Excel. Например:

**Worksheets("Лист1").Range("A3") = x**

Здесь значение  $x$  выводится в ячейку **A3**:

**Worksheets("Лист1").Cells(3, 1) = x**

Значение  $x$  выводится в ячейку с номером строки 3 и номером столбца 1, т. е. в ячейку **A3**.

### 5.3. Линейные программы. Расчет по формулам

В линейных программах выполняются расчеты по формулам и другие действия, не требующие разветвления процессов и их повторов.

*Пример 5.2.* Для значений  $x = 15$ ,  $a = 3.75$  и  $m = 0.5 \cdot 10^{-4}$  вычислить значение  $z$  по следующим формулам:

$$w = 0.1xa(1 - m^2), \quad z = \sin \frac{w}{2 + w}.$$

В первом варианте программы ввод исходных данных организован с клавиатуры, вывод – в окно сообщений:

```

Sub CommandButton1_Click()
Dim x As Single, a As Single, m As Single
Dim w As Single, z As Single
x = Val(InputBox("Введите x"))
a = Val(InputBox("Введите a"))
m = Val(InputBox("Введите m"))
w = 0.1 * x * a * (1 - m ^ 2)
z = Sin(w / (2 + w))
MsgBox (w) : MsgBox (z) ' Два оператора на одной строчке
End Sub

```

Во втором варианте программы ввод данных организован из ячеек листа Excel, вывод – в ячейки листа. Используются объекты **Range** и **Cells**:

```

Sub CommandButton1_Click()
Dim x As Single, a As Single, m As Single
Dim w As Single, z As Single
a = Worksheets("Лист1").Range("B3")
m = Worksheets("Лист1").Range("B4")
w = 0.1 * x * a * (1 - m ^ 2)
z = Sin(w / (2 + w))
Worksheets("Лист1").Range("B8") = w
Worksheets("Лист1").Cells(2, 9) = z
End Sub

```

## 5.4. Циклические программы

Если в программе надо повторить один оператор или последовательность операторов несколько раз, используются *операторы циклов*. Операторы циклов заключают такой оператор или группу операторов между ключевыми словами начала и конца цикла (**For...Next** или **Do...Loop**) и определяют условия повтора выполнения.

Циклы **For...Next** используются, когда заранее определено, сколько раз должно выполняться повторение группы операторов:

```

For <переменная> = <нач. знач.> To <кон. знач.> Step <приращение>
  <оператор1>
  <оператор2>
  ...
Next

```

Например, чтобы вывести таблицу значений аргумента  $x$  и функции  $\sin x$  на интервале от 0 до 1 с приращением значения аргумента 0.1, необходимо записать:

```

For x = 0 To 1 Step 0.1
  y = Sin(x)
  MsgBox (y)
Next

```

Если необходимо вывести в этом фрагменте программы результаты в первый столбец, начиная с первой строки на рабочем листе с именем **Лист1**, то программа может выглядеть так:

```

i = 1
For x = 0 To 1 Step 0.1
  y = Sin(x)
  Worksheets("Лист1").Cells(i, 1) = y
  i = i + 1
Next

```

Когда **Step** отсутствует, то приращение равно единице.

Циклы типа **Do...Loop** используются тогда, когда неизвестно, сколько раз должно быть повторено выполнение группы операторов. Такой цикл продолжает работу до тех пор, пока не будет выполнено определенное условие. Существуют 4 типа операторов цикла **Do...Loop**:

1. Цикл с ключевым словом **While** продолжает свою работу, пока условие остается *истинным*, т. е. условие выполняется, и задается в двух вариантах, представленных ниже.

<b>Do While</b> <условие> <операторы > <b>Loop</b>	Условие проверяется до того, как выполнится группа операторов, образующих тело цикла
<b>Do</b> <операторы> <b>Loop While</b> <условие>	Условие проверяется после того, как операторы, составляющие тело цикла, будут выполнены хотя бы один раз

2. Цикл с ключевым словом **Until** продолжает свою работу, пока условие является *ложным*, т. е. условие не выполняется, и задается в двух вариантах, приведенных ниже.

<b>Do Until</b> <условие> <операторы> <b>Loop</b>	Условие проверяется до того, как выполнится группа операторов, образующих тело цикла
<b>Do</b> <операторы> <b>Loop Until</b> <условие>	Условие проверяется после того, как операторы, составляющие тело цикла, будут выполнены хотя бы один раз

*Пример 5.3.* Пусть необходимо заполнить первый столбец на рабочем листе результатами вычислений по формуле

$$q = \sqrt{z^2 + 5z} \cdot \ln(z + 0.33)$$

при  $z = \{1; 4; 7.5; 9; 15\}$ .

Программа:

```

Sub CommandButton1_Click()
For i = 1 To 5
z = Val(InputBox("Введите значение z"))
q = Sqr(z ^ 2 + 5 * z) * Log(z + 0.33)
Worksheets("Лист1").Cells(i, 1) = q
Next i
End Sub

```

*Пример 5.4.* Вычислить значения  $t$  по следующей формуле:

$$t = \sin^2 x + e^{3-x},$$

где  $x = 3(0.1)4$  ( $x$  меняется от 3 до 4 с шагом 0.1).

Программа:

```



Sub CommandButton1_Click()
x = 3 : n = 1
Do While x <= 4
t = Sin(x) ^ 2 + Exp(3 - x)
Worksheets("Лист1").Cells(1, n) = t
x = x + 0.1
n = n + 1
Loop
End Sub

```

В этой программе результаты размещаются в первой строке на рабочем листе Excel.

**Отладка программы.** Если при вводе оператора строчка помечается красным цветом, значит, имеется ошибка в записи оператора (отсутствуют нужные пробелы, количество закрывающих скобок не равно количеству открывающих, вместо точки в числе стоит запятая и т. п.). Надо проанализировать запись и исправить ошибки.

При возникновении ошибки во время запуска программы (например, деление на 0) происходит прерывание выполнения программы и вывод сообщения о типе ошибки. В окне сообщения можно выбрать режим остановки программы (**End**) или режим перехода в окно отладки программы (**Debug**), при котором строка с ошибкой или процедура, где возникла ошибка, будет подсвечена желтым цветом.

Для исправления ошибки можно прервать программу, нажав кнопку **Reset**  на панели инструментов редактора VBA, исправить ошибку и запустить программу заново, нажав кнопку **Run** .

### 5.5. Задание для выполнения на компьютере

1. В табл. 5.2 приведены формулы и три варианта исходных данных, по которым можно составить три программы с одними и теми же расчетными формулами. Номер варианта (строки из таблицы) для каждого студента определяет преподаватель.

Таблица 5.2

Исходные данные для расчета

№ п/п	Формулы для вычислений	Исходные данные	
		I вариант	II и III варианты
1	2	3	4
1	$h = (10r - j) / c^2 + e^{-m} \cdot \ln c;$ $y = (hm - j^2) + (0.001c)^2$	$c = 2.1;$ $r = 1.4 \cdot 10^{-4};$ $m = 7; j = 3$	$j = 0(0.1)1.7;$ $j = \{4.2; 0.3; 1.7\}$
2	$y = \sqrt{ i \cos^3 b - b^2 a };$ $z = y \operatorname{tg}(n / 4) - e^{1+b} / 100$	$a = 0.2 \cdot 10^{-3};$ $b = 8.5; n = 2;$ $i = 12$	$i = 1(0.5)10;$ $i = \{2; 1; 8.3\}$
3	$z = \sqrt{t+m} + 4e^{-2.2j};$ $w = \ln(0.8a) / (1 + 4.7z)$	$a = 1.17;$ $t = 5 \cdot 10^{-5};$ $m = 2; j = 7$	$j = 0.2(0.1)1;$ $j = \{1; 3; 6; 7; 3\}$
4	$z = \sqrt{a+1} - \operatorname{tg}(4 + j + y);$ $q = e^{-j \cdot 0.01y} + (kz)^2$	$y = -1.5;$ $k = 6; j = 3;$ $a = 2.7 \cdot 10^{-4}$	$k = -2(0.2)0;$ $k = \{8; -3; 1; 6.4\}$

1	2	3	4
5	$y = (mj + b) / \operatorname{tga} - e^{10a};$ $z = 2yb + \sqrt{a+b} / mb$	$b = 5; m = 3;$ $a = 1.4 \cdot 10^{-3};$ $j = 0.7$	$j = b(0.25)3.5;$ $j = \{0.7; -2.6; 2\}$
6	$c = 2(b+i) / k - \sqrt{2y};$ $d = e^{1-c} + 4.9(\sin^4 x^2 + 1)$	$b = -1.4; i = 3;$ $y = 1.06 \cdot 10^{-4};$ $x = -1.6; k = 6$	$i = 0(0.5)3;$ $i = \{0.9; 8.4; 2\}$
7	$v = 0.4u + b - 2\sqrt{7k + 3 / m};$ $w = 100e^{-0.2u} - \ln(80.1u)$	$b = -5; m = 4;$ $u = 0.2 \cdot 10^{-4};$ $k = 4.2$	$k = 1(0.5)5.5;$ $k = \{6; 4; 0.3; 7\}$
8	$z = ty^2 - \sqrt{i+x} / (x+1)\operatorname{tgy};$ $q = \sqrt{z^2 + 5z} \cdot \ln(y + 0.33)$	$x = 0.1; y = 2;$ $t = 6.9 \cdot 10^{-5};$ $i = 0.45$	$i = 1(0.2)2;$ $i = \{9; 1.4; 5\}$
9	$w = 0.76j + e^{ij}(\cos^3 y / n)^2;$ $s = 2\sqrt{ w - 0.1t } / (2 + n^2) - j$	$y = 2.5;$ $t = -7.4;$ $n = 9; j = 3$	$j = 0(0.1)2;$ $j = \{0.5; 8.4; 3\}$
10	$y = s / \ln(5f) / (e^{-s} + \sqrt{s+k});$ $v = (1 + my - mk) / \ln 1 + y $	$s = 7.4; m = 10;$ $f = 3.2 \cdot 10^{-4};$ $k = 0.06$	$k = 0(0.5)10;$ $k = \{4; 0.5; 8\}$
11	$w = \operatorname{tg}^3 a + (1 + 3t) + e^{a/m};$ $r = 0.9\sqrt{ 3w + j } +  a + a^2 - 1 $	$a = -1.2; m = 7;$ $t = 15 \cdot 10^{-5};$ $j = 1.4$	$j = 1(0.2)3;$ $j = \{0.5; 9.1; 5\}$
12	$d = 2\sin^2(k/a) / \cos(m+b);$ $c = d / (d^2b + 1) / (1 - e^{0.2k})$	$a = 8; m = 6;$ $b = 5 \cdot 10^{-3};$ $k = 0.002$	$k = a(-0.5)3;$ $k = \{1.6; 9.1; 8\}$
13	$y = a / (b + e^{-10b}) / (1 + ji);$ $t = y + (y+1) / \sqrt{ -1.2j }$	$b = 2.9; j = 6;$ $a = 1.5 \cdot 10^{-8};$ $i = 14$	$i = b(-0.1)2;$ $i = \{7; 4; 2; 6\}$
14	$z = \sqrt{ta + m} + 4e^{-2.2j};$ $w = \ln(0.4m) / (3 + 0.7a - z)$	$a = -1.7; y = 4;$ $t = 4 \cdot 10^{-5};$ $m = 4; j = 0.92$	$j = 0.8(0.02)1;$ $j = \{5; 3; 1; 7; 3\}$
15	$r = jx + \sqrt{1 + 20a} / (1 + 5a);$ $b = r + 12r - e^{-7.5}(x - r/n)$	$x = -1.5 \cdot 10^{-4};$ $a = 4.3; n = 15;$ $j = 8$	$n = a(0.1)5.43;$ $n = \{5.4; 2.7; 0.1\}$

Создать командную кнопку с заголовком «Расчет по формулам» и записать для нее программу для первого варианта исходных данных. Предусмотреть вывод результатов в окна сообщений.

2. Для второго варианта исходных данных написать программу с использованием операторов цикла **Do...Loop** и кнопку для этой программы надписать «Цикл Do...Loop». Предусмотреть вывод результатов в ячейки таблицы на рабочем листе Excel.

3. Создать кнопку «Цикл For...Next» и написать программу с использованием операторов цикла **For...Next** для третьего варианта исходных данных.

4. Выполнить программы и проанализировать результаты.

5. В случае, когда при выполнении программы возникают ошибки, необходимо выбрать режим отладки программы (**Debug**). Строчка, содержащая ошибку, будет подсвечена желтым цветом. Следует попытаться исправить все ошибки в программе самостоятельно.

## 6. РАЗВЕТВЛЕНИЯ В ЦИКЛИЧЕСКИХ ПРОГРАММАХ

Если некоторые действия в программе, например вычисления по формулам, должны происходить только при выполнении какого-либо условия, то такая программа называется разветвляющейся, а разветвление вычислений обеспечивается *условным оператором*.

Для записи проверяемого условия используются операции сравнения:  $>$  (больше),  $\geq$  (больше либо равно),  $<$  (меньше),  $\leq$  (меньше либо равно),  $=$  (равно),  $<>$  (не равно).

Каждое условие всегда имеет два значения: либо оно верно (**True** – правда), либо неверно (**False** – ложь). Проверяемое условие может быть и сложным, состоящим из нескольких простых условий. Для записи сложного условия применяются логические операции объединения: **And** – сложное условие верно, когда оба простых условия верны, иначе сложное условие будет ложным; **Or** – сложное условие верно, когда хотя бы одно из простых условий верно, и будет ложным только, когда все простые условия будут ложными.

Например, сложное условие  $x > y$  **And**  $a \geq 5$  будет верно только тогда, когда верны одновременно оба условия  $x > y$  и  $a \geq 5$ . Сложное условие  $x > y$  **Or**  $x = a$  будет верно тогда, когда выполняется либо условие  $x > y$ , либо  $a \geq 5$ , либо оба этих условия одновременно.

Объединяя условия логическими операциями **And** и **Or**, можно составить и более сложное условие, например:

$x \geq 1$  **And**  $x \leq 10$  **Or**  $x = 15$

Это условие будет верно, если  $x$  находится внутри интервала [1:10] либо если  $x = 15$ .

### 6.1. Условный оператор If

Условный *строчный* оператор имеет общий вид **If...Then...Else...** (если...тогда...иначе...) и представляет в общем случае конструкцию:

**If** <условие> **Then** <оператор1> **Else** <оператор2>

которая имеет следующий логический смысл: *если условие верно, то нужно выполнить оператор1, иначе – оператор2*.



В операторе может отсутствовать **Else**.

Если при проверке условия должны выполняться несколько операторов, то условный оператор записывается в *блочной* форме и заканчивается строкой с ключевыми словами **End If**:

**If** <условие> **Then**

<оператор1а>

<оператор1б>

**Else**

<оператор2а>

<оператор2б>

**End If**

Когда необходимо проверить более одного условия, можно использовать вложение операторов **If** друг в друга.

*Пример 6.1.* Программа вычислений значений  $w$  с использованием строчного условного оператора **If**:

$$y = e^{-2x} + 10; \quad z = \frac{\ln x}{x + 1}; \quad w = \begin{cases} \sqrt{xy}, & x < z^2, \\ nx + 2, & x \geq z^2. \end{cases}$$

Исходные данные:  $x = 0.5(0.5)^4$ ,  $n = 7$ . Текст программы:

```
Sub CommandButton1_Click()
Dim x As Single, y As Single, w As Single, n As Single
n = Val(InputBox("Введите n"))
For x = 0.5 To 4 Step 0.5
y = Exp(-2 * x) + 10
z = Log(x) / (x + 1)
If x < z ^ 2 Then w = Sqr(x * y) Else w = n * x + 2
Msgbox ("w=" & w)
Next
End Sub
```

*Пример 6.2.* Вычислить значение  $y$  для вводимых с клавиатуры некоторых значений  $n$  и  $x$ :

$$y = \begin{cases} \sqrt{x} & \text{при } x \geq 0, n \geq 0, \\ nx + 2 & \text{при } x < 0, n < 0. \end{cases}$$

Программа:

```
Sub CommandButton2_Click()
Dim x As Single, n As Single, y As Single
```

```

'Ввод исходных данных
x = Val(InputBox("Введите x"))
n = Val(InputBox("Введите n"))
'Проверка условий и расчет значений
If x >= 0 And n >= 0 Then y = Sqr(x)
If x < 0 And n < 0 Then y = n * x + 2
MsgBox (y) 'Вывод результата
End Sub

```

*Пример 6.3.* Вычислить значение  $y$  для вводимого с клавиатуры некоторого значения  $x$ :

$$y = \begin{cases} x + 2 & \text{при } x < 0, \\ \sqrt{5x} & \text{при } 0 \leq x \leq 5, \\ x^2 & \text{при } x > 5. \end{cases}$$

В программе использован вложенный оператор **If**:

```

Sub CommandButton3_Click()
Dim x As Single, y As Single
'Ввод исходных данных
x = Val(InputBox("Введите x"))
'Проверка условия и расчет значений
If x < 0 Then y = x + 2 Else If x <= 5 Then y = Sqr(5 * x) Else y = x ^ 2
MsgBox (y) 'Вывод результата
End Sub

```

*Пример 6.4.* С использованием блочного оператора **If** написать программу вычисления значений  $y$  и  $w$  для следующего условия: если  $x < 5$ , то  $y = \sin^2 x$ ,  $w = \operatorname{ctg} x$ ; если  $x \geq 5$ , то  $y = 1 - \sin x$ ,  $w = \operatorname{arctg} x$ .

Исходные данные  $x = \{9; 0.1; -4; 5; 12\}$  записаны на рабочем листе в первом столбце. Результаты надо поместить во второй и третий столбцы:

```

Private Sub CommandButton4_Click()
Dim x As Single, y As Single, z As Single
Dim w As Single, l As Integer
For i = 1 to 5
x = Cells(i, 1)
If x > 5 Then
y = Sin(x) ^ 2
w = Cos(x) / Sin(x)

```

```

Else
  y = 1 - Sin(x)
  w = Atn(x)
End If
Cells(i, 2) = y
Cells(i, 3) = w
Next
End Sub

```

Поскольку кнопка находится на том же рабочем листе, что и данные, то можно в программе не указывать название рабочего листа (**Worksheets(имя)**).

## 6.2. Вычисление сумм, произведений, экстремумов

Рассмотрим примеры, демонстрирующие возможности использования в программах операторов цикла и условных операторов.

*Пример 6.5.* Написать программу вычисления суммы элементов массива  $b = \{5.2; 4.5; 1; 2.9; 3\}$ . Иначе это условие можно записать:

$$s = \sum_{i=1}^5 b_i.$$

Пусть значения  $b$  записаны в первом столбце на рабочем листе. Результат надо поместить в ячейку **B1**:

```

Sub CommandButton5_Click()
Dim b As Single, s As Single, i As Integer
s = 0
For i = 1 To 5
b = Cells(i, 1)
s = s + b
Next
Range("B1") = s
End Sub

```

*Пример 6.6.* Написать программу вычисления произведения элементов массива  $t = \{0.4; -1.5; 2.8; 3; 0.9; 7.3\}$  и значения  $s$  в соответствии с формулой

$$s = 2.4 + \prod_{k=1}^6 \sin t_k.$$

Исходные данные записаны в третьем столбце на рабочем листе. Результат надо поместить в ячейку **D1**:

```
Sub CommandButton6_Click()
Dim t As Single, s As Single
Dim p As Single, k As Integer
p = 1
For k = 1 To 6
t = Cells(k, 3)
p = p * Sin(t)
Next
s = 2.4 + p
Range("D1") = s
End Sub
```

*Пример 6.7.* Определить максимальный элемент массива  $d = \{12; 0.4; 30; -2; 5; 9.3\}$  и номер этого элемента. Исходные данные записаны в пятом столбце на рабочем листе. Результат надо поместить в ячейки **F1** и **F2**:

```
Sub CommandButton7_Click()
Dim d As Single, max As Single, n As Integer, i As Integer
max = Cells(1, 5) : n = 1
For i = 2 To 6
d = Cells(i, 5)
If d > max Then max = d: n = i
Next
Range("F1") = max
Range("F2") = n
End Sub
```

### 6.3. Одномерные массивы

*Массивами* называются совокупности данных одного типа, объединенных одним именем. Элементы массивов называются индексированными переменными.

Одномерные массивы имеют один индекс, например **a(i)**, где **a** – имя массива; **i** – номер элемента массива.

Массивы до их использования в программе должны быть объявлены в операторе **Dim**, например:

```
Dim a(5) As Single
```

Здесь определено, что будет использоваться одномерный массив с шестью элементами вещественного типа одинарной точности. Число в скобках указывает номер последнего доступного для использования номера индекса. Нумерация индексов начинается с нуля.

После объявления элементы массива могут использоваться в выражениях подобно простым переменным, но с указанием индекса в круглых скобках. Например, после приведенного выше объявления массива **a(i)** в программном коде можно обращаться к следующим элементам массива: **a(0)**, **a(1)**, **a(2)**, **a(3)**, **a(4)**, **a(5)**.

Если необходимо использовать определенную нумерацию элементов массива, например с 5 до 10, то это указывается при объявлении массива следующим образом:

```
Dim a(5 to 10) As Single
```

После этого в программном коде будут доступны следующие элементы массива: **a(5)**, **a(6)**, **a(7)**, **a(8)**, **a(9)**, **a(10)**.

Ввод элементов массива может производиться с помощью оператора присваивания или в режиме диалога.

*Пример 6.8.* Программу предыдущего примера можно записать с использованием одномерных массивов следующим образом:

```
Sub CommandButton8_Click()  
Dim d(1 To 6) As Single, max As Single, n As Integer, i As Integer  
For i = 1 To 6  
d(i) = Val(InputBox("Введите элемент массива d"))  
Next  
max = d(1) : n = 1  
For i = 1 To 6  
If d(i) > max Then max = d(i): n = i  
Next  
MsgBox ("Макс. знач. = " & max & " имеет элемент с номером " & n)  
End Sub
```

Для ввода данных используется клавиатура. Результат выдается в окно сообщений.

#### **6.4. Задание для выполнения на компьютере**

1. Разработать программу по условиям, приведенным в табл. 6.1.
2. Создать кнопку «Сумма» и написать программу, вычисляющую сумму элементов массива из табл. 6.1.

3. Для командной кнопки «Произведение» создать и выполнить программу, определяющую произведение элементов массива.

4. Для командной кнопки «Максимум» создать и выполнить программу, вычисляющую максимальный элемент массива.

Таблица 6.1

Исходные данные для расчета

№ п/п	Формулы для вычислений	Исходные данные
1	2	3
1	$y = \sqrt{a + m^2 x^2} / (a + x)m;$ $z = \begin{cases} y + 1 & \text{при }  y  < 1, \\ \sin^2 y & \text{при }  y  \geq 1 \end{cases}$	$a = 1.774;$ $m = 5; \quad x = 3.2(0.2)5$
2	$d = \begin{cases} a + bi & \text{при } i > 3b, \\ \operatorname{tg} b - ai & \text{при } i \leq 3b; \end{cases}$ $z = (da / 4) / (3ab - e^{i/4} - e^{1+d} / 100)$	$b = -0.05; \quad a = 1.72;$ $i = -5(2)5$
3	$s = e^{-ax} - \ln(i / xm) / \ln^2 m;$ $p = \begin{cases} (-is)^2 & \text{при } s > 2x, \\ \sin(-6s) & \text{при } s \leq 2x \end{cases}$	$x = \{8; 1.99; 4; 0.2\};$ $a = 1.05 \cdot 10^{-4};$ $m = 4; \quad i = 7$
4	$x = (\operatorname{ctgy})^2 / (j + 2ay);$ $z = \begin{cases} e^{-j} & \text{при } x \geq \sqrt{y}, \\ (0.5y / j)^2 & \text{при } x < \sqrt{y} \end{cases}$	$y = 2.75; \quad a = -5.5 \cdot 10^{-4};$ $j = \{6; -8.1; 15; 4.2\}$
5	$p = e^{\sin(j/x)} \ln(x / y)x;$ $q = \begin{cases} \sqrt{ p/m } & \text{при } p \leq y^2, \\ \sqrt{2x} / (j + p) & \text{при } p > y^2 \end{cases}$	$y = 1.4 \cdot 10^{-3}; \quad x = 0.2;$ $m = 4; \quad j = 5(2)11$
6	$d = i + 2 / (t + i) / (1 + \sqrt{2a^2});$ $f = \begin{cases} ti & \text{при } d \geq t(t + i), \\ e^{t-d} + 9(a + i) & \text{при } d < t(t + i) \end{cases}$	$a = -4.2;$ $t = \{5; 1.75; -3; 1; 29; -1\};$ $i = 4$
7	$x = \operatorname{tg}(a^2 - 1) / (d + 1);$ $y = \begin{cases} ak + d & \text{при } 3x < ac, \\ \cos(ak) \cdot e^{a+1} & \text{при } 3x \geq ac \end{cases}$	$c = 3.7(0.1)5;$ $d = 51.9 \cdot 10^{-5};$ $a = 4; \quad k = 1$
8	$t = ai / (a^2 - b) \cdot e^{-a};$ $x = \begin{cases} 4.8 \cdot 10^{-3} + ia & \text{при } t > 5c, \\ a + i^2 t & \text{при } t \leq 5c \end{cases}$	$b = 3; \quad i = 8(4)24;$ $c = 6 \cdot 10^{-4}; \quad a = 6$

1	2	3
9	$s = w / (w^2 - j) \ln  w ;$ $y = \begin{cases} s - e^{f/i} & \text{при } s \geq 3\sqrt{f}, \\ s^2 / (w-1) & \text{при } s < 3\sqrt{f} \end{cases}$	$w = 1.1; i = 0.5;$ $f = 2.1 \cdot 10^{-2};$ $j = \{0.3; -1; 9; 2.1; -5\}$
10	$w = \begin{cases} \sqrt{0.2xk} & \text{при } x < m/2, \\ e^{-2xk} & \text{при } x \geq m/2; \end{cases}$ $v = \sqrt{w^3 +  x-a } / \ln(1+a)$	$a = 1.2; k = 3;$ $x = 7 \cdot 10^{-3};$ $m = 4(0.2)6$
11	$d = e^{-x/j} + (x-a) / \ln x;$ $b = \begin{cases} 3.6dx & \text{при } d \geq k\sqrt{a}, \\ \sin(a/x+1) & \text{при } d < k\sqrt{a} \end{cases}$	$a = 1.33 \cdot 10^{-3};$ $x = 6.8; k = 6;$ $j = \{0.2; 1; -3; -6.1\}$
12	$c = 2\text{tg}(f/2) + \ln t;$ $d = \begin{cases} ye^{-2t} + f & \text{при } c \geq 3, \\ y - y^3 & \text{при } c < 3 \end{cases}$	$f = -1.25 \cdot 10^{-6};$ $y = 1.711;$ $t = 1.1(0.1)2$
13	$y = \cos^2 x^2 /  x ;$ $z = \begin{cases}  y  & \text{при } y < axn, \\ \sqrt{ 1+e^{-y} } & \text{при } y \geq axn \end{cases}$	$a = 1.055; x = 0.6;$ $n = 6(2)14$
14	$w = \sqrt{s^3 - e^{-2s}};$ $v = \begin{cases} \cos^2 jx & \text{при } 2s > xj, \\ 2\text{tg} jx & \text{при } 2s < xj, xj \leq 3s, \\ 5 - e^{x/2} & \text{при } xj > 3s \end{cases}$	$s = 5.2;$ $x = \{5; 0.04; -2.1; 6\};$ $j = 10.4 \cdot 10^4$
15	$b = 12s - e^{-s/2}(x-j);$ $z = \begin{cases} \sqrt{-2xj} + b & \text{при } b < 1.5, \\  13xj  + b & \text{при } b \geq 1.5 \end{cases}$	$x = -4 \cdot 10^{-3}; s = 1.1;$ $j = 4(0.5)7$

5. Для самостоятельного освоения способов программирования различных задач написать программы для всех условий из табл. 6.2.

Таблица 6.2

## Исходные данные для расчета

№ п/п	Формулы для вычислений	Исходные данные
1	2	3
1	$h = d + \sum_{i=1}^n a_i^2$	$n = 5; d = 12.55 \cdot 10^{-4};$ $a = \{0.8; 4; -0.17; 2; 0.9; 1\}$

1	2	3
2	$g = c \prod_{j=1}^m (b_j + 1)^2$	$m = 4; c = -0.0045;$ $b = \{0.49; 0.55; -2; -0.71\}$
3	$z = 0.1x_3 + \sum_{i=1}^m (x_i - 2)^2$	$m = 5;$ $x = \{-2.1; 0.6; 1.1; 2.7; 4\}$
4	$f = k!;$ $g = f / \sum_{i=1}^5 (a_i / i)$	$k = 6;$ $a = \{2.3; 7; -7.2; -4; 9\}$
5	$w_i = \begin{cases} a + v_i & \text{при } v_i > 0, \\ b / v_i & \text{при } v_i \leq 0 \end{cases}$	$a = 1.75; b = -4.15;$ $v = \{1; 1.5; -4; -12.9; 3\}$
6	$q = \sum_{i=1}^n (x_i y_i)$	$n = 5;$ $x = \{-2.7; -5; 4; 3.5; -7.7\};$ $y = \{2; 3; -1.5; -2; 1\}$
7	$c_i = \sqrt{a_i + b_i};$ $d = \sum_{i=1}^n (c_i / i)$	$n = 6;$ $a = \{0.5; 2; 2.5; 1; 0; 0.7\};$ $b = \{2.3; 4; 0.5; 2; 3; 9\}$
8	$d = \sum_{i=1}^{n_1} a_i - c \sum_{i=1}^{n_2} (b_i - 1)^2$	$n_1 = 6; n_2 = 5;$ $a = \{0.8; 12; -4; 39.2; 3; 0.4\};$ $b = \{19; 1; -24; 4.2; 8\}$
9	$a_i = (de^{-x_i} \sin x_i) / (\sqrt{5 + \cos x_i})$	$d = 51.05 \cdot 10^{-4};$ $x = \{0.7; 6; -7; 0.9; -0.2; 1\}$
10	$a = \sum_{i=1}^n x_i \text{ при } x_i > 0;$ $b = \prod_{i=1}^n x_i \text{ при } x_i \leq 0$	$n = 6;$ $x = \{3; -2; 0.7; -4.1; -2; 7\}$
11	$s = a + \sum_{i=1}^n x_i^2 / (i+1);$ $q = s \prod_{i=1}^n y_i / (i^2 + 1)$	$a = 5.45; n = 5;$ $x = \{-4; 2.1; -2; 4; -3\};$ $y = \{2.1; 7.7; -4.9; 5; 9\}$
12	$z = \max \{x_i\};$ $y = z \sum_{i=1}^5 x_i^2$	$x = \{1; 2.7; 4.7; 6; 10\}$



1	2	3
13	$p = \min\{y_i\};$ $q = \prod_{i=1}^n (y_i - 5) + p$	$y = \{3; -2; 0.9; 0.75; 1\}$
14	$z = \begin{cases} \sin^2 c & \text{при } \sum_{i=1}^6 y_i > c, \\ \cos^2 c & \text{при } \sum_{i=1}^6 y_i \leq c \end{cases}$	$y = \{4; -6; 3; -3; 9; 11\};$ $c = 10.1$
15	$q = t + \frac{x_1 + 1}{x_1} + \frac{x_2 + 1}{x_2} + \dots + \frac{x_6 + 1}{x_6}$	$t = 0.45;$ $x = \{1.1; 6.2; 3; -4; 6; 1\}$

## 7. ФУНКЦИИ И ПРОЦЕДУРЫ ПОЛЬЗОВАТЕЛЯ

### 7.1. Функции пользователя

Если имеются однотипные вычисления в программе, то целесообразно выделять операторы, реализующие их, в отдельную пользовательскую подпрограмму. Тогда в так называемой основной программе можно будет не писать каждый раз заново похожие операторы, а просто обращаться к подпрограмме.

Различают два типа пользовательских подпрограмм:

- подпрограммы типа **Function** (функция), которые производят действия, определенные ее программным кодом, и в результате этого действия *возвращают одно* значение определенного типа;
- подпрограммы типа **Sub** (процедуры), которые производят действия, определенные ее программным кодом, и *возвращают столько значений, сколько необходимо*.

Функция пользователя в общем виде имеет следующий вид:

```
Function <имя функции> (<аргум.1> <As тип>, ...) <As тип>  
<программный код для определения функции>  
End Function
```

Описание функции пользователя начинается с заголовка, в котором указывается имя функции после ключевого слова **Function**, в скобках через запятую перечисляются формальные аргументы с указанием их типа и после скобок записывается тип данных, которому принадлежит значение, возвращаемое функцией. При этом типы аргументов функции и самой функции можно не указывать, в этом случае им присваивается тип **Variant**. *Конечный результат помещается в переменную с тем же именем, что и имя функции.*

Заканчивается описание функции пользователя ключевыми словами **End Function**.

Например, функция  $v(h, r)$ , вычисляющая объем цилиндра  $v$  в зависимости от его высоты  $h$  и радиуса основания  $r$ , может быть задана в следующем виде:

```
Function v(h As Single, r As Single) As Single  
v = 3.14 * h * r ^ 2  
End Function
```

Для обращения к функции пользователя в так называемой основной программе записывается имя функции, а в скобках – фактические аргументы, которые *должны соответствовать формальным по количеству, типу и порядку расположения*.

Например, ниже приведен фрагмент программы вычисления объема тела из двух поставленных друг на друга цилиндров высотой по 10 см и радиусами основания соответственно 3 и 5 см:

```
Dim vol As Single  
vol = v(10, 3) + v(10, 5)  
MsgBox ("Объем равен" & vol & "см³")
```

Необходимо отметить, что в скобках перед аргументами функции выражением **ByVal** можно указать, что фактический аргумент как параметр передается в функцию по его значению, иначе, по умолчанию, параметр будет передаваться в функцию по ссылке **ByRef**.

Дело в том, что когда по умолчанию значения формальных параметров передаются в процедуру по ссылке, то VBA для них не отводит дополнительного места в памяти. Поэтому формальные параметры не являются настоящими переменными – это только ссылки на значения соответствующих переменных из вызывающей программы. Если же в ходе выполнения функции пользователя эти параметры меняют свои значения, то в итоге может быть получен неверный результат. Чтобы этого не произошло, используется передача формального параметра в процедуру по его значению с помощью записи **ByVal** перед именем параметра. В этом случае VBA будет хранить в памяти копию оригинала значения формального параметра как переменной и ее значения останутся неизменными после выполнения процедуры.

Например, заголовок функции может иметь следующий вид:

```
Function F(ByVal x As Single, n As Integer) As Single
```

Здесь объявляется функция **F** вещественного типа (**Single**), зависящая от двух аргументов: **x** – вещественного типа (**Single**), передаваемого в процедуру по значению, и **n** – целочисленного типа (**Integer**), передаваемого в процедуру по ссылке.

*Пример 7.1.* Функция пользователя  $F(k)$  для расчета факториала числа  $k$  имеет вид:  $k! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot k$ .

Программа:

```
Function F(k)  
Dim i As Integer
```

```

F = 1
For i = 2 To k
F = F * i
Next i
End Function

```

Теперь для вычисления выражения  $C = M!N! / (M + N)!$  можно использовать следующую программу:

```

Sub CommandButton1_Click()
Dim C As Single
Dim N As Integer, M As Integer
M = Val(InputBox("Введите M"))
N = Val(InputBox("Введите N"))
C = F(M) * F(N) / F(M + 1)
MsgBox C
End Sub

```

## 7.2. Создание пользовательской функции листа

Excel позволяет создавать пользовательские функции листа, написанные на языке VBA и реализующие алгоритм пользователя. После создания функции ее имя помещается в категорию функций «Определенные пользователем», которая отображается в окне **Мастера функций** и воспринимается Excel как встроенная функция. Функция вводится в ячейки Excel по обычным правилам ввода функции.

Для создания пользовательской функции листа нужно на листе редактора VBA выполнить команду **Вставка/Модуль (Insert/Module)** и написать функцию пользователя в появившемся окне.

*Пример 7.2.* Создать функцию, которая вычисляет значения  $\sin x$ , когда аргумент  $x$  представлен в градусах. Текст программы:

```

Function MySin(x As Single) As Single
'Определение константы
Const pi As Single = 3.14159
Dim y As Single
'Перевод значения x в радианы
y = (x / 180) * pi
MySin = Sin(y)
End Function

```

Вызов функции будет иметь вид, представленный на рисунке.

	А	В
1	Угол	Синус угла
2	0	=mysin(A2)
3	30	=mysin(A3)
4	60	=mysin(A4)
5	90	=mysin(A5)
6	180	=mysin(A6)
7		

Использование пользовательской функции листа

### 7.3. Процедуры пользователя типа **Sub**

Общая структура процедуры имеет следующий вид:

```
Sub <имя процедуры> (<параметр1>, <параметр2>, ...)  
<программный код для определения процедуры>  
End Sub
```

Часть формальных параметров, описанных в заголовке, может быть использована для ввода исходных данных, а другая часть – для записи в них результатов выполнения программного кода процедуры.

В частном случае процедура может иметь и пустой список параметров, тогда ее заголовок будет выглядеть следующим образом:

```
Sub <имя процедуры> ( )
```

Для того чтобы использовать процедуру типа **Sub** в основной программе, надо записать ее имя в отдельной строке и через пробел после имени указать фактические аргументы, разделив их запятыми.

По существу в предыдущих разделах рассматривались процедуры, но процедуры обработки событий, привязанные к объекту, над которым это событие совершается. Например, запись **Sub CommandButton1\_Click()** означает, что программа выполняется, когда по кнопке **CommandButton1** делается щелчок **Click**.

Пользовательские процедуры могут быть доступны как для отдельной формы, так и для всех форм приложения. Если текст пользовательской процедуры записывается в окне программного кода рабочего листа, то она будет доступна во всех процедурах данного листа, но недоступной в программах других листов. Если же текст пользовательской процедуры записывается в специальном окне модуля (**Insert/Module**) Редактора **Visual Basic**, то она будет доступна для процедур всей книги.

*Пример 7.3.* Рассмотрим пример использования процедуры со списком формальных параметров. Предположим, необходимо вычислить сумму двух различных произведений:

$$S = \prod_{k=1}^4 f_k + \prod_{k=2}^4 \sin f_k,$$

где  $f_k = \{5; -3; 1; 2\}$ .

Для решения такой задачи сначала надо создать процедуру, которая производит вычисление произведений любого количества любых сомножителей из элементов массива:

```
Sub product(k, z, p)  
Dim i As Integer  
p = 1  
For i = 1 To k  
p = p * z(i)  
Next  
End Sub
```

Здесь введены следующие имена для процедуры и параметров: **product** – имя процедуры; **k** – количество сомножителей в произведении; **z** – имя массива, произведение элементов которого перемножается; **p** – параметр, в котором после расчета будет храниться значение вычисленного произведения.

Эту процедуру можно использовать для решения исходной задачи:

```
Sub CommandButton1_Click()  
Dim f(4) As Single, f1(4) As Single, S As Single  
Dim p As Single, p1 As Single, i As Integer  
For i = 1 To 4  
f(i) = Val(InputBox("Введите f(i)"))  
f1(i) = Sin(f(i))  
Next  
'Обращение к процедуре  
product 4, f, p1  
product 3, f1, p2  
S = p1 + p2  
MsgBox S  
End Sub
```

Здесь после ввода в память массивов **f** и **f1** дважды вызывается процедура **product**. При этом первый раз в нее передаются фактические значения для формальных параметров **k** и **z**, равные соответ-

венно 4 и **f**, а результат сохраняется в переменной **p1**. Во второй раз – значения параметров **k** и **z**, равные соответственно 3 и **f1**, а результат сохраняется в переменной **p2**.

#### 7.4. Задание для выполнения на компьютере

1. Записать пользовательскую функцию листа для вычисления любого арифметического выражения и опробовать ее работу.

2. Создать на рабочем листе две кнопки. Одну кнопку использовать для программы вычисления выражений, представленных в таблице.

##### Исходные данные для расчета

№ п/п	Условие задачи	Исходные данные
1	2	3
1	$D = \frac{\prod_{i=1}^6 x_i + \prod_{j=1}^5 x_j^2}{150 + \prod_{k=1}^4 (x_k + 1)}$	$x = \{4; 1; 3; -2; -1; 0.1\}$
2	$W = \frac{\sum_{i=1}^7 (x_i^2 + 5)}{\sum_{i=1}^4 (x_i^2 + 5) \sum_{i=1}^5 (x_i + 5)}$	$x = \{-2; -1.5; 0.1; 2.5; 2.7; 4.1; 6\}$
3	$E = \frac{1}{2} \left( \prod_{j=1}^6 y_j^2 + \prod_{i=2}^n y_i^2 \right)$	$n = 5; y = \{1; -3; 0.2; 5; 8; -4\}$
4	$S = a \sum_{i=1}^4 x_i + a^2 \sum_{i=1}^5 x_i^2$	$a = 1.2; x = \{2; 4; -3; 1; 7\}$
5	$T = \prod_{i=2}^5 (F_i + 1) - \prod_{j=1}^m (F_j + 1)$	$m = 7; F = \{2; -3; 1; 4; -1; 0.2; 7\}$
6	$W = \sum_{i=1}^7 x_i^2 + \sum_{i=3}^6 (x_i + 3)^2$	$x = \{5; -1; -2; 4; 3; 2; 6\}$
7	$A = \prod_{i=N}^M O_i^2 + B + \prod_{i=N+1}^M O_i^3$	$N = 2; M = 5; B = 3.4;$ $O = \{1; 2; -4; 5; 0.3\}$
8	$B = \frac{\sum_{k=1}^4 x_k}{\sum_{i=1}^3 x_i^2} + \sum_{i=1}^4 (x_i + 1)^2$	$x = \{20; -10; 5; 4\}$
9	$C = D \sum_{i=1}^4 T_i^2 + \sum_{i=1}^5 T_i$	$D = 0.3; T = \{4; 1; 0.2; 5; 8\}$

Окончание таблицы

1	2	3
10	$D = \prod_{k=2}^4 A_k + \prod_{k=1}^3 A_k^2$	$A = \{14; -12; 7; 1\}$
11	$T = \frac{B \prod_{i=1}^5 (y_i^2 - 1)^2}{\prod_{i=1}^4 (y_i^2 + 1)}$	$B = 0.2; y = \{2; 3; 7; 8; 9\}$
12	$D = AB \left( \sum_{c=1}^4 K_c - \sum_{c=1}^4 (K_c + 3)^2 \right)$	$A = 2.3; B = 4; K = \{2; 3; 1; -4\}$
13	$A = \sum_{i=1}^3 A_i D_i + \sum_{j=1}^3 (A_j + D_j)^2$	$A = \{5; 1; -3\}; D = \{4; 6; 2\}$
14	$C = \frac{\prod_{i=1}^6 A^2}{\prod_{i=3}^8 A^{-2}} + T$	$T = 12.8; A = \{5; 6; 1; 4; 7; -3; 1; 4\}$
15	$W = \prod_{k=1}^4 P_k + \prod_{k=2}^4 (P_k + 1)$	$P = \{6; 5; 2; 3\}$

В программном коде записать основную программу и процедуру типа **Sub**. В процедуре вычислить сумму или произведение. Вторую кнопку использовать для решения этой же задачи с функцией пользователя.



## 8. ФОРМЫ ПОЛЬЗОВАТЕЛЯ

Управление электронными документами в Excel часто производится с помощью вспомогательных диалоговых окон языка VBA, которые в проекте документа называются пользовательскими формами **UserForms**.

Для того чтобы в проект документа включить диалоговую пользовательскую форму, необходимо в редакторе VBA выполнить команду **Вставка/UserForm (Insert/UserForm)**. После этого в проект документа добавится объект **UserForm1**, а на экране появится заготовка для формы пользователя с именем **UserForm1** и панель элементов **Toolbox**, почти совпадающая с панелью инструментов **Элементы управления**.

На форме пользователя так же, как и на рабочем листе, можно создавать кнопки, текстовые поля, надписи, списки и другие объекты.

### 8.1. Свойства и методы

Состояние объекта можно изменить двумя способами: либо изменить значение *свойств* (**Properties**), принадлежащих объекту (например, для текстового окна изменить его размер, цвет, тип шрифта и т. п.), либо применить к нему некоторые действия, так называемые *методы* (**Methods**), определенные для данного объекта (например, показать форму, сдвинуть текстовое поле и т. п.).

Задание конкретных значений свойств объекта осуществляется в окне свойств или программным путем. В последнем случае используется следующий синтаксис:

объект.свойство = значение

Например, можно задать заголовок окна (свойство **Caption**) формы с именем **UserForm1** (свойство **Name**) в программном коде следующим образом:

**UserForm1.Caption = "Список сотрудников"**

Чтобы задать нужный размер букв (свойство **Size**) для определенного шрифта (свойство **Font**), составляющих заголовков формы, необходимо записать следующую строку в программном коде:

## **TextBox1.Font.Size = 14**

Для применения некоторого метода действия на объект используется следующий синтаксис:

объект.метод

Например, для появления на экране пользовательской формы **UserForm1** следует в программном коде применять метод **Show (UserForm1.Show)**, а чтобы убрать форму – метод **Hide**.

Перечислим еще некоторые свойства объектов.

**BackColor** – цвет фона. Выбрав свойство в окне свойств, необходимо щелкнуть в правой части, появится маленькая кнопка, нажатие на которую вызовет появление палитры цветов. Цвет выбирают щелчком квадратика с нужной окраской. При изменении свойства в программе его значение определяется функцией **RGB**. В этой функции задаются цвета цифрами от 0 до 255: оттенки красного – на первом месте (red), оттенки зеленого (green) – на втором, оттенки синего (blue) – на третьем.

Например, можно создать форму **UserForm1** и на рабочем листе создать кнопку со следующей программой:

```
Private Sub CommandButton1_Click()  
UserForm1.BackColor = RGB(255, 0, 0)  
UserForm1.Show  
End Sub
```

При выполнении этой программы появляется форма, окрашенная в красный цвет.

**Caption** – заголовок. Заголовком формы является текст, выводимый в строку заголовка окна формы.

**Font** – шрифт. Для задания значения этому свойству необходимо выбрать его в списке свойств и раскрыть стандартное диалоговое окно выбора шрифтов, нажав кнопку с многоточием. При установке шрифта программным путем используются следующие свойства: **FontBold** – полужирный, **Fontlitalic** – курсив, **FontUnderline** – подчеркнутый. Они принимают значения **True** или **False**. Например, оператор **UserForm1.TextBox1.FontBold = True** определяет шрифт в текстовом окне как полужирный.

**FontSize** – размер шрифта. **ForeColor** – основной цвет. Это свойство задает цвет текста. **Height** – высота окна формы, **Width** – ширина окна формы. **Name** – имя формы. Это свойство определяет имя формы, по которому можно обращаться к ней в программном коде.

**Picture** – отображение рисунка. Свойство позволяет отобразить на форме рисунок, файл с которым можно открыть, нажав кнопку в правой части.

**Top** – координата верхней границы, **Left** – координата левой границы окна формы на экране.

Элементы управления, такие как кнопки, поля, надписи и др., имеют и вышеперечисленные свойства и собственные свойства. Например, текстовое поле имеет свойство **Text**, в котором записывается содержимое поля. Полный перечень свойств объектов можно найти в справочной системе VBA.

## 8.2. Примеры использования форм

*Пример 8.1.* Рассмотрим пример работы с формой и элементами управления. На рабочем листе создадим кнопку с именем **Работа с формой** и форму, изображенную на рис. 8.1.

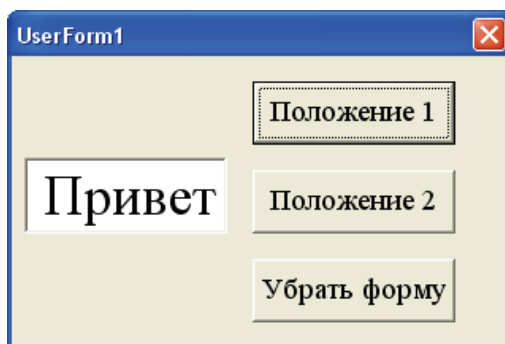


Рис. 8.1. Форма пользователя

На форме размещено поле (**TextBox1**) и три кнопки. Названия на этих кнопках можно ввести при создании кнопок, щелкнув на имени, либо, можно в контекстном меню выполнить команду **Properties** и в свойстве **Caption** ввести нужное наименование кнопки.

Программа для первой кнопки **Положение 1**:

```
Sub CommandButton1_Click()  
UserForm1.TextBox1.Text = "Привет"  
UserForm1.TextBox1.Top = 10  
UserForm1.TextBox1.Left = 10  
UserForm1.TextBox1.FontSize = 10  
UserForm1.TextBox1.ForeColor = RGB(0, 0, 255)  
End Sub
```

Здесь в текстовое окно помещается слово «Привет». Программным путем меняются свойства **Top** и **Left**, определяющие координаты верхнего левого угла текстового поля, выбирается размер шрифта в поле и цвет. **RGB(0, 0, 255)** задает красный цвет.

Программа для второй кнопки **Положение 2** меняет расположение поля, размер и цвет текста в поле:

```
Sub CommandButton2_Click()
UserForm1.TextBox1.Text = "Привет"
UserForm1.TextBox1.Top = 80
UserForm1.TextBox1.Left = 10
UserForm1.TextBox1.FontSize = 20
UserForm1.TextBox1.ForeColor = RGB(255, 0, 0)
End Sub
```

Программа для третьей кнопки убирает форму с экрана:

```
Sub CommandButton3_Click()
UserForm1.Hide
End Sub
```

Программа для кнопки **Работа с формой** на рабочем листе вызывает появление формы на экране:

```
Sub CommandButton3_Click()
UserForm1.Show
End Sub
```

С нажатия этой кнопки можно начать работу. На появившейся форме попеременно нажимая кнопки **Положение 1** и **Положение 2**, можно понаблюдать за изменениями на экране.

*Пример 8.2.* Рассмотреть пример движения элемента управления через определенные промежутки времени.

Пусть на форме имеется текстовое поле. Программа, приведенная ниже, производит следующие действия: поле окрашивается в зеленый цвет, определяется положение поля на форме и оно меняется 7 раз через 1 с:

```
Sub CommandButton1_Click()
UserForm1.TextBox1.BackColor = RGB(0, 255, 0)
For i = 10 To 70 Step 10
UserForm1.TextBox1.Top = 10 + i
UserForm1.TextBox1.Left = 10 + i
If Application.Wait(Now + TimeValue("0:00:1")) Then
```

**End If**  
**Next**  
**End Sub**

Встроенная функция **Now** определяет текущее время, с помощью встроенной функции **TimeValue** задается интервал времени (в нашем примере – 1 с), метод **Wait** в приложении (**Application**) переводит компьютер в режим ожидания на некоторый отрезок времени.

*Пример 8.3.* Пусть разработана форма, представленная на рис. 8.2.

Рис. 8.2. Ввод информации и вывод результатов

Программа для кнопки **Вывод результатов** вводит значение **x** из верхнего поля, рассчитывает значения **a**, **b**, **z** и выводит их в поля **TextBox1**, **TextBox2**, а значение **z** выводит в надпись **Label2**, расположенную в нижней части формы **UserForm1**:

```
Sub CommandButton1_Click()  
Dim x As Single, y As Single  
Dim a As Single, b As Single, z As Single  
x = Val(UserForm1.TextBox1.Text)  
y = Format(x, "#####.###")  
a = (x + y) ^ 2 : b = Sin(a) - Sin(b) ^ 3  
UserForm1.TextBox2.Text = "a=" & a  
UserForm1.TextBox3.Text = "b=" & b  
UserForm1.Label2.Caption = "z=" & 5 * Sin(10) / 3  
End Sub
```

Программу можно запустить нажатием кнопки **Run** на листе редактора Visual Basic.

### 8.3. Работа со списками

VBA предоставляет возможность использования элементов управления, служащих для хранения и обработки одномерных массивов. Это простые списки **ListBox** и комбинированные (раскрывающиеся) списки **ComboBox**. Комбинированные списки, в отличие от простых, содержат, кроме собственно списка, еще и текстовое поле, через которое можно записывать и вводить данные в список, добавляя последовательно к существующим новые элементы массива.

Основные *свойства* списков: **List(i)** – *i*-й элемент в списке; **ListIndex** – номер выбранного элемента в списке; **ListCount** – количество элементов в списке; **RowSource** – источник заполнения списка **ComboBox** или **ListBox**.

Основные *методы* обработки списков: **AddItem** – добавить элемент в список; **RemoveItem** – удалить указанный элемент из списка; **Clear** – удалить весь массив из списка.

Например, чтобы добавить элемент **Cos(4)** в список **List1**, необходимо записать (и затем выполнить) строку кода: **List1.AddItem Cos(4)**.

Чтобы удалить из комбинированного списка **Combo1** второй элемент, необходимо записать **Combo1.RemoveItem 2**.

### 8.4. Использование комбинированного списка

*Пример 8.4.* Пусть имеется список группы, в котором содержатся сведения об оценках, полученных студентами во время сессии (рис. 8.3).

	A	B	C	D	E	F
1			Список группы			
2						
3	<b>№</b>	<b>Ф.И.О.</b>	<b>Матем.</b>	<b>История</b>	<b>Физика</b>	<b>Химия</b>
4	1	Акатова А.О.	6	5	6	7
5	2	Петров С.Л.	7	9	6	8
6	3	Мицкевич Н.И.	6	5	7	6
7	4	Вилков С.Д.	5	5	4	4
8	5	Дроздов А.К.	9	9	8	8
9						
10						
11			Средний балл			
12						
13						

Рис. 8.3. Рабочий лист Excel со списком группы

Требуется подсчитать средний балл для конкретного студента, выбранного из списка.

На рабочем листе с помощью панели **Элементы управления** создадим кнопку с надписью «Средний балл».

Для выбора фамилии студента разработаем форму, представленную на рис. 8.4. На этой форме с именем **UserForm1** создадим комбинированный список **ComboBox1** с надписью «Раскройте список, выберите фамилию, нажмите кнопку» в элементе управления **Label1**.

Командная кнопка с заголовком «Расчет среднего балла» запускает программу расчета. В поле **TextBox1** выводится полученный результат, перед этим полем в метке **Label2** сделана надпись «Средний балл».

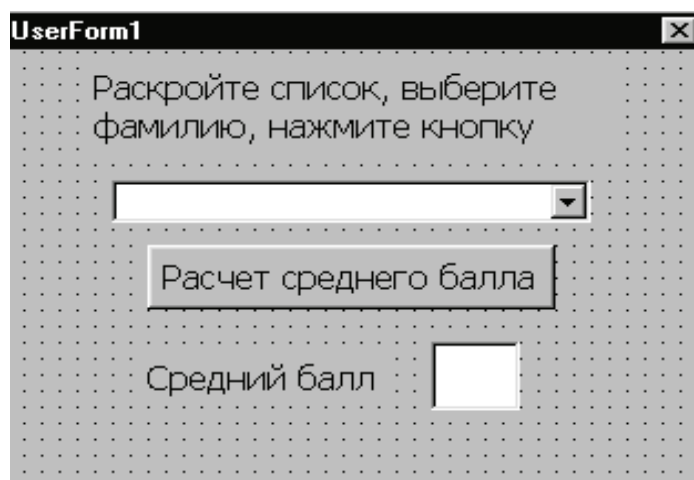


Рис. 8.4. Пользовательская форма

Для кнопки «Средний балл», которая находится на рабочем листе **Лист1**, запишем программу:

```
Sub CommandButton1_Click()  
I = 4  
Do Until Worksheets("Лист1").Cells(I, 1) = ""  
I = I + 1  
Loop  
J = I - 1  
UserForm1.ComboBox1.RowSource = "Лист1!b4:b" & J  
UserForm1.Show  
End Sub
```

Здесь "" означает пустую ячейку. В переменной **J** определяется номер последней заполненной строки в таблице. Затем с помощью метода **RowSource** заполняется комбинированный список **ComboBox1**

информацией из ячеек **b4:bJ**. С помощью метода **Show** форма вызывается на экран.

Для кнопки «Расчет среднего балла» запишем программу:

```
Sub CommandButton1_Click()  
n = UserForm1.ComboBox1.ListIndex + 1  
s = 0  
For i = 1 To 4  
b = Worksheets("Лист1").Cells(n + 3, i + 2).Value  
s = s + b  
Next  
a = s / 4  
UserForm1.TextBox1.Text = a  
End Sub
```

В этой программе определяется номер выбранного элемента списка посредством свойства **ListIndex**, а затем высчитывается средний балл. Полученный результат помещается в текстовое окно **TextBox1** на форме.

## 8.5. Работа с простым списком

*Пример 8.5.* Пусть требуется по результатам сессии определить фамилии двоечников для списка студентов (рис. 8.3 на с. 70).

Создадим форму пользователя, в заголовке которой напомним «Список двоечников» (в свойстве **Caption**), имя формы изменим на **Neud** (в свойстве **Name**). На форме разместим достаточно большой простой список (**ListBox1**). Тогда программа для некоторой кнопки на рабочем листе может выглядеть следующим образом:

```
Sub CommandButton1_Click()  
i = 4  
Do Until Worksheets("Лист").Cells(i, 1) = ""  
For j = 1 To 4  
oc = Worksheets("Лист1").Cells(i, j + 2)  
If oc < 4 Then  
Neud.ListBox1.AddItem Worksheets("Лист1").Cells(i, 2)  
GoTo 1  
End If  
Next j  
1: i = i + 1  
Loop  
End Sub
```



Здесь  $i$  – номер строки в таблице на рабочем листе **Лист1** со списком группы, он начинается с 4, так как фамилии студентов записываются с четвертой строки.

В цикле **Do...Loop** просматривается содержимое списка студентов по строчкам до тех пор, пока не встретится пустая ячейка.

В цикле **For...Next** проверяются оценки каждого студента и, если встретилась оценка ниже 4, то фамилия такого студента помещается в список на форме. При этом проверять другие оценки этого студента уже не нужно, поэтому оператор **GoTo** выполняет переход по метке 1, где наращивается номер строки в таблице. В качестве метки может использоваться любое сочетание буквенных и цифровых символов.

## 8.6. Задание для выполнения на компьютере

1. Опробовать приведенные выше примеры 8.1–8.3. Внести изменения в программы с тем, чтобы изменить параметры шрифта, цвет, направление движения элементов управления.

2. Для программ, разработанных по условиям табл. 5.2 (см. на с. 45–46), организовать ввод исходных данных из полей формы и размещение полученных результатов в поля или надписи на форме.

3. Опробовать примеры 8.4 и 8.5. Добавить в таблицу (см. рис. 8.3 на с. 70) дополнительные сведения о годе рождения студентов, адресе и т. п.

4. Разработать пользовательскую форму, в которой выбиралась бы фамилия студента и определялся год рождения, адрес или другие параметры. Записать программы для кнопок, размещенных на рабочем листе и на пользовательской форме.

## 9. ПРИМЕР РАЗРАБОТКИ ИНЖЕНЕРНОГО ПРОЕКТА

### 9.1. Постановка и решение задачи

Пусть на рабочем листе с названием **Товар** располагается таблица вида.

Наименование	Цена, руб.	Количество
Стул	80 000	50
Стол	120 000	40
Кресло	200 000	15
Шкаф	1200 000	20

Шапка таблицы записана в первой строке. На рабочем листе **Заказчики** хранится информация о заказчиках.

Фирма	Адрес	Телефон	Расчетный счет
Рассвет	Минск, ул. Коржа, 2	211-23-34	10010007
Восток	Витебск, пр-т Мира, 5	222-45-67	10020009
Вест	Гродно, ул. Кирова, 1	333-45-78	10030008
Луч	Гомель, ул. Буре, 4	444-56-90	10040005

Требуется сформировать информацию о заказанном товаре на рабочем листе **Платеж**, приведенном на рис. 9.1.

	A	B	C	D	E
1				Фирма	"Белмебель"
2				Адрес	ул. Кленовая, 5
3				Телефон	291-00-11
4				Расчетный счет	100200449
5					
6				Платежная ведомость	
7					
8				Заказчик	
9				Адрес	
10				Расчетный счет	
11					
12	Наименование	Цена	Количество	Общая стоимость	
13					
14					
15					
16					
17	Дата				
18					

Рис. 9.1. Рабочий лист **Платеж**

Для организации диалога при оформлении заявки заказчиком надо создать диалоговое окно **UserForm1** с интерфейсом, представленным на рис. 9.2.

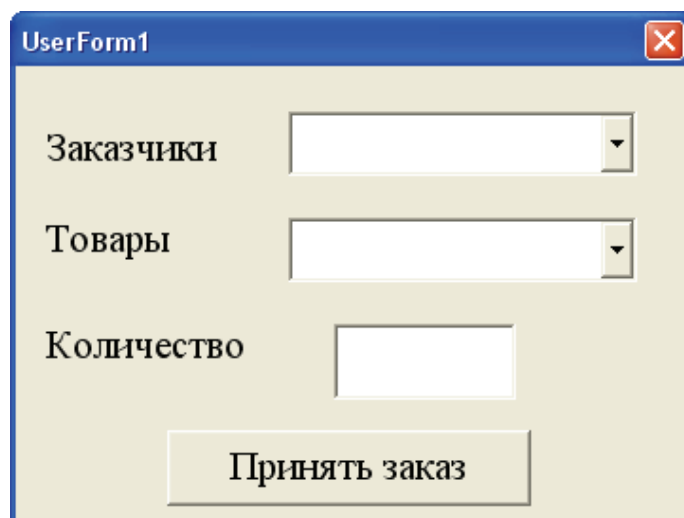


Рис. 9.2. Форма для выбора товара

На рабочем листе **Платеж** следует расположить кнопки с названиями «Заказ товара» и «Очистка» вне области с информацией. Командная кнопка «Заказ товара» должна заполнить раскрывающиеся списки на форме названиями товаров и фирм-заказчиков исходя из информации, имеющейся на листах **Товары** и **Заказчики**, и вызвать на экран форму **UserForm1**:

```
Sub CommandButton1_Click()  
k = 2  
Do While Worksheets("Заказчики").Cells(k, 1) <> ""  
UserForm1.ComboBox1.AddItem Worksheets("Заказчики").Cells(k, 1)  
k = k + 1  
Loop  
k = 2  
Do While Worksheets("Товары").Cells(k, 1) <> ""  
UserForm1.ComboBox2.AddItem Worksheets("Товары").Cells(k, 1)  
k = k + 1  
Loop  
UserForm1.Show  
End Sub
```

В тексте программы учтено, что первая строка на рабочих листах **Товары** и **Заказчики** содержит названия столбцов, и поэтому цикл

начинается со значения **k = 2**. При этом циклы для заполнения комбинированных списков **Заказчики** и **Товары** практически совпадают, однако их нельзя объединить в один, поскольку количество заказчиков и товаров может быть разным.

Командная кнопка «Очистка» на листе **Платеж** должна очищать те ячейки, которые будут заполняться после выполнения диалога в окне **UserForm1**. Для очистки ячеек используется метод **ClearContents**, а область очищаемых ячеек задается объектом **Range**:

```
Sub CommandButton2_Click()  
Worksheets("Платеж").Range("b8:b10").ClearContents  
Worksheets("Платеж ").Range("a13:d13").ClearContents  
Worksheets("Платеж ").Range("b17").ClearContents  
End Sub
```

Поскольку кнопка «Очистка» находится на листе **Платеж**, то указание имени листа (**Worksheets("Платеж").**) можно опустить.

В ходе выполнения диалога в окне **UserForm1** необходимо раскрыть и выбрать из списка «Заказчик» фирму-заказчика, а из списка «Товар» – наименование товара. В поле **Количество** заказчик определяет нужное ему количество товара.

Программа для командной кнопки «Принять заказ» на форме пользователя должна заполнить ячейки на рабочем листе **Платеж**, подсчитать стоимость товара и заполнить поле даты. В программе учтено то, что нумерация элементов в комбинированном списке начинается с нуля:

```
Sub CommandButton1_Click()  
n = ComboBox1.ListIndex + 1  
Worksheets("Платеж").Range("b8") = Worksheets("Заказчики").Cells(n + 1, 1)  
Worksheets("Платеж").Range("b9") = Worksheets("Заказчики").Cells(n + 1, 2)  
Worksheets("Платеж").Range("b10") = Worksheets("Заказчики").Cells(n + 1, 4)  
m = ComboBox1.ListIndex + 1  
Worksheets("Платеж").Range("a13") = Worksheets("Товары").Cells(m + 1, 1)  
Worksheets("Платеж").Range("b13") = Worksheets("Товары").Cells(m + 1, 2)  
Worksheets("Платеж").Range("c13") = TextBox1.Text  
a1 = Worksheets("Платеж").Range("b13")  
a2 = Worksheets("Платеж").Range("c13")  
Worksheets("Платеж").Range("d13") = a1 * a2  
Worksheets("Товары").Cells(m + 1, 3) = Worksheets("Товары").Cells(m + 1, 3) -  
- TextBox1.Text  
Worksheets("Платеж").Range("b17") = Date
```

```
Worksheets("Платеж").Activate
End
End Sub
```

Встроенная функция **Date** определяет текущую дату. При выполнении оператора **End** произойдет закрытие диалогового окна пользовательской формы и на экране останется лист **Платеж**.

После формирования листа **Платеж** его можно напечатать с помощью кнопки «Печать», которая может быть размещена на этом же листе вне ячеек с информацией:

```
Sub CommandButton3_Click()
Worksheets("Платеж").Range("a1:e18").PrintOut
End Sub
```

Естественно, что печать можно осуществлять в том случае, когда подключено печатное устройство.

## 9.2. Формирование новой таблицы программным путем

В качестве простейшего примера совместной работы двух приложений Excel и Word рассмотрим задачу рассылки извещений, созданных в редакторе Word, тем заказчикам, которые пожелали приобрести товар (в задаче предыдущего параграфа), но не перевели деньги на счет фирмы-продавца.

Предварительно следует добавить изменение в программу кнопки «Принять заказ» на форме пользователя с тем, чтобы информация о сделанном заказе поступала на отдельный рабочий лист с именем **Заказы**, структура которого представлена ниже.

Заказчик	Товар	Количество	Общая стоимость	Дата	Оплата
Луч	Стул	2	160 000	11.09.08	Да
Восток	Кресло	1	120 000	11.09.08	
Восток	Стол	2	400 000	12.09.08	Да
Вест	Кресло	3	360 000	12.09.08	

Пусть шапка таблицы на листе имеется, надо программным путем сформировать содержимое. Для этого в программе для кнопки «Принять заказ» перед оператором **End** надо добавить следующий программный код:

```

i = 2
Do While Worksheets("Заказы").Cells(i, 1) <> ""
i = i + 1
Loop
Worksheets("Заказы").Cells(i, 1) = Worksheets("Платеж").Range("b8")
Worksheets("Заказы").Cells(i, 2) = Worksheets("Платеж").Range("a13")
Worksheets("Заказы").Cells(i, 3) = Worksheets("Платеж").Range("c13")
Worksheets("Заказы").Cells(i, 4) = Worksheets("Платеж").Range("d13")
Worksheets("Заказы").Cells(i, 5) = Worksheets("Платеж").Range("b17")

```

Тогда программа, помимо формирования платежной ведомости, будет заполнять первые пять столбцов списка на листе **Заказы**. В таблице **Заказы** имеется столбец «Оплата», который заполняется бухгалтером при получении сведений об оплате заказа вручную.

Затем в текстовом редакторе Word надо создать текстовый документ «Извещение» с интерфейсом, представленным на рис. 9.3.

<p>Уважаемый руководитель фирмы!</p> <p>Напоминаем Вам о необходимости произвести оплату на наш расчетный счет 200200449 за выписанный по заявке Вашей фирмы товар _____ стоимостью _____ рублей.</p> <p style="text-align: right;">Директор фирмы «Белмебель»                      Н. И. Кузнецов</p>	
--	--

Рис. 9.3. Текст документа «Извещение»

### 9.3. Использование Мастера слияния

Печать документа «Извещение» может быть осуществлена с использованием **Мастера слияния**. Для этого в окне документа «Извещение» надо выполнить команду **Сервис/Письма и рассылки/Мастер слияния**. Для версии Word 2007 – **Рассылки/Начать слияние/Пошаговый мастер слияния**. В правой части экрана появится окно **Слияние**, в котором следует поэтапно выбрать содержимое текстовых полей.

На первом шаге можно определить тип документа – письма. Затем нажать ссылку **Далее**.

На втором этапе необходимо взять за основу для создания письма текущий документ.

На третьем этапе надо нажать кнопку **Обзор**, открыть файл Excel с нужными таблицами, выбрать рабочий лист **Заказы** и отсеять те

строки, в которых имеется слово «Да» в шестом столбце, щелкнув по кнопке в названии столбца и выбрав пункт **Пустые**.

На четвертом этапе следует поставить курсор после слова «фирмы» в документе «Извещение», нажать ссылку **Другие элементы (Вставить поле слияния для Word 2007)**, в появившемся окне выбрать поле **Заказчик**, нажать кнопку **Вставить**, потом кнопку **Заккрыть**. Затем следует поместить курсор после слова «товар» в документе «Извещение» и вставить поле **Товар**. И, наконец, нужно поставить курсор после слова «стоимостью» и выбрать в таблице поле **Стоимость**.

Пятый этап служит для просмотра писем, на шестом этапе можно их печатать.

#### **9.4. Задание для выполнения на компьютере**

1. Создать описанные выше объекты и программы. Проверить и отладить работу всех программ.

2. На бланке **Платеж** сформировать в шестой строке номер заказа, увеличивая его на единицу при поступлении заказа. Внести для этого соответствующие изменения в программу кнопки «Принять заказ».

3. Сформировать документ для печати с использованием **Мастера слияния**.

## 10. БАЗЫ ДАННЫХ

**База данных (БД)** – это совокупность взаимосвязанных данных при такой минимальной избыточности, которая допускает их использование оптимальным образом в определенной предметной области.

**Система управления базами данных (СУБД)** – это программное обеспечение, предназначенное для создания, ведения и совместного использования БД многими пользователями.

Для хранения БД может использоваться как один компьютер (локальная БД), так и множество взаимосвязанных компьютеров (распределенная БД).

При создании базы данных должна быть обеспечена целостность данных. Целостность данных – это набор правил, которые предназначены для защиты информации от ее искажения при вводе, удалении, изменении или обработке.

Информация в БД представляется в виде одинаковых по своей структуре **записей**. Запись включает в себя группу взаимосвязанных **полей** данных.

Различают иерархические, сетевые и реляционные модели данных. В *иерархической базе* данные имеют древовидную структуру, где каждый элемент может быть связан с одним или несколькими элементами на более низком уровне. В *сетевой структуре* любой элемент может быть связан с любым другим элементом. В основе *реляционной модели* лежит понятие отношения, которое представляется в виде двумерной таблицы (т. е. это набор взаимосвязанных таблиц).

### 10.1. Access – реляционная база данных

Приложение Microsoft Access входит в комплект Microsoft Office и представляет собой мощное средство для работы с базами данных.

С помощью Access можно выбрать из базы разнообразную информацию для просмотра и анализа, представить данные на экране в виде таблиц или специально сконструированных форм, вывести на печать отчеты, выполнить различные вычисления в процессе отбора данных и подготовки отчетов.

Данные в приложении Access хранятся в таблицах.



*Таблицы (отношения)* используются для представления объектов, а также для представления связей между объектами. Они имеют уникальные имена и состоят из строк и столбцов. Строки соответствуют записям, столбцы – полям.

То поле (атрибут), которое однозначно идентифицирует объект, называется *ключом*, или *индексом*. Обычно ключи располагаются в первом столбце таблицы. При большом объеме файла они образуют отдельный индексный файл с первичными ключами для быстрого поиска данных.

На данные в реляционных БД накладываются следующие ограничения:

- не может быть одинаковых первичных ключей;
- все строки таблицы должны иметь одну и ту же структуру;
- имена столбцов должны быть различны;
- порядок следования строк несущественен.

В приложении Access используются *таблицы* для хранения данных, *запросы* – для отбора данных по определенным условиям, *формы* – для удобного ввода, просмотра и редактирования данных, *отчеты* – для отображения данных при выводе на печать. Пользователь имеет возможность не только использовать программы Access, но и создавать собственные пользовательские программы.

**Этапы проектирования БД.** В процессе проектирования базы данных необходимо сформулировать постановку задачи, разработать структуры таблиц, используемых форм, отчетов, запросов, определить содержимое полей, установить связи между таблицами, которые можно будет использовать для поиска связанной информации в разных таблицах.

Важным этапом является нормализация таблиц. *Нормализация* – это процедура, в процессе выполнения которой атрибуты данных группируются в таблицы, а таблицы, в свою очередь, – в базы данных. Главными целями нормализации являются исключение дублирования информации, ее избыточности, обеспечение возможности проведения непротиворечивых и корректных изменений данных в таблицах, упрощение и ускорение поиска информации.

Процесс нормализации состоит в приведении таблиц к нормальным формам. Всего существует пять нормальных форм, которые удовлетворяют соответствующим правилам нормализации.

Чтобы таблица соответствовала *1-й нормальной форме*, необходимо, чтобы все значения ее полей были неделимыми и невычисляемыми, при этом не должно быть полностью совпадающих строк.

Чтобы таблица соответствовала *2-й нормальной форме*, требуется, чтобы все неключевые поля полностью зависели от ключевого. Для удовлетворения этому правилу нормализации иногда необходимо разбить исходную таблицу на две или более. Такое преобразование называется *разложением*. Чтобы перейти к *3-й нормальной форме*, нужно, чтобы таблицы находились во *2-й нормальной форме* и все неключевые поля в таблицах зависели только от ключа таблицы и не зависели непосредственно друг от друга. Для выполнения этого правила также бывает необходимо выполнить разложение. В большинстве случаев оптимальная структура БД достигается при выполнении этих первых трех правил нормализации.

Приведение базы данных к *4-й и 5-й нормальным формам* нужно производить лишь в специальных случаях.

Таблицы, формы, отчеты, запросы можно создавать вручную (в режиме **Конструктора**) или использовать автоматизированные средства (программы-**Мастера**).

После запуска приложения Access в правой части экрана появится окно, в котором имеются три опции, предназначенные для создания новой БД, создания нового приложения на основе имеющихся шаблонов или открытия созданной ранее БД.

Для создания новой БД нужно выбрать опцию **Новая база данных** и нажать <ОК> (или выполнить команду **Файл/Создать** или использовать кнопку **Создать базу данных** на панели инструментов). Откроется диалоговое окно **Файл новой базы данных**, в котором следует указать имя базы данных.

В Access 2007 лента заменила меню и панели инструментов предыдущих версий Access. Лента дает возможность находить группы связанных команд. Например, если нужно создать новую форму или отчет, команды следует искать на вкладке **Создание**. Команды на вкладках, используемых в Access 2007, расположены ближе к поверхности и работать с ними удобнее.

## 10.2. Создание таблиц и установление связей

Можно создавать таблицы с помощью **Конструктора** и **Мастера**.

Рассмотрим процесс создания таблицы с помощью **Конструктора**. В окне **Конструктора** таблиц (рис. 10.1) в поле **Имя поля** следует ввести нужное имя, а в поле **Тип данных** определить требуемый тип.

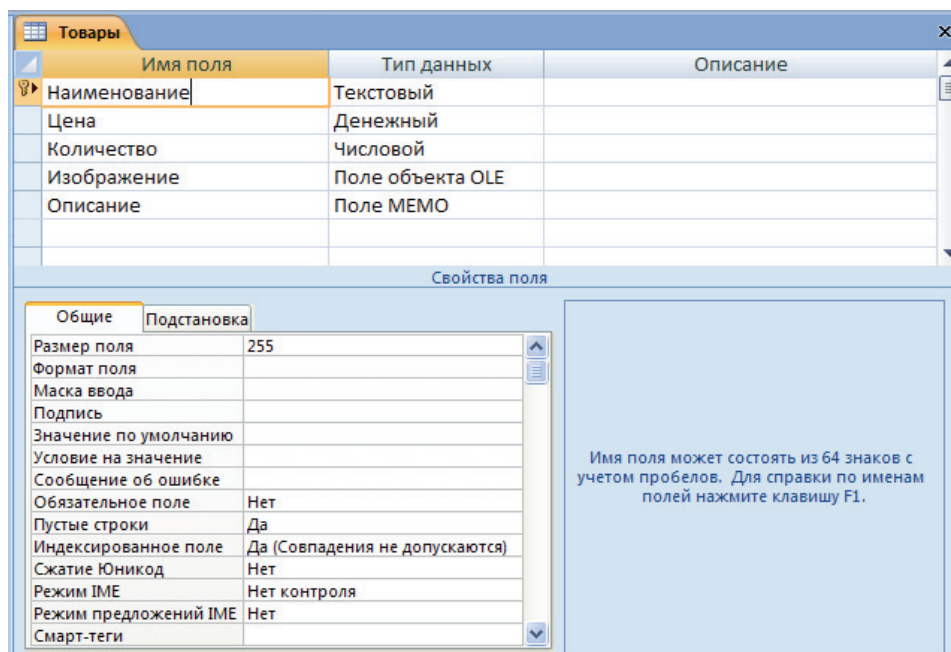


Рис. 10.1. Окно Конструктора таблиц

Имена полей могут быть заданы строчными или прописными буквами русского или латинского алфавита (не более 64 символов). При их наборе также могут использоваться цифры и пробелы. Имя поля не должно начинаться с пробела.

**Текстовый** тип (значение по умолчанию) определяет текст или числа, не требующие проведения расчетов, например номера телефонов.

**Поле MEMO** – это длинный текст или комбинация текста и чисел.

**Числовой** тип определяет числовые данные, используемые для проведения расчетов.

**Дата/время** описывает дату и время.

**Денежный** тип предназначен для описания денежных значений и числовых данных, используемых в математических расчетах, которые проводятся с точностью до 15 знаков в целой и до 4 знаков в дробной части.

**Счетчик** используется Access для создания уникальных последовательно возрастающих на единицу или случайных чисел, автоматически вводящихся при добавлении каждой новой записи в таблицу.

**Логический** тип относится к логическим значениям, а также к полям, которые могут содержать одно из двух возможных значений (Истина/Ложь, Да/Нет, Вкл/Выкл).

**Поле объекта OLE** – объект (например, электронная таблица Microsoft Excel, документ Microsoft Word, рисунок, звукозапись или другие данные в двоичном формате), связанный или внедренный в таблицу Microsoft Access.

При необходимости можно установить свойства полей на вкладке **Общие** в нижней части окна.

Одно из полей должно быть определено как ключевое. Свойства этого поля: обязательное, совпадения не допускаются. Для определения ключа необходимо в окне **Конструктора** щелкнуть мышкой слева от имени того поля, которое будет ключом, и выполнить команду **Правка/Ключевое поле** или нажать кнопку **Ключевое поле** на панели инструментов.

После определения полей следует нажать кнопку закрытия окна и на запрос о сохранении таблицы ввести ее имя.

**Установление связей.** Связь между таблицами устанавливается с помощью поля (полей), которое содержит одинаковые значения для разных таблиц. Чаще всего эта связь делается посредством первичного ключа одной таблицы с совпадающим полем (внешним ключом) другой таблицы.

Различают три вида связей:

- связь 1:1, когда каждому экземпляру одной таблицы соответствует в точности один экземпляр второй и наоборот;
- связь 1:n, когда может существовать экземпляр одной таблицы, который соответствует нескольким экземплярам другой, и обратное не допускается;
- связь m:n, когда экземпляр одной таблицы соответствует нескольким экземплярам другой таблицы и наоборот.

Для определения связей между таблицами необходимо выполнить команду **Сервис/Схема данных...** или нажать кнопку **Схема данных** на панели инструментов. При первом определении связей откроется окно **Схема данных**, после чего диалоговое окно **Добавление таблицы**, в котором следует выделить нужные таблицы и нажать кнопку **Добавить**, затем **Заккрыть**. В окно **Схема данных** будут помещены выделенные таблицы (рис. 10.2).

Сначала надо определить, какая таблица главная, а какая – подчиненная. Затем следует тянуть мышкой поле от главной таблицы к нужному полю в подчиненной таблице.

В открывшемся диалоговом окне **Изменение связи** первая строка будет заполнена полями, по которым связываются таблицы. Надо установить флажок **Обеспечение целостности данных**. Это позволит Access при вводе данных проводить дополнительный контроль. После установления флажка **Обеспечение целостности данных** становятся активными еще два флажка: **Каскадное обновление связанных полей** и **Каскадное удаление связанных записей**. Эти флажки нужно установить.

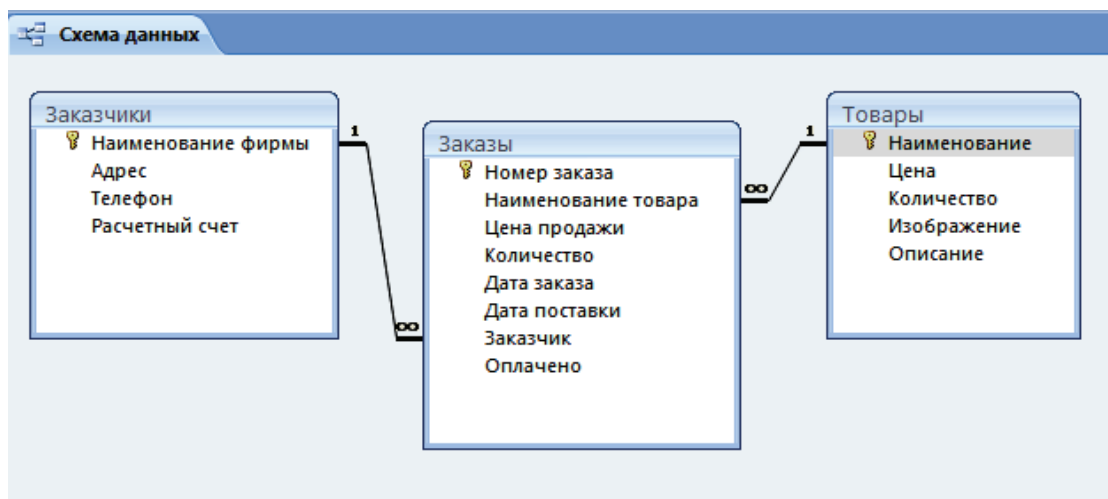


Рис. 10.2. Схема данных

После нажатия кнопки **Создать** на схеме данных будет отображена созданная связь.

Чтобы удалить связь, необходимо сделать на ней двойной щелчок и нажать **<Delete>**.

**Модификация структуры таблицы, схемы данных.** Для изменения структуры таблицы ее нужно открыть в режиме **Конструктора** (выделить таблицу и нажать кнопку **Конструктор**) и внести изменения.

С целью удаления поля его надо выделить и нажать **Del**, для вставки поля – выделить то поле, перед которым вставляется новое, и выполнить команду **Вставка/Строки**.

Удаление и переименование таблиц выполняется в окне БД из контекстного меню таблицы.

Для удаления связи между таблицами необходимо открыть схему данных, выделить связь щелчком мыши, нажать клавишу **Del**. Аналогично удаляются таблицы из схемы данных.

**Заполнение, сохранение и редактирование данных в таблице.** Порядок заполнения таблиц зависит от связей в таблицах. Вначале заполняются таблицы главные, затем – подчиненные.

Необходимо дважды щелкнуть по таблице для заполнения ее данными, после чего она будет открыта в режиме таблицы, т. е. имена полей будут записаны в качестве наименования столбцов.

Каждая строка этой таблицы представляет собой запись БД. Процесс заполнения производится по записям. При вводе данных в области выделения записи появляется маркер изменяемой записи в форме карандаша. Для перехода от одного столбца к другому используется клавиша **<Tab>**. Введенные в строку данные сохраняются при переходе к другой строке.

В режиме таблицы можно просматривать и вводить данные в несколько таблиц. Для этого между таблицами должны быть определены связи. Для раскрытия связанной записи надо щелкнуть по значку разворачивания записи (+).

После окончания ввода данных таблица закрывается кнопкой закрытия. Данные сохраняются автоматически.

Редактирование данных осуществляется обычными средствами редактирования.

### 10.3. Задание для выполнения на компьютере

1. Создать таблицу следующего вида.

Имя таблицы	Имя поля	Тип данных	Тип поля
Товары	Наименование	Текстовый	Ключевое
	Цена	Денежный	
	Количество	Числовой	
	Изображение	Поле объекта OLE	
	Описание товара	Поле MEMO	
Заказчики	Наименование фирмы	Текстовый	Ключевое
	Адрес	Текстовый	
	Телефон	Текстовый	
	Расчетный счет	Текстовый	
Заказы	Номер заказа	Числовой	Ключевое
	Наименование товара	Текстовый	
	Цена продажи	Денежный	
	Количество	Числовой	
	Дата заказа	Дата/время	
	Дата поставки	Дата/время	
	Заказчик	Текстовый	
	Оплачено	Текстовый	

2. Определить свойства полей **Цена** и **Цена продажи** в таблицах «Товары» и «Заказы», установив число десятичных знаков – 0. Ключевые поля сделать обязательными, совпадения не допускаются.

3. Создать связи между таблицами, используя поля **Наименование**, **Наименование товара**, **Наименование фирмы**, **Заказчик**.

4. Наполнить таблицы информацией.

5. Сохранить файл под именем «Продажи».

## 11. ФОРМИРОВАНИЕ ЗАПРОСОВ В ACCESS

Быстрый и эффективный доступ к данным обеспечивают запросы. Access позволяет выполнять следующие виды запросов:

- запрос на выборку данных;
- условный запрос;
- запрос с вычисляемыми полями;
- итоговый запрос;
- параметрический запрос;
- запрос на обновление записей;
- запрос на удаление записей;
- запрос на добавление записей;
- перекрестный запрос.

Для работы с запросами нужно щелкнуть по кнопке (вкладке) **Запрос**.

### 11.1. Условный запрос

Для создания условного запроса следует:

1. Нажать кнопку **Создать**, выбрать способ создания (**Конструктор**) или дважды щелкнуть по кнопке **Создание запроса в режиме конструктора**.

2. В окне **Добавление таблицы** выделить таблицу (таблицы), из которой будут выбираться данные. Нажать кнопку **Добавить**, затем кнопку **Заккрыть**.

3. Заполнить бланк запроса нужными полями перетягиванием имени поля из таблицы в верхней части окна в строку **Поле** бланка запроса (или двойным щелчком по полю).

4. В строке **Условие отбора** того поля, на которое налагается условие, ввести условие отбора. При этом используются операторы:

- логические операторы: **OR**, **AND**, **NOT**;
- операторы сравнения: **<**, **<=**, **>**, **>=**, **=**, **<>**;
- оператор **IN**, который проверяет значение поля на равенство любому значению из списка (операндом является список, заключенный в круглые скобки);

– оператор **BETWEEN**, который проверяет значение на принадлежность диапазону (**BETWEEN 15 AND 30** – аналогично условию  $\geq 15$  and  $\leq 30$ );

– оператор **LIKE**, который устанавливает соответствие текстового поля заданному шаблону символов (**Like Ф\*** – текст, начинающийся на букву Ф).

Если условие налагается на несколько полей и они связаны логическим оператором **И**, то условия вводятся в одной строке под нужными полями, если логическим оператором **ИЛИ** – то в разных строках под нужными полями.

5. Для выполнения запроса нажать кнопку **Запуск** на панели инструментов. Можно просмотреть полученные результаты.

6. Нажать кнопку закрытия окна, на вопрос о сохранении ответить **Да** и ввести имя запроса. Для *удаления таблицы* из бланка запроса нужно выделить ее щелчком мыши и нажать **Del**. Для *удаления поля* из бланка запроса следует выделить его по области выделения (в верхней части поля) и нажать **Del**.

На рис. 11.1 сформирован запрос, в котором определяются все заказчики и товары из задачи «Продажи»; поставки осуществлены в ноябре 2009 г.

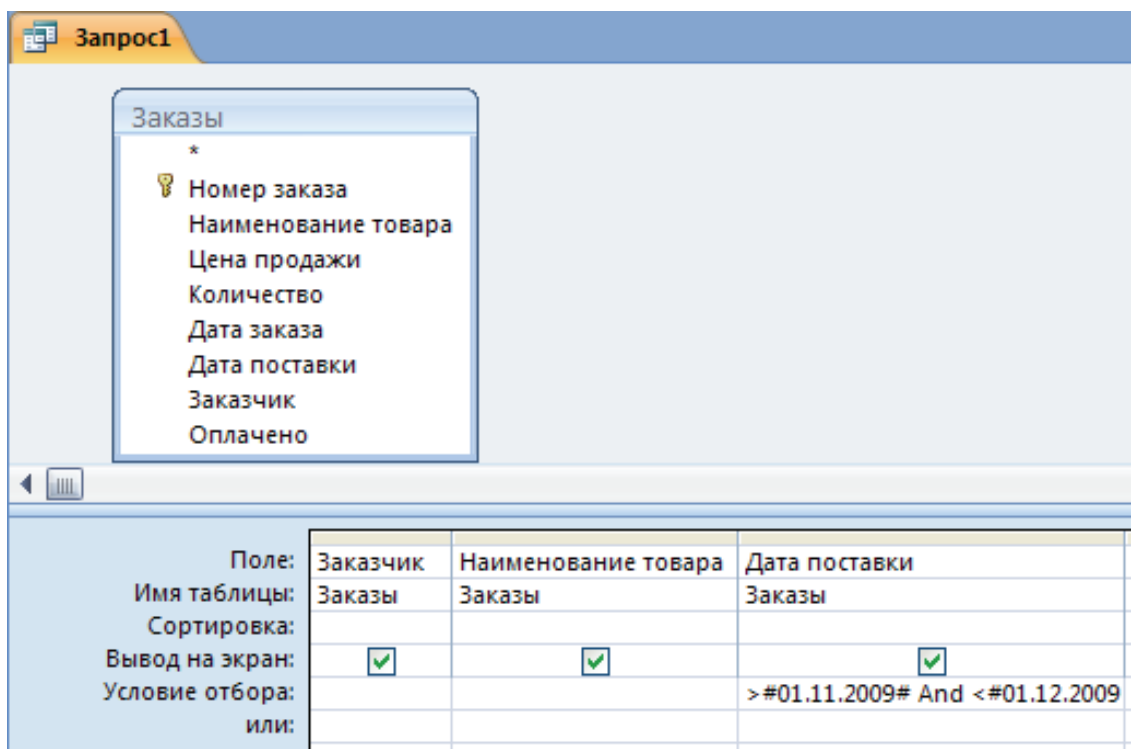


Рис. 11.1. Запрос на выборку с условием



## 11.2. Запросы с вычисляемыми полями

При построении запросов можно создавать вычисляемые поля, значения которых будут результатом выполнения операций над другими полями таблицы. Для этого нужно в пустую ячейку строки **Поле** бланка запроса ввести выражение. В выражениях можно использовать знаки арифметических операций и знак операции над текстом (знак объединения «&»). Во избежание ошибок для построения выражений лучше использовать **Построитель выражений**.

Например, пусть необходимо создать запрос «Разница в цене» для задачи «Продажи».

1. Сначала нужно указать поля, значения которых будут выводиться в запросе (в нашем запросе это поля **Наименование товара**, **Цена** из таблицы «Товары» и поле **Цена продажи** из таблицы «Заказы»).

2. Для вычисления разницы в цене необходимо установить курсор в пустом столбце строки **Поле** и нажать кнопку **Построить** на панели инструментов.

Откроется диалоговое окно **Построитель выражений**. В верхней части окна расположена область ввода выражения (рис. 11.2). В нижней части окна находятся три списка для поиска имен полей и встроенных функций, необходимых для создания выражения.

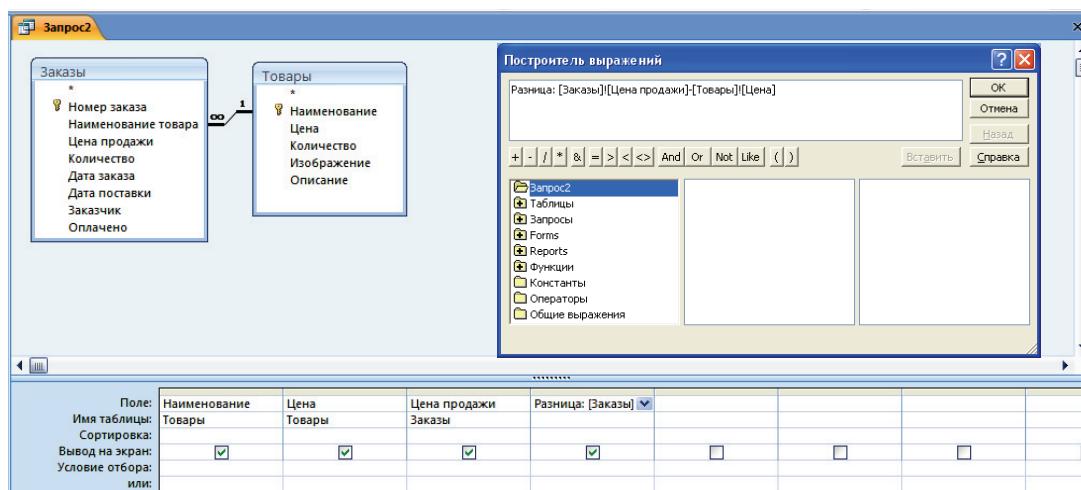


Рис. 11.2. Формирование запроса с **Построителем выражений**

Для создания выражения следует:

– в левом нижнем окне дважды щелкнуть по слову **Таблицы** (откроются имена таблиц);

– щелкнуть по таблице «Заказы» и в среднем окне дважды щелкнуть по полю **Цена продажи** (при этом ссылка на поле **Цена продажи** будет введена в верхнем окне), щелкнуть по кнопке «–», затем щелкнуть в левом окошке по таблице «Товары», в среднем окне дважды щелкнуть по полю **Цена** и нажать <ОК>. В поле будет введено построенное выражение вида:

**Выражение1: [Заказы]![Цена продажи]-[Товары]![Цена]**

У вычисляемого поля вместо слова **Выражение1** можно ввести смысловое название столбца, например **Разница**.

3. Запустить запрос на выполнение.

### 11.3. Параметрический запрос

В условиях отбора на бланке запроса обычно вводятся конкретные значения (константы). Но иногда условия отбора необходимо изменять при каждом запросе. Тогда используется параметр запроса, который делает поле переменной величиной. При каждом выполнении запроса значение параметра будет запрашиваться.

Для определения параметра в строке **Условие отбора** на поле вместо конкретного значения вводится смысловая фраза (имя параметра), заключенная в квадратные скобки ([ ]), которую Access будет рассматривать как переменную и выводить в диалоговом окне для запроса конкретного значения. Можно в одном запросе использовать несколько параметров, при этом каждый параметр должен иметь уникальное имя. Всем параметрам по умолчанию присвоен текстовый тип данных.

Например, пусть необходимо просматривать информацию о продаже конкретного товара. В бланке запроса надо определить нужные поля из таблицы «Заказы»: **Наименование товара**, **Цена продажи**, **Количество**, **Заказчик**. Для поля **Наименование товара** в строке

**Условие отбора** нужно записать текст [Введите товар].

Тогда при каждом выполнении запроса будет появляться окно, в котором нужно будет вводить наименование товара (рис. 11.3).

Можно строить более сложный параметрический запрос.

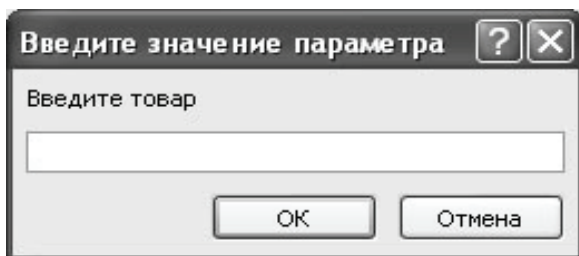


Рис. 11.3. Окно параметрического запроса

Если нужно задать временной интервал выборки информации по дате, например, после 1 сентября, но до 1 декабря текущего года, то необходимо в строке **Условие отбора** записать:

>[Введите дату начала периода] and <[Введите дату конца периода]

Вначале будет задан первый вопрос для ввода даты начала периода, а затем второй – для ввода даты конца периода.

## 11.4. Итоговые запросы

Для того чтобы подводить итоговые значения по группам данных, используются итоговые запросы (рис. 11.4). Например, пусть необходимо получить итоговые суммы по продажам отдельных групп товаров.

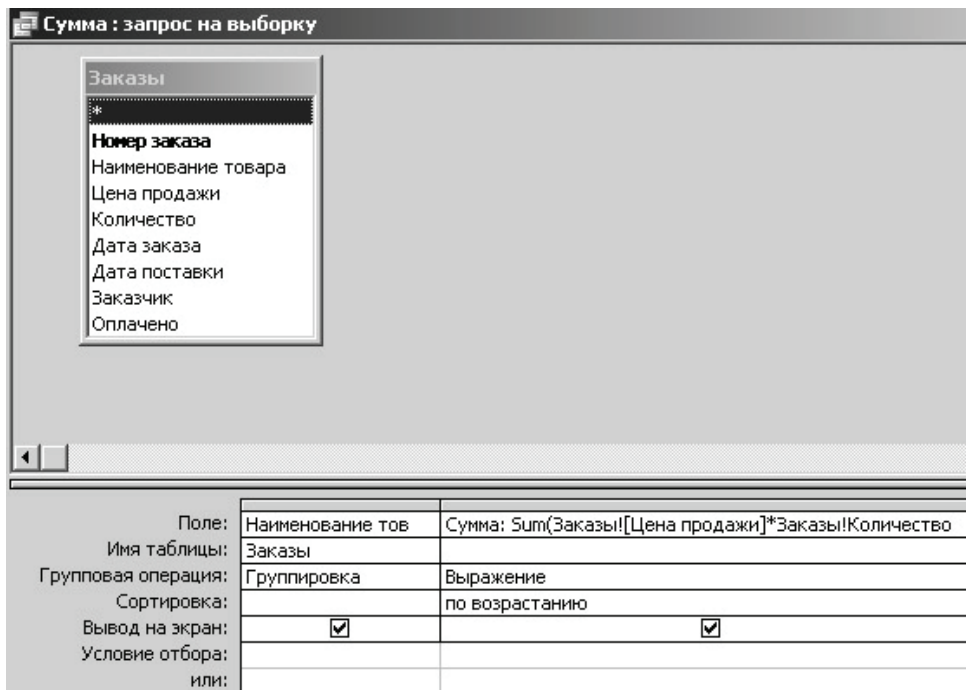


Рис. 11.4. Формирование итогового запроса

Для построения запроса нужно по таблице «Заказы» сконструировать бланк запроса следующим образом:

1. В первом столбце установить поле **Наименование товара**, во втором столбце построить выражение:

**Сумма: Sum[Заказы][Цена продажи]\*[Заказы][Количество]**

2. Нажать кнопку **Групповые операции** ( $\Sigma$ ) на панели инструментов конструктора запроса, при этом в бланке запроса появится

строка **Групповая операция** и в этой строке будет выведена установка **Группировка** для каждого поля, внесенного в бланк. Это значение нужно оставить для поля, которое объединяет записи в группы (для нашего запроса – поле **Наименование товара**). Во втором столбце надо выбрать групповую операцию.

Для выполнения групповых операций имеется несколько функций, которые выбираются в строке **Групповая операция** бланка запроса из раскрывающегося списка:

- **Sum** – вычисляет сумму значений поля в каждой группе;
- **Avg** – рассчитывает среднее арифметическое всех значений в каждой группе;
- **Min, Max** – вычисляет минимальное (максимальное) значение в каждой группе;
- **Count** – определяет количество записей в каждой группе.

Для нашего запроса надо во втором столбце установить функцию **Sum**.

3. Запустить запрос.

### 11.5. Запросы с сортировкой и запросы на обновление

**Сортировка.** В предыдущем примере на бланке запроса можно определить условия сортировки (по возрастанию или по убыванию) для поля в строке **Наименование товара**. Тогда результаты выполнения запроса будут отсортированы по наименованию товара.

**Запрос на обновление.** Для создания запроса на обновление таблицы надо на бланке запроса установить нужные поля. Затем выполнить команду **Обновление** из пункта меню **Запрос**. На бланке запроса в поле **Обновление** следует вызвать контекстное меню и выбрать команду **Построить**. В появившемся окне сформировать нужное выражение.

Например, пусть необходимо в таблице «Заказы» изменить цену всех продаваемых столов, увеличив ее на 10 000 руб. В бланке запроса нужно указать поле **Цена продажи**, выполнить команду **Обновление** из пункта меню **Запрос**. В строке **Обновление** с помощью контекстного меню следует вызвать **Построитель выражений**. Далее необходимо сделать двойной щелчок по пункту **Функции**, затем в среднем окне выбрать **Встроенные функции** и **Управление**. После двойного щелчка по **Иф** в верхнем окне надо, выбирая поля из соответствующих таблиц, построить выражение:

**IIf(Заказы![Наименование товара]="Стол";Заказы![Цена продажи]+10 000; Заказы![Цена продажи])**

Чтобы обновить базу **Заказы**, надо дважды щелкнуть на имени созданного запроса и ответить на появившиеся вопросы.

Следует отметить, что при выполнении запроса Access выбирает нужные данные из соответствующей таблицы или собирает данные из разных таблиц, производит при необходимости их обработку и представляет результат в виде новой таблицы. При этом результат запроса *не сохраняется*, т. е. таблица запроса существует только как результат выполнения запроса, а в самом запросе сохраняется только инструкция по выборке и обработке данных. Таким образом, запрос представляет собой некую *программу*, написанную на так называемом языке запросов SQL (Structured Query Language – язык структурированных запросов).

### **11.6. Задание для выполнения на компьютере**

Для задачи «Продажи» сформировать следующие запросы:

- определить товары, поставка которых предусмотрена в третьем квартале, отсортировать их по датам поставки;
- найти клиентов, заказавших товар, начинающийся на букву «С»;
- определить товары, цена которых находится в пределах от 30 000 до 100 000 руб.;
- установить заказчиков, которые не оплатили заказ;
- рассчитать прибыль, полученную в результате продажи кресел во втором квартале;
- определить заказчиков, заказавших шкаф;
- найти заказы определенной фирмы по ее названию;
- определить поставки, стоимость которых больше заданной величины;
- вычислить общую стоимость поставленных в первом квартале товаров;
- определить суммарную стоимость поставленных за год товаров отдельно по названиям.

## 12. СОЗДАНИЕ ФОРМ И ОТЧЕТОВ

### 12.1. Формы

Формы являются основным средством организации интерфейса в приложении Microsoft Access. Формы создаются на основе таблиц или запросов для ввода, вывода и редактирования данных, для создания диалоговых окон, для работы с другими объектами баз данных.

Для работы с формами нужно перейти на страничку **Формы** нажатием кнопки **Формы**. Формы создаются с помощью **Конструктора** или **Мастера**.

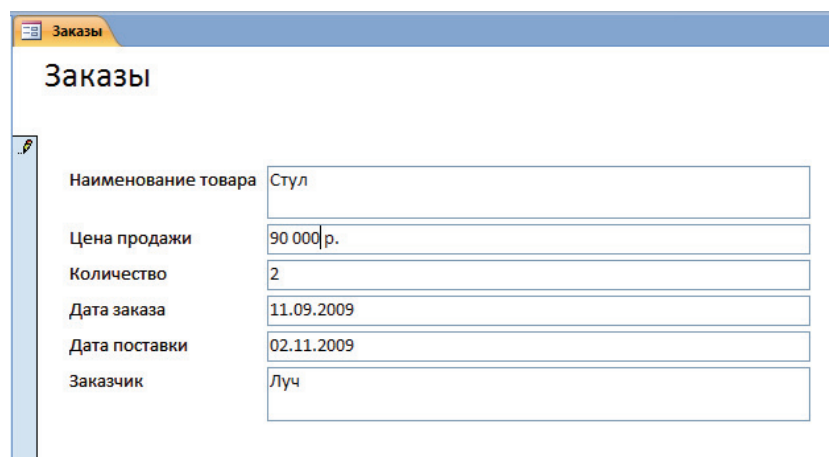
*Создание формы для просмотра и ввода данных с помощью Мастера.* Для создания формы нужно выполнить следующие действия:

1. Выбрать режим **Создание формы с помощью Мастера**. В раскрывающемся списке **Таблица/Запрос** задать имя таблицы (запроса), которая будет служить источником данных для формы. В списке **Доступные поля** выделить поля, которые будут включены в форму, и перенести их в список **Выбранные поля** кнопкой со стрелкой вправо. Нажать кнопку **Далее**.

2. Указать формат создаваемой формы (внешний вид), нажать кнопку **Далее**.

3. Выбрать стиль оформления создаваемой формы. Нажать кнопку **Далее**.

4. Ввести заголовок формы, форма сохранится под этим именем. Нажать кнопку **Готово**. Форма автоматически откроется (рис. 12.1).



Заказы	
Наименование товара	Стул
Цена продажи	90 000 р.
Количество	2
Дата заказа	11.09.2009
Дата поставки	02.11.2009
Заказчик	Луч

Рис. 12.1. Форма для ввода информации в таблицу «Заказы»

Для внесения изменений в форму ее нужно открыть в режиме **Конструктора**.

**Создание формы в режиме Конструктора.** Если форма включает поля из нескольких таблиц, то *предварительно нужно построить запрос*, включающий эти поля.

Для создания формы в режиме **Конструктора** следует:

1. Нажать кнопку **Создать**, в появившемся окне выбрать способ создания **Конструктор**, задать таблицу (запрос), которая будет служить источником данных. Нажать **<ОК>**.

2. В появившемся окне конструктора с пустой областью данных будет выведена **Панель элементов** (панель может также открываться по команде **Вид/Панели инструментов/Панель элементов** или после щелчка по кнопке **Панель элементов** на панели инструментов **Конструктор форм**) и список полей источника данных (список может также открываться по команде **Вид/Список полей** или после щелчка по кнопке **Список полей** на панели инструментов **Конструктор форм**).

3. В области данных формы разместить требуемые поля перетягиванием их мышкой из **Списка полей**. Размещаемый элемент состоит из надписи и поля данных.

Элементы управления можно перемещать по форме, изменять их размеры. Поле и присоединенная надпись передвигаются одновременно перетаскиванием мыши. Для независимого перемещения относительно друг друга поле и надпись передвигаются за левый верхний угол.

Для выравнивания выделенных элементов относительно друг друга по горизонтали и вертикали служит команда **Формат/Выровнять**. При этом выделение нескольких объектов осуществляется с удерживанием клавиши **<Shift>**.

Интервалы между элементами управления можно изменить с помощью команды **Формат/Интервал по горизонтали (Интервал по вертикали)**.

Изменение цвета фона, шрифта и размера шрифта выполняется с помощью кнопок на панели инструментов **Форматирование**.

4. Чтобы задать заголовок форме, надо выбрать команду **Вид/Заголовки/Примечание формы**. Перед областью данных будет выведена область для заголовка формы. На панели элементов нужно выбрать элемент **Надпись**, разместить его в области заголовка формы и ввести текст заголовка (другим способом можно ввести текст, вызвав свойства элемента **Надпись**, где на вкладке **Макет** записать текст в свойстве **Подпись**).

5. Перейти в режим **Формы** командой **Вид/Режим формы**.
6. Закрыть форму. На запрос о сохранении ответить **Да** и задать нужное имя формы.

**Создание формы с подчиненной формой в режиме Конструктора.** Подчиненная форма – это форма, которая внедрена в другую форму (основную). Такие формы базируются на таблицах, связанных отношением «один-ко-многим». Допускается несколько уровней подчиненности.

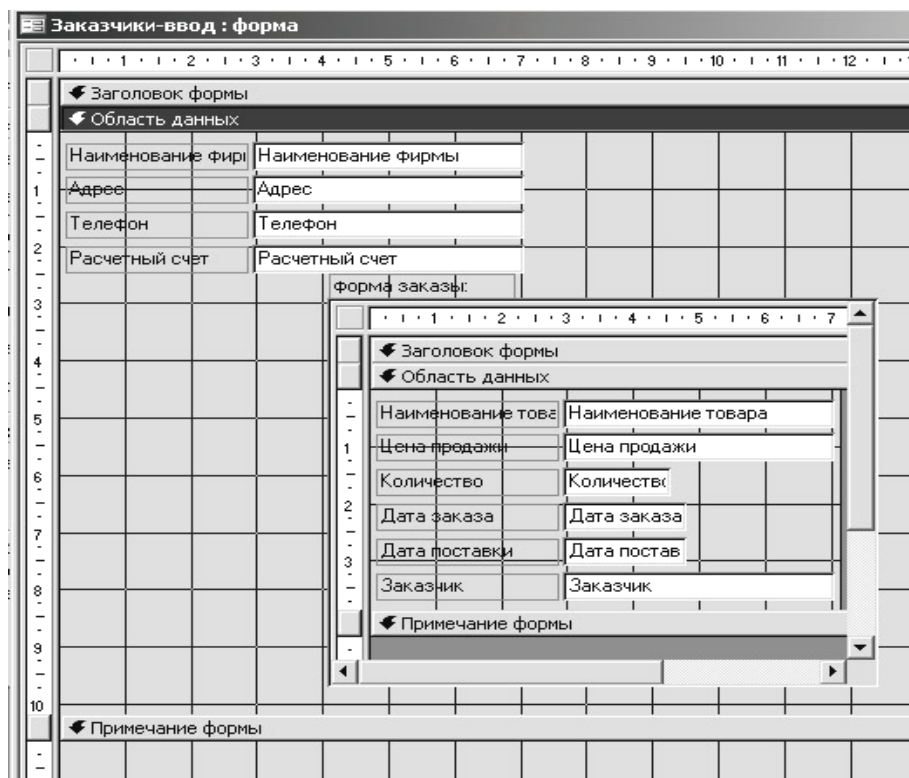


Рис. 12.2. Основная форма с подчиненной

Для создания формы с подчиненной формой (рис. 12.2) нужно:

1. Создать подчиненную форму (в режиме **Конструктора**).
2. Создать основную форму (в режиме **Конструктора**).
3. Добавить в основную форму в область данных элемент управления **Подчиненная форма/Отчет**. Этот элемент должен иметь размеры, в которые будет помещаться вся подчиненная форма.
4. Вызвать свойства внедренного элемента. Перейти на вкладку **Данные**. В свойстве **Объект-источник** указать имя подчиненной формы. Закрыть окно свойств.
5. Вместо надписи **Внедренный** ввести название подчиненной формы.



6. Перейти в режим **Формы**, выбрав команду **Вид/Режим формы**.


7. Закрывать форму. На вопрос о сохранении ответить **Да**.

**Создание кнопочной формы.** Кнопочная форма содержит кнопки для открытия объектов БД (запросов, форм, отчетов) и является основной формой для пользователя при работе с ними.

Для создания кнопочной формы с кнопкой на открытие некоторой формы нужно выполнить следующие действия:

1. Нажать кнопку **Создать** и выбрать способ создания **Конструктор**. Нажать **<ОК>** (или выбрать **Создание формы в режиме конструктора**).

2. Откроется окно **Конструктора** с пустой областью данных и будет выведена **Панель элементов** (если панель не выведена, то ее можно открыть щелчком по кнопке **Панель элементов** на панели инструментов **Конструктор форм**).

3. На **Панели элементов** кнопка **Мастера**  должна быть *активна (вдавлена)*. На **Панели элементов** выбрать элемент **Кнопка** и расположить его на форме. В окне **Создание кнопок** указать категорию действий для кнопки (**Работа с формой**) и задать действие (**Открыть форму**). Нажать кнопку **Далее**.

4. Выбрать имя открываемой формы. Нажать кнопку **Далее**.

5. Установить переключатель на **Открыть форму и показать все записи**. Нажать кнопку **Далее**.

6. Расположить надпись на кнопке или выбрать для нее рисунок. Нажать кнопку **Далее**.

7. Задать имя кнопке или оставить имя, заданное по умолчанию. Нажать кнопку **Готово**.

Для создания кнопок на закрытие формы в п. 3 следует выбрать категорию **Работа с формой** и действие **Закрывать форму**.

С целью создания кнопки на выход из приложения необходимо выбрать категорию **Приложение** и действие **Выйти из приложения**.

Для создания кнопок для работы с отчетами в п. 3 нужно выбрать категорию **Работа с отчетом** и действие **Просмотр отчета**.

## 12.2. Разработка отчетов

Отчет является средством для представления информации из БД в виде печатного документа. Для работы с отчетами нужно перейти на страничку **Отчеты** нажатием кнопки **Отчеты**.

**Создание отчета с помощью Мастера.** Для создания отчета необходимо:

1. Нажать кнопку **Создать** и задать способ создания **Мастер отчетов**, нажать **<ОК>** (или **Создание отчета с помощью мастера**).

2. В открывшемся окне в поле **Таблицы/Запросы** выбрать нужную таблицу или запрос, на основе которого будет построен отчет. В поле **Выбранные поля** перенести те поля, которые нужно отобразить в отчете (используя кнопку со стрелкой или с двумя стрелками).

3. Если в отчет необходимо включить поля из нескольких таблиц, повторить п. 2. Нажать кнопку **Далее**.

4. Если включены поля из нескольких таблиц, выбрать тип представления данных в отчете. Нажать кнопку **Далее**.

5. Добавить (если нужно) дополнительные поля для группировки записей. Нажать кнопку **Далее**.

6. Задать условия сортировки строк в области данных отчета. Если необходимо, то, нажав кнопку **Итоги**, можно применить групповые операции для числовых данных, установив флажки для выбранных функций. Нажать **<ОК>**, а затем **Далее**.

7. Выбрать макет и ориентацию страницы отчета. Нажать кнопку **Далее**.

8. Выбрать стиль отчета. Нажать кнопку **Далее**.

9. Задать имя отчета (которое также будет и заголовком отчета), установить переключатель **Просмотреть отчет** и нажать кнопку **Готово**.

**Создание отчета с помощью Конструктора.** Если отчет включает поля из нескольких таблиц, то предварительно нужно построить запрос, включающий эти поля.

Чтобы создать отчет, следует:

1. Нажать кнопку **Создать** и задать способ создания **Конструктор**, установить источник данных для отчета, нажать **<ОК>**.

2. Откроется окно **Конструктора** с пустой областью верхнего и нижнего колонтитула, областью данных, будет выведена **Панель элементов** (или можно ее открыть по команде **Вид/Панель элементов**) и список полей источника данных (список полей можно открыть по команде **Вид/Список полей** или щелкнув по кнопке **Список полей** на панели инструментов).

3. Выполнить команду **Вид/Заголовок/Примечание отчета** для отображения области заголовка и примечания. Ввести заголовок отчета с помощью элемента управления **Надпись**.

4. В области данных разместить поля перетягиванием из **Списка полей**, располагая их в строку. При этом поле нужно оставить в области

данных, а подпись поля переместить в область **Верхнего колонтитула** (шапка отчета), для чего надо выделить подпись, вырезать и вставить в область верхнего колонтитула, выровнять над полями или создать названия полей с помощью элемента управления **Надпись**.

5. В области нижнего колонтитула можно вставить номер страницы, дату (команда **Вставка/Номера страниц** или **Дата и время**).

6. В области **Примечания отчета** можно разместить итоги для всего отчета (разместить элемент управления **Поле**, вызвать его **Свойства**, на вкладке **Данные** в поле **Данные** выбрать **Построитель выражений...**, построить выражение, закрыть окно **Свойств**, задать для выражения **Надпись**).

7. Перейти в режим предварительного просмотра (**Вид/Предварительный просмотр**).

8. Закрыть отчет, на запрос о сохранении ответить **Да**.

**Отчеты с промежуточными полями.** Иногда в отчете нужно вычислять промежуточные итоги. Тогда последовательность действий лишь немного отличается от описанной выше:

1. Нажать кнопку **Создать** и задать способ создания **Конструктор**. Установить источник данных для отчета, нажать **<ОК>**.

2. Откроется окно **Конструктора** с пустой областью верхнего и нижнего колонтитула, областью данных, будет выведена **Панель элементов** и список полей источника данных.

3. Выполнить команду **Вид/Заголовок/Примечание отчета** для отображения области заголовка и примечания. Ввести заголовок отчета с помощью элемента управления **Надпись**.

4. Вызвать команду **Вид/Сортировка и группировка**. В открывшемся окне в столбце **Поле/Выражение** указать поле, по которому будет произведена группировка записей (в нашем примере – **Фамилия**). В свойствах группы в полях **Заголовок группы** и **Примечание группы** установить значения **Да**. Закрыть окно.

5. В области данных разместить поля перетягиванием из **Списка полей**, располагая их в строку. При этом поле нужно оставить в области данных, а подпись поля переместить в область **Верхнего колонтитула**. В области заголовка группы разместить поле, которое группирует данные в группы.

6. В области нижнего колонтитула можно вставить номер страницы, дату (команда **Вставка/Номера страниц** или **Дата и время**).

7. В области **Примечания отчета** можно разместить итоги для всего отчета (разместить элемент управления **Поле**, вызвать его **Свойства**, на вкладке **Данные** в поле **Данные** выбрать **Построитель**

**выражений...**, построить выражение, закрыть окно **Свойств**, задать для выражения **Надпись**). В области **Примечания группы** можно создать поле, содержащее выражение.

8. Перейти в режим предварительного просмотра (**Вид/Предварительный просмотр**).

9. Закрывать отчет, на запрос о сохранении ответить **Да**.

Для распечатки отчета нужно выполнить команду **Файл/Печать**.

### 12.3. Задание для выполнения на компьютере

1. Для задачи «Продажи» создать форму с помощью **Мастера** для ввода данных о товарах.

2. Создать форму в режиме **Конструктора** для ввода информации о заказчиках и подчиненную ей форму для ввода данных о заказанных товарах.

3. Создать отчет на основании таблицы «Заказы» с группировкой данных по заказанным товарам.

4. Создать отчет по заказчикам, не оплатившим заказы.

5. Разработать кнопочную форму на открытие страницы «Формы».

6. Создать кнопочную форму на открытие страниц «Формы» и «Отчеты» и на выход из приложения. На страницах «Формы» и «Отчеты» разместить кнопки для открытия форм и для перехода на основную страницу кнопочной формы.

## 13. ПРИЛОЖЕНИЕ POWERPOINT

**Microsoft PowerPoint** – программа подготовки и проведения презентаций. Она предназначена для создания презентационных материалов в виде слайдов и их вывода на бумагу или экран.

**Презентация** – это совокупность слайдов по определенной тематике, обычно оформленных в едином стиле. Слайд представляет собой объект, который может включать заголовки, текст, таблицы, графические объекты, схемы организации, звуковые фрагменты, видеоклипы и гиперссылки. Каждый слайд сопровождается страницей заметок, на которую можно заносить поясняющий текст как во время создания, так и при его демонстрации.

**Демонстрация презентации** представляет собой процесс показа слайдов в некотором порядке, при этом смена слайдов осуществляется вручную или автоматически.

### 13.1. Создание презентации

Процесс создания презентации состоит из таких действий, как выбор общего оформления, добавление новых слайдов и их содержимого, выбор разметки слайдов, изменение при необходимости оформления слайдов, изменение цветовой схемы, применение различных шаблонов оформления и создание эффектов анимации при демонстрации слайдов.

Презентация может создаваться с помощью **Мастера автосодержания**, с помощью шаблонов и вручную (режим **Пустая презентация**).

**Мастер автосодержания** предлагает несколько образцов презентаций на различные темы. По указанным пользователем параметрам строится предварительная схема презентации с размеченными слайдами. Разметка слайда представляет собой схему размещения меток-заполнителей (заголовка, текста, таблицы, графического объекта). Пользователю необходимо изменить содержимое этих слайдов в соответствии со своими потребностями.

**Шаблон презентации** дает возможность выбрать вид оформления слайда и содержит набор цветовых схем слайдов. Кроме стандартных шаблонов PowerPoint, можно использовать самостоятельно созданные шаблоны.

**Пустая презентация** формирует презентацию «с нуля» и предполагает работу подготовленного пользователя.

**Режимы просмотра.** Существуют различные режимы просмотра создаваемой презентации. Переход от одного режима в другой осуществляется с помощью команд меню **Вид** или кнопок в нижней левой части окна презентации:

– обычный режим (команда **Вид/Сортировщик слайдов**) – это послайдный просмотр и редактирование. С одного слайда на другой можно переключиться при помощи вертикальной полосы прокрутки, а также клавиш **<PageUp>** и **<PageDown>**;

– режим сортировщика слайдов (команда **Вид/Сортировщик слайдов**) служит для просмотра всей последовательности слайдов и изменения порядка их расположения в презентации. В режиме сортировщика маленькие изображения слайдов выстраиваются на экране одно за другим в том порядке, в каком их будут показывать во время демонстрации. В этом режиме можно создавать и копировать слайды по одному и группами, а также менять их последовательность. Под стандартной панелью инструментов появляется панель **Сортировщик слайдов**, в которой кнопка **Смена** позволяет выбрать способ перехода от одного слайда к другому;

– режим страницы заметок (команда **Вид/Страницы заметок**) используется для заполнения или просмотра страниц заметок для каждого слайда. Здесь страница делится надвое. В верхней части изображен слайд, а в нижней – поле для заметок докладчика. Заметки вносятся при создании презентации. Затем их можно вывести на печать;

– режим показа слайдов (команда **Вид/Показ слайдов**) предназначен для демонстрации презентации. Обычно показ полноэкранный, а слайды меняются либо автоматически, либо по щелчку мыши. Но эти параметры можно изменить (выполнить команду **Показ слайдов/Настройка презентации** и внести нужные коррективы в появившемся окне).

**Стиль презентации.** Существует три способа управления стилем презентации: шаблоны оформления, цветовые схемы и образцы.

**Шаблон оформления** содержит образцы слайдов, цветовое оформление и заголовки с настраиваемыми форматами и стилизованными шрифтами. Можно создавать и свой шаблон презентаций.

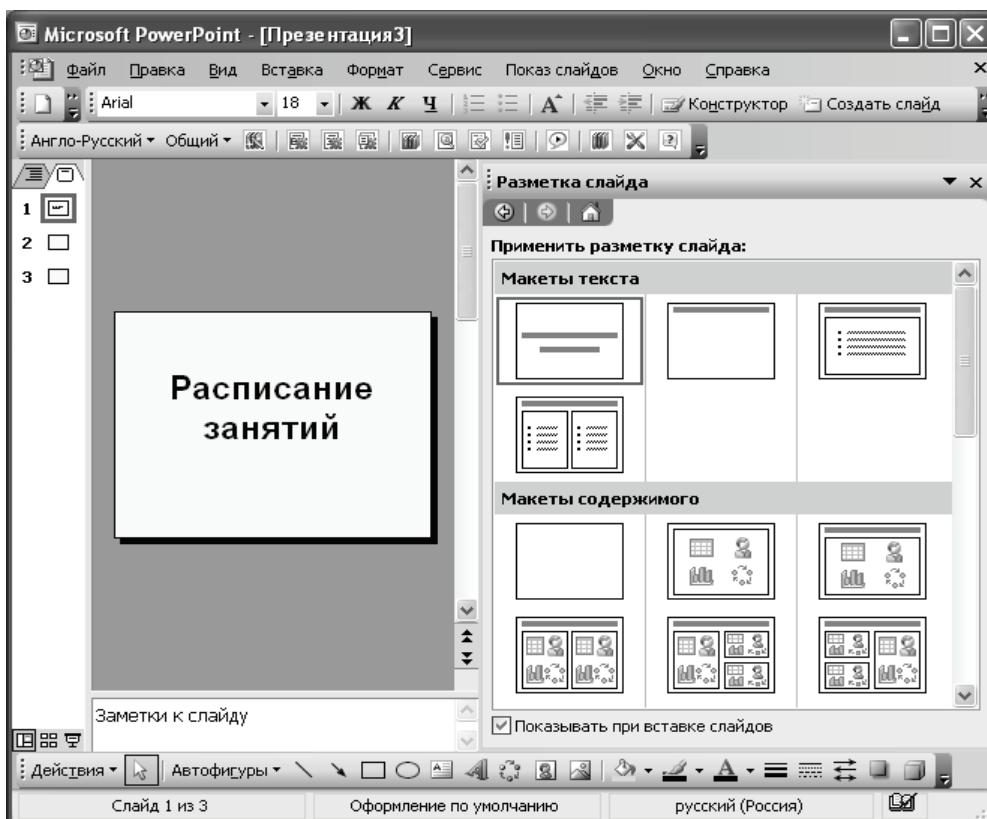
**Цветовая схема** состоит из восьми цветов, используемых в качестве основных для текста, фона, заливки, акцентов и т. п. Каждый цвет схемы используется автоматически для различных элементов слайда. При применении шаблонов оформления можно выбрать цветовую схему из набора, хранящегося в каждом шаблоне.

**Образец слайдов** задает расположение и некоторые параметры текста (шрифт, размер и цвет, цвет фона, заливка и стиль маркера списка). Он содержит пустые рамки для текста и колонтитулов, включающих дату, время и номер слайда.

Любое изменение образца отражается на каждом слайде презентации. Например, при добавлении графического объекта в образец он появится на всех слайдах презентации. С другой стороны, если созданы уникальные слайды (например, использованы цвета, отличные от цветовой схемы образца слайдов), они сохранят свое оформление, несмотря на изменения образца слайдов.

### 13.2. Создание слайдов и объектов

Для того чтобы создать новый слайд, необходимо выполнить команду **Вставка/Создать слайд** или воспользоваться соответствующей кнопкой на панели инструментов. В появившемся окне в правой части экрана надо выбрать структуру слайда (рисунок) и заполнить появившийся на экране шаблон необходимой информацией.



Окно PowerPoint

Можно создавать слайды также следующими способами:

1. Выделить уже имеющийся слайд презентации, выполнить команду **Вставка/Дублировать слайд**.

2. Команда **Вставка/Слайды из файлов** позволяет копировать слайды из одной презентации в другую. При ее использовании открывается окно, в котором, воспользовавшись кнопкой **Обзор**, можно выбрать презентацию с нужным слайдом. Для этого надо щелкнуть по кнопке **Показать**, выбрать слайды и щелкнуть по кнопке **Вставить**.

3. Команда **Вставка/Слайды из структуры** позволяет вставить любой текстовый файл, абзацы которого PowerPoint преобразует в слайды презентации.

Слайды PowerPoint состоят из объектов (текст, график, рисунок, схема организации, видеоклип, электронная таблица Excel, документ Word и др.). К выделенным объектам применимы все известные средства форматирования.

Для художественного оформления презентации применяется дизайн, цветовая схема, фон и разметка слайдов. Для настройки этих параметров используются команды **Разметка слайда**, **Фон**, **Оформление слайда** из пункта меню **Формат** или контекстного меню слайда в режиме послайдного редактирования. Дизайн распространяется на всю презентацию. Разметка, цветовая схема и фон могут изменяться от слайда к слайду.

По команде **Фон** открывается диалоговое окно, в котором определяется цвет, узор, градиентная заливка, текстура или изображение, которое послужит фоном для текущего слайда или всей презентации.

Для того чтобы изменить оформление группы слайдов, нужно перейти в режим **Сортировщика слайдов**, выделить группу и выбрать команды меню **Формат**.

Чтобы вставить текст вне разметки или фигуры, можно воспользоваться инструментом **Надпись**, расположенным на панели инструментов **Рисование**, или командой **Вставка/Надпись**.

Для вставки рисунка служит команда **Вставка/Рисунок**, для вставки других объектов – команда **Вставка/Объект/Тип объекта**.

При размещении на слайде нескольких объектов PowerPoint автоматически формирует порядок их расположения, помещая каждый на своем слое. Перемещаться от объекта к объекту можно нажатием клавиши **<Tab>**. Для изменения порядка размещения объектов служит команда **Порядок** из контекстного меню слайда.

На слайде можно разместить гиперссылку на другое место в презентации, на другой файл, на страницу сайта в Интернете. Для этого



надо выделить мышкой нужный текст, в контекстном меню выполнить команду **Гиперссылка** и в появившемся окне выбрать нужный вариант связи.

Для текста и любого объекта можно задать способ появления на экране, например вылет слева, появление текста по буквам, словам. Можно также задать поведение объектов при добавлении нового элемента – затемнение или изменение цвета. Для этих целей применяются команды **Показ слайдов/Эффекты** или **Показ слайдов/Настройка анимации**.

Управление сменой слайдов и выбором специальных эффектов при их появлении во время демонстрации осуществляется с помощью команды **Показ слайдов/Смена слайдов**. В появившемся справа окне можно определить способ перехода к следующему слайду: автоматически через заданное время или по щелчку мыши, а также эффект перехода слайда, например наплыв вниз, растворить, жалюзи и т. д.

### 13.3. Презентация

Для запуска презентации следует выбрать первый слайд и выполнить команду **Показ слайдов/Начать показ** или нажать <F5>.

Для смены порядка показа слайдов надо перейти в режим **Сортировщика слайдов** и поменять порядок размещения слайдов.

Каждый слайд презентации присутствует на экране определенное время, длительность которого можно регулировать вручную или установить предварительно, задав смену слайда через некоторый промежуток времени командой **Показ слайдов/Настройка времени**. Для получения информации о временных интервалах следует открыть презентацию в режиме **Сортировщика слайдов**, в котором под каждым слайдом указана продолжительность его присутствия на экране.

Для управления ходом показа слайдов можно использовать управляющие кнопки. Чтобы поместить кнопку на слайд, нужно выполнить команду **Показ слайдов/Управляющие кнопки** и указать требуемую кнопку (например **Домой**, **Назад**, **Далее**, **В начало** или **Возврат**). Затем в окне **Настройка действия** принять гиперссылку, предложенную в списке **Перейти по гиперссылке**, или задать другую. В качестве гиперссылки можно определить слайд в текущей презентации, переход в другую презентацию, запуск любого приложения, адрес какого-либо сайта. Выполнение выбранного действия можно задать по щелчку на управляющей кнопке или при наведении на нее указателя мыши.

Если выполнено ответвление к другой презентации, то после ее демонстрации происходит автоматический возврат к исходному слайду. При переходе к слайду внутри презентации для возвращения к исходному необходимо использовать **Навигатор слайдов** или создать соответствующую кнопку.

Чтобы приостановить демонстрацию слайдов, надо щелкнуть правой кнопкой мыши, выбрать **Экран/Пауза/Завершить показ слайдов** или нажать клавишу **<Esc>**.

### **13.4. Особенности работы с приложением PowerPoint 2007**

Приложение Office PowerPoint 2007 предоставляет пользователю новые и улучшенные эффекты, темы и усовершенствованные параметры форматирования.

Средства и команды распределены по вкладкам и связанным группам. При выборе параметров форматирования из коллекций типовых экспресс-стилей, макетов, форматов таблиц, эффектов и других средств можно экономить время и создавать презентации хорошего качества. Режим динамического просмотра позволяет просмотреть форматирование перед его применением (выбрать на ленте раздел **Анимация/Переход к этому слайду**).

Лента содержит разделы: **Главная, Вставка, Дизайн, Анимация, Показ слайдов, Рецензирование, Вид, Разработчик**.

Можно создавать презентации с широким диапазоном возможностей форматирования текста, включая перенос текста в пределах фигуры, текст в столбцах или текст, отображающийся на слайде вертикально, а также линейки на уровне параграфа. Работа с текстом осуществляется при помощи раздела **Главная/Шрифт** или **Главная/Абзац**.

Новые стили символов предоставляют дополнительный выбор для текста. В дополнение ко всем стандартным стилям из предыдущих версий приложения в PowerPoint 2007 есть такие стили, как «Все прописные» и «Малые прописные», «Зачеркивание» и «Двойное зачеркивание», «Двойное подчеркивание» и «Цветное подчеркивание». К тексту добавляются заливки, линии, тени, свечение, кернинг и объемные эффекты. Выбрать стили можно из раздела **Главная/Рисование**.

С помощью тем можно изменить вид презентации одним щелчком мыши. Выбрав другой параметр, можно поменять шрифты темы (**Дизайн/Темы**).

### **13.5. Задание для выполнения на компьютере**

1. Создать презентацию расписания занятий. Использовать различные способы оформления, появления объектов на экране и эффекты появления слайдов.

2. Опробовать демонстрацию презентации в ручном режиме и в автоматическом.

3. Добавить на слайды гиперссылки на документы Word или другие объекты и управляющие кнопки. Опробовать их работу в процессе презентации.

4. Создать презентацию выполненной в семестре работы. Данная презентация должна содержать по одному слайду на каждую выполненную лабораторную работу в текущем семестре. Обязательными элементами должны быть гиперссылки на файлы лабораторных работ, а также требования, изложенные в п. 1–3.

## 14. КОМПЬЮТЕРНАЯ ГРАФИКА. ПРИЛОЖЕНИЕ MS VISIO

### 14.1. Виды компьютерной графики

Выделяют три вида компьютерной графики: растровую, векторную и фрактальную. Они различаются принципами формирования изображений.

*Растровые изображения* состоят из массива точек различного цвета. Растровая графика отличается хорошим качеством, ее применяют в основном при разработке мультимедийных программ и полиграфических изданий. Растровыми являются все сканированные изображения и цифровые фотографии, а также изображения, созданные в некоторых графических редакторах, таких как Paint, Photoshop. Основным недостатком растровой графики является потеря качества изображения при его увеличении. Форматы растровых изображений: **bmp, jpg, gif, psd, pcx, tiff** и др.

*Векторные изображения* создаются из элементарных фигур: линий, кривых, прямоугольников и др. При изменении размеров векторного рисунка компьютер прорисовывает линии и фигуры заново. Таким образом, не происходит искажения рисунка, характерного для растровых изображений. Это свойство векторной графики широко используется при создании чертежей и в картографии. Векторные изображения создаются в приложениях Microsoft Visio, CorelDraw, Macromedia Flash и т. д.

*Фрактальная графика* основана на генерации изображения с использованием программирования. Все изображение строится по уравнению, причем мелкие элементы фрактального объекта повторяют свойства всего объекта. Многие элементы живой природы обладают фрактальными свойствами, к примеру дерево или снежинка. Фрактальная графика получила широкое распространение при создании трехмерной графики (в играх, программах для ландшафтного проектирования и т. д.).

### 14.2. Разрешение изображения и цветовые модели

Следует различать понятия: разрешение экрана, разрешение принтера и разрешение изображения.

**Разрешение экрана** – это свойство компьютерной системы и операционной системы. Оно зависит от возможностей видеокарты и монитора, от настроек операционной системы. Разрешение экрана измеряется в пикселах и определяет размер изображения, которое может поместиться на экране.

**Разрешение принтера** – это свойство принтера, указывающее на количество точек, которое будет напечатано на квадратном участке единичной длины.

**Разрешение изображения** – это свойство самого изображения. Измеряется в точках на дюйм (dpi) и задается при создании изображения. Значение разрешения изображения хранится в файле изображения. Оно непосредственно связано с физическим размером изображения, которое может измеряться как в пикселах, так и в единицах измерения длины.

**Цветовые модели.** Информация, которой манипулируют графические редакторы для воссоздания в своем рабочем пространстве изображения, хранится в цветовых каналах файла изображения. То, каким образом распределена информация по каналам, определяется цветовой моделью, используемой для данного изображения.

**Цветовая модель RGB.** Цветовая модель **RGB** строится по принципу сочетания **красного (Red)**, **зеленого (Green)** и **синего (Blue)** различных яркостей. В режиме **RGB** осуществляется сканирование изображений, эта же модель положена в основу конструкции экрана монитора. Человеческий глаз содержит рецепторы трех видов, каждый из которых реагирует на определенную длину волны, эти длины соответствуют красному, зеленому и синему цветам.

**Цветовая модель HSB.** Цветовая модель **HSB** описывает, скорее, качество световых волн, воспринимаемых глазом, а не цветовые составляющие. Цветовая модель **HSB** представлена тремя компонентами: **H** – **тон (Hue)**, **S** – **насыщенность (Saturation)**, **B** – **яркость (Brightness)**. **Hue** (цветовой тон) – частота световой волны, отражающейся от объекта. Именно этот параметр света и позволяет отличать один цвет от другого. Еще одной характеристикой является насыщенность (**Saturation**). Чем сильнее, чище воспринимается цвет, тем больше он насыщен.

**Цветовая модель LAB.** Достоинством этой модели является ее независимость от способа производства цвета, в ее системе можно описывать как цвета печати, так и цвета, излучаемые монитором. Для построения модели **LAB** также используются три компонента: **яркость (Lightness)** и **интенсивность (Chroma)**, которые вместе

составляют **информацию об освещенности (Luminance)** в изображении, содержащуюся в канале **L**. Канал **A** хранит информацию о тонах от зеленого до пурпурного, и, наконец, информация о тонах от голубого до желтого приходится на канал **B**.

**Цветовая модель CMYK.** Цвета модели **CMYK** (от слов **Cyan, Magenta, Yellow, black** – голубой, пурпурный, желтый, черный) известны как цвета многослойной печати. Именно с использованием чернил из этого набора базовых цветов сделана большая часть печатной продукции.

### 14.3. Приложение Microsoft Visio 2003

**Microsoft Visio** – программа для создания технических, архитектурных чертежей, бизнес-диаграмм, блок-схем программ, планов местности, схем тепло-, водо-, электроснабжения, работы с техническими документами и диаграммами.

Всего в **Visio** имеется 16 основных категорий изображений: **Block Diagram** – шаблоны для построения различных изображений (в этих шаблонах находятся трафареты со всеми основными графическими фигурами); **Brainstorming** – шаблоны для построения схем, описывающих новые идеи на заданную тему; **Building plan** – шаблоны для построения разнообразных планов зданий, помещений и т. д.; **Business Process** – шаблоны для создания схем бизнес-процессов; **Charts and Graphs** – шаблоны для построения диаграмм и графиков; **Database** – шаблоны для построения описания структуры базы данных; **Electrical Engineering** – шаблоны для построения электрических схем; **Flowchart** – шаблоны для построения блок-схем; **Map** – шаблоны для создания схем и планов местности; **Mechanical Engineering** – шаблоны для создания схем технических установок; **Network** – шаблоны для проектирования схем локальных или глобальных компьютерных сетей; **Organization Chart** – шаблоны для создания организационных диаграмм; **Process Engineering** – шаблоны для создания схем, связанных с *перемещением* объектов, например схем трубопроводов; **Project Schedule** – шаблоны, позволяющие создавать схемы, которые связаны с процессами, зависящими *от времени*; **Software** – шаблоны, позволяющие создавать схемы, связанные с работой программного обеспечения, например схемы распределения системных ресурсов, связи отдельных файлов проекта между собой и т. д.; **Web Diagram** – шаблоны для проектирования структуры web-узлов.

**Типы фигур.** В Visio различают фигуры одномерные (1-D), двумерные (2-D) и трехмерные (3-D). Одномерные фигуры представляют собой отрезок прямой и имеют один параметр – длину. Положение и размер двумерных фигур определяется описывающим прямоугольником, по углам и центрам сторон которого установлены восемь маркеров выделения. Фигуры трехмерные – это те же двумерные фигуры с тенью, имеющей маркер управления, с помощью которого можно менять угол наклона и размер тени.

Для изменения типа фигур существует команда меню **Формат/Поведение**. Чтобы сделать выделенную фигуру трехмерной, используется **Формат/Тень**, в диалоговом окне можно задать или изменить цвет и другие свойства тени.

**Создание фигур.** Создание фигуры может идти двумя путями: с помощью мастеров из панели трафаретов или посредством инструментов рисования.

1. Чтобы создать фигуру с помощью мастера, надо задать категорию командой **Файл/Объекты**, выбрать в ней шаблон, щелкнуть левой кнопкой мыши по мастеру (фигуре) и перетащить его на лист рисунка. Таким образом создавать рисунок, перетаскивая мышью объекты.

2. Фигуры, созданные с помощью инструментов рисования, называются пользовательскими. Основные инструменты рисования расположены на панели **Рисунок**, которая открывается при нажатии кнопки **Инструменты рисования** на панели **Стандартная**. Это **Карандаш**, **Линия**, **Дуга**, **Рисование** (разнообразные линии и способы редактирования), **Прямоугольник** и **Эллипс** (двумерные фигуры).

Инструмент **Карандаш** предназначен как для рисования линий и дуг, так и для управления маркерами. Кроме того, используя его, можно удалять вершины (установить курсор на вершине и нажать клавишу **<Delete>**) и добавлять их (установить курсор на контур и, нажав клавишу **<Ctrl>**, щелкнуть левой кнопкой мыши). При нажатой клавише **<Shift>** инструмент **Линия** рисует линии под углом, кратным 45°. Дуга между конечными маркерами, нарисованная с помощью инструмента **Дуга**, представляет собой четверть эллипса.

Инструмент **Рисование** позволяет нарисовать произвольную сглаженную кривую, состоящую из множества дуг, содержащих точки контроля. С помощью инструментов **Прямоугольник**, **Эллипс** рисуют соответственно прямоугольники и эллипсы. Чтобы изобразить квадрат или круг, надо нажать и удерживать клавишу **<Shift>**.

**Изменение геометрии фигур и их выделение.** Изменения выполняются с помощью маркеров – специальных символов, которые

появляются в области фигуры при ее выделении и при других действиях. Как правило, все действия, в том числе и инструментальные, выполняются над фигурой после того, как она выделена.

Способы выделения:

- по одиночной или сгруппированной фигуре щелкнуть левой кнопкой мыши;
- охватить рамкой выделения фигуру или группу смежных фигур;
- для выделения несмежных фигур удерживать кнопку **<Shift>** и щелкать на каждой фигуре левой кнопкой мыши.

С помощью маркеров выделения (зеленые квадратики) можно менять размеры фигуры. Маркеры конечные размещаются в начале линии (знак «×») и в конце (знак «+»), служат для изменения размера, угла наклона и радиуса кривизны нелинейной фигуры. Маркеры вращения (зеленые кружки) предназначены для поворота фигуры на произвольный угол. Эти маркеры появляются при использовании команд **Вращение** или **Текстовый блок** из меню **Формы**. Маркеры контроля (зеленые кружки) появляются при выборе инструмента **Карандаш** и расположены между вершинами фигуры. Используются для изменения формы фигуры. Маркеры вершины (зеленые ромбы) появляются в вершинах выделенной фигуры при выборе любого из инструментов: **Карандаш**, **Линия**, **Дуга**, **Рисование**.

Фигура может быть защищена от любых изменений. При этом маркеры, отвечающие за тот или иной вид действий, принимают вид замка. Для установки «замков» надо воспользоваться командой меню **Формат/Защита** и установить флажки защиты.

**Редактирование линий.** Чтобы изменить свойства рисуемой линии, в главном или выпадающем меню надо выполнить команду **Формат/Линия**. В окне диалога можно переопределить цвет линии, толщину, узор, прозрачность и другие параметры.

Инструменты редактирования цвета отдельных элементов фигуры используются часто, поэтому они вынесены на панель форматирования: **Цвет текста**, **Цвет линии**, **Цвет заливки**.

Если пользовательская фигура включает в себя несколько однотипных элементов, то для их копирования можно выбрать команду **Правка/Дублировать**.

**Текстовые элементы.** Текстовые элементы служат для создания заголовков, пояснений, обозначений, подписей в специальных полях фигур.

На панели **Стандартная** имеются кнопки **Текст**, **Текстовый блок** для вставки и редактирования текста. Текст вводится обычным



образом, и завершается работа с ним нажатием клавиши <Esc> или кнопки **Указатель** на панели **Стандартная**.

Можно вставить документ Word, для чего следует выполнить команду **Вставка/Объект** и в окне диалога выбрать тип документа – **Document MS Word**. Аналогично могут быть внедрены и другие объекты из приложений Windows, например PowerPoint, MS Excel.

Средства форматирования текста идентичны соответствующим командам в Word и составляют основное содержание панели инструментов **Форматирование**.

Все параметры форматирования текста доступны в меню **Формат/Текст**. В окне диалога пять вкладок: **Шрифт**, **Абзац**, **Блок текста** (вертикальное выравнивание, границы полей и цвет фона), **Табуляция** (параметры табуляции), **Буллеты** (примеры символов).

Создание заголовков, рамок листа и штампов облегчает трафарет **Borders and Titles**. Множество заготовок находится на трафарете **Title Blocks**; всех их можно найти, выполнив команду **Файл/Объекты/Visio Extras**.

**Ввод формул и графики.** Набор сложных формул удобно выполнять средствами редактора формул **Microsoft Equation**. Для этого нужно выбрать команду **Вставка/Объект** и запустить редактор формул.

Для ввода простых формул служит команда **Вставка/Картинки/Уравнение**. На панели **Формула** имеются шаблоны.

Работа с графиками осуществляется с использованием команды **Вставка/Картинки/График**.

#### 14.4. Операции над фигурами в MS Visio

Операции применяются к предварительно объединенным фигурам, причем для некоторых операций имеет значение порядок следования фигур. Чтобы объединить фигуры, надо выделить их, при этом для первой фигуры маркеры выделения будут закрашены в зеленый цвет, для остальных – в голубой. Затем следует выполнить команду **Форма/Операция** и выбрать одну из следующих операций:

- **Фрагмент** – разбивает на отдельные фигуры по линиям пересечения;
- **Комбинировать** – удаляет пересечение фигур и объединяет не пересекающиеся;
- **Объединение** – объединяет все фигуры в одну;

- **Вычитание** – из крайней левой фигуры вырезаются все пересечения;
- **Пересечение** – общий фрагмент из пересечения всех исходных;
- **Соединение** – то же, что и объединение, но остаются все контурные линии;
- **Отсечение** – то же, что и разбиение, но остаются только контурные линии;
- **Сдвиг** – двойная копия фигуры «наружу и вовнутрь» со сдвигом, указанным в окне диалога.

#### **14.5. Задание для выполнения на компьютере**

1. Создать план зданий БГТУ. Сначала сделать двумерный план, затем преобразовать его в трехмерный. Под изображением разместить подпись.
2. Опробовать различные операции над фигурами. Используя эти операции, разработать логотип некоторой фирмы.

## 15. СОЗДАНИЕ И ОФОРМЛЕНИЕ ДОКУМЕНТОВ MS VISIO

### 15.1. Настройка параметров листа и стиль

Документ Visio является многостраничным, причем каждый лист может иметь свои настройки: имя листа, размер, масштаб и т. п. Настройка выполняется в окне диалога **Файл/Настройка страницы**, содержащем несколько вкладок, на которых можно определить тип размещения листа (**Лицевая** – на переднем плане, **Фон** – как фон), выбрать размеры страницы и ориентацию листа (**Портрет** или **Альбом**), задать параметры и стили соединения фигур и объектов.

**Масштаб документа.** В пункте меню **Вид** имеются команды масштабирования: **Вид/Масштаб** и **Вид/Положение&Масштаб**.

При использовании **Положение&Масштаб** появляется новое окно просмотра, в котором отображается весь рисунок и рамка красного цвета вокруг видимой части. Если захватить мышкой край или угол рамки и уменьшить ее, выделенная часть будет показана в увеличенном масштабе, а при увеличении размера рамки – наоборот. Свойства окна просмотра можно настроить в контекстном меню. Вид окна **Положение&Масштаб** можно изменить двойным щелчком мыши на его заголовке.

**Цветовая схема.** Большинство фигур, вставляемых с трафаретов, имеют черно-белую окраску. По умолчанию, в Visio принята цветовая схема документа **В&W**, которая включает в себя указатели цвета линий, заливки, текста и символов, тени, фона.

Для смены цветовой схемы можно использовать команду **Инструменты/Дополнения/Visio Extras/Color Schemes**. Здесь же, нажав кнопку **Новая** и установив необходимые параметры, можно создать собственную цветовую схему. Как правило, бывает достаточно или выбрать новый стиль в списке **Стиль**, или «подправить» существующую схему.

Для создания новой схемы на основе текущего документа следует нажать кнопку **Использовать цвета текущего документа**, и образцы цвета появятся в соответствующих прямоугольниках. Новая схема добавляется в список **Выбор цветовой схемы**.

Следует иметь в виду, что на элементы фигуры, у которых изменили цвет вручную, схема не действует. Цветовая схема удобна, когда множеству фигур, переносимых с различных трафаретов, необходимо придать одинаковый стиль.

**Стиль фигур.** Стиль включает в себя все общие свойства фигуры и складывается из стилей линии, заполнения, текста. В Visio предусмотрено несколько стандартных стилей, по умолчанию установлен стиль **Normal**. Списки всех доступных стилей открываются в меню **Формат/Стиль** или на дополнительной панели инструментов **Форматировать форму**. Можно отдельно изменить стиль текста, линий, заливки. Смена стиля является самым радикальным средством форматирования фигур, преобразуя ее к другому стандарту.

## 15.2. Соединение фигур

На панели инструментов находится элемент **Соединитель**, с помощью которого осуществляется соединение фигур. Основное отличие линий-соединителей от обычных линий заключается в том, что при перемещении фигуры соединители также перемещаются, оставаясь связанными с точками соединений.

Маркеры соединения (синие крестики) обозначают точки на фигуре, в которых возможно ее соединение с другой фигурой.

В Visio имеются два способа соединения: «фигура к фигуре» и «точка к точке».

1. Способ соединения «фигура к фигуре» является основным. В этом режиме автоматически соединяются соседние фигуры по кратчайшему пути. Чтобы воспользоваться этим методом, надо выбрать инструмент **Соединитель** и перенести с трафарета на лист первую фигуру. При переносе второй и следующих фигур они будут автоматически соединяться. Отменить действие инструмента **Соединитель** можно, нажав кнопку **Указатель** на панели инструментов.

Чтобы соединить фигуры, уже размещенные на листе, надо также выбрать инструмент **Соединитель** и, поместив курсор мыши на фигуру, добиться появления рамки красного цвета – признака выделения фигуры для соединения. Удерживая нажатой левую кнопку мыши, переместить курсор мыши на фигуру, с которой выполняется соединение. Как только на второй фигуре появится рамка соединения красного цвета, кнопку мыши надо отпустить.

2. При способе соединения «точка к точке» перемещение фигур влияет только на геометрию соединителя, а точки соединения фиксированы. Способом «точка к точке» соединяются фигуры, уже размещенные на листе. Сначала надо установить точки соединения, для чего следует выбрать инструмент **Точка соединения**, нажать <Ctrl>

и отметить нужные места. Затем выбрать инструмент **Соединитель** и соединить отмеченные точки, которые в момент наведения курсора мыши выделяются красной рамкой.

Свойства соединителя и вид соединения фигур можно изменить. Для этого надо выделить редактируемое соединение и выполнить команду **Формат/Поведение/Разъем**. В появившемся окне следует выбрать формат соединителей и стиль, вид перемычек-мостиков, появляющихся при пересечении соединителей. (Большое количество соединителей содержится в трафарете **Файл/Объекты/Visio Extras/Connectors**.)

**Соединение форм, их выравнивание.** Для автоматического соединения форм служит команда **Соединить формы** в пункте меню **Формы**. Предварительно надо расположить на листе и выделить все необходимые фигуры.

Для корректировки вида соединителей и структуры соединения используется команда **Формы/Расположить формы**. В окне диалога можно выбрать стиль и направление развития диаграммы от корневого элемента. Можно также сменить способ и точки соединения и выбрать вид соединителей – прямолинейный или дугообразный.

С целью выравнивания фигур применяется команда **Форма/Выровнять формы**. В окне диалога можно выбрать способ выравнивания.

### 15.3. Группировка фигур. Слои и печать

Для сгруппированной фигуры можно выполнять такие операции, как копирование, удаление, перемещение, вращение, изменение размера и защита. Группа формируется следующими действиями:

- выделить несколько фигур контурным прямоугольником или щелчком левой кнопки мыши, удерживая клавишу **<Shift>**;

- выполнить группировку с помощью команды **Форма/Группировка/Группировать** или выбрать команду **Группировать** в контекстном меню.

Чтобы добавить несколько фигур в уже сформированную группу, нужно выделить эти фигуры и группу, затем выполнить команду **Форма/Группировка/Добавить группу**.

Группу, как и фигуру, можно защитить от изменений, выбрав команду **Формат/Защита** и установив нужные флажки.

Обратная операция – разгруппировка – над выделенной группой может быть выполнена, используя команду **Форма/Группировка/**

**Разгруппировать.** Если нужно только изъять одну или несколько фигур, не разрушая группы, следует, выделив их, выбрать команду **Форма/Группировка/Удалить из группы**.

**Порядок следования слоев.** Фигуры на листе располагаются слоями, при этом первая фигура, перенесенная на лист с трафарета (или созданная инструментами), занимает самый нижний слой, далее по возрастанию, и последняя фигура располагается в верхнем слое. Поэтому при наложении вторая фигура закроет первую, третья – вторую и первую и т. д. Порядок следования слоев можно изменить с помощью команды **Форма/Порядок**.

Слой может включать несколько фигур, что используется при построении сложных изображений (окна и двери – один слой, мебель – другой, схема электропроводки – третий). При работе над частью проекта остальные слои можно заблокировать. Некоторые слои создаются автоматически, например все соединители объединяются в один слой.

Чтобы добавить элемент на слой, надо выполнить команду **Вид/Свойства слоя**, в появившемся окне нажать кнопку **Новый**, ввести имя слоя и щелкнуть **<ОК>**. Затем следует выбрать параметры слоя (видимость, блокировка, цвет, прозрачность) и нажать кнопку **Применить**.

Чтобы выделить фигуру в слое, надо выполнить команду **Правка/Выбрать по типу**, при этом в переключателе **Выбрать** надо щелкнуть **Слой**.

Для удаления слоя служит команда **Вид/Свойства слоя**, в появившемся окне необходимо выбрать имя, нажать **Переместить** и **<ОК>**.

**Вывод документа на печать.** Заключительную стадию разработки называют распространением, под этим термином подразумевается экспорт готового документа в другие приложения (или документы других приложений), публикацию в Интернет в виде web-страницы и вывод документа на печать.

Если размер листа изображения отличен от листа печати, то следует выполнить предварительную подготовку документа и настройку принтера, используя **Файл/Настройка страницы**. В появившемся окне для изменения размеров страницы на вкладке **Настройка печати** в разделе **Масштаб печати** надо установить переключатель в положение **Увеличить до...** и задать коэффициент сжатия; установив переключатель в положение **Разместить в...** в полях **Таблица вокруг** и **Таблица внизу**, задать число страниц, на которое будет поделен лист чертежа при печати, а коэффициент сжатия в этом случае будет установлен автоматически.

Дополнительные настройки можно выполнить, нажав кнопку **Настройка**, и в появившемся окне задать размер печатного листа и поля

страницы. Чтобы страница располагалась по центру печатного листа, следует установить флажки **Центрировать вертикально**, **Центрировать горизонтально**.

Пункт меню **Файл/Печать** открывает окно настроек принтера.

#### 15.4. Пользовательская среда разработки

**Стиль.** Для использования уже имеющегося стиля служит кнопка **Формат** на панели **Стандартная**. Применение его известно по офисным приложениям: надо выделить фигуру-образец, нажать кнопку **Формат** на панели инструментов, перевести курсор на нужную фигуру и щелкнуть левой кнопкой мыши.

Чтобы выполнить операцию многократно, следует дважды щелкнуть по кнопке **Формат** и осуществить копирование формата образца на несколько фигур. Для выхода из режима необходимо нажать **<Esc>**.

Создать новый стиль можно с помощью команды **Формат/Назначить стили**. В окне диалога следует выбрать или задать необходимые параметры. Новый стиль теперь может быть назначен существующим на листе фигурам.

**Создание новых трафаретов и мастеров.** Трафарет содержит набор фигур (мастеров) определенного назначения. Для создания нового трафарета служит команда **Файл/Объекты/Новый шаблон**, после выполнения которой в главном окне на панели трафаретов появится заготовка – пустое окно с заголовком, в котором на фоне иконки в левом углу располагается красная звездочка – признак режима редактирования. Чтобы изменить имя трафарета и другие свойства, надо щелкнуть правой кнопкой мыши на заголовке (или левой – на значке) – появится контекстное меню, в котором следует выбрать команду **Свойства (Properties)**. На вкладке **Общие** нужно заполнить поля **Титул** – краткое имя и **Тема** – полное наименование трафарета. Остальные поля носят информационный характер.

Заполнение трафарета – это перетаскивание фигур с листа рисунка или других трафаретов с помощью мыши на панель трафаретов. Можно также использовать команды **Копировать** и **Вставить** из контекстного меню трафаретов.

Первая перемещенная фигура получает имя **Master.0**, далее **Master.1** и т. д. При перемещении мастера со стандартного трафарета его имя не изменится. Чтобы изменить описание перемещенной фигуры, надо в контекстном меню выполнить команду **Master Properties**,

в окне диалога задать новое имя, в поле **Подсказка** ввести текст всплывающей подсказки, выбрать размер иконки и способ выравнивания. Закончив заполнение трафарета и редактирование мастеров, надо сохранить его с помощью контекстного меню.

Можно создать пользовательский каталог личных трафаретов, но лучше разместить его в папке **Solutions**, в которой находятся все стандартные трафареты. Тогда новый пользовательский трафарет становится *зарегистрированным* и доступным в любой момент, как и другие трафареты. После перезагрузки приложения его можно видеть в списке.

Для создания нового мастера надо в контекстном меню выполнить команду **Новый мастер**, откроется окно диалога, в котором следует задать параметры: имя, подсказку и др. После закрытия окна диалога значок мастера появится на трафарете, после чего можно отредактировать фигуру мастера. Для нового мастера, как правило, следует сменить значок, отображаемый на трафарете. Чтобы сделать это, надо вызвать специальный редактор командой **Редактировать значок** из контекстного меню. Откроется окно с мозаичным полем и контуром фигуры на нем. Выполнив все операции, надо отменить режим редактирования, снова выполнив команду **Редактировать**. Затем изменения сохранить.

Изменение мастера возможно только в режиме редактирования. Для этого надо щелкнуть правой кнопкой мыши на заголовке (или левой – на значке) трафарета, в контекстном меню выбрать команду **Редактировать**, тогда на значке трафарета появится красная звездочка – признак режима редактирования. Следует выбрать значок нужного мастера и в контекстном меню выполнить команду **Редактировать мастер**. Откроется новое окно с фигурой мастера, которую можно изменить с помощью инструментов. Затем закрыть окно редактирования.

**Создание новых шаблонов.** Поскольку пользовательские стили, мастера, трафареты создаются для многократного применения, целесообразно сохранить их в виде нового шаблона, который может быть создан на основе стандартного шаблона или готового документа. Для этого нужно выполнить следующие действия:

1. Открыть существующий шаблон или документ командой **Файл/Открыть**. Если исходный документ надо сохранить, в списке настроек кнопки **Открыть** следует установить флажок **Копировать** – будет загружена копия документа.

2. Открыть командой **Файл/Объекты/Открыть шаблон** необходимые трафареты.

3. Установить настройки листа.

4. Изменить параметры фигур и стили.

5. Создать, если необходимо, новые трафареты.



6. Сохранить текущий документ как новый шаблон: в окне диалога **Сохранить как...** в поле **Сохранить в...** выбрать каталог, в поле **Файл** задать имя шаблона, в списке типов выбрать **Шаблон**.

Чтобы иметь возможность загружать шаблон при запуске приложения как стандартный и иметь к нему доступ через меню, следует выполнить *регистрацию шаблона*. Для этого надо при сохранении выбрать каталог **Solutions** так же, как и в случае регистрации трафарета. Шаблоны собраны в подкаталоги по категориям. Если пользовательский шаблон не относится ни к одной из стандартных категорий, рекомендуется создать в каталоге **Solutions** собственную папку. Редактируется шаблон как обычный документ.

### 15.5. Задание для выполнения на компьютере

Нарисовать технологическую схему, представленную ниже.

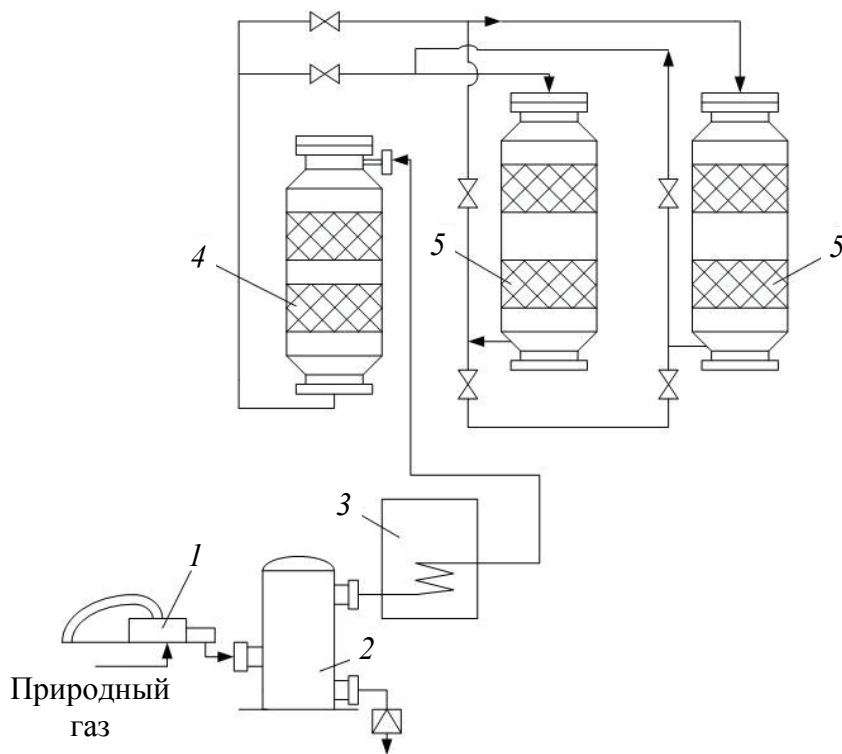


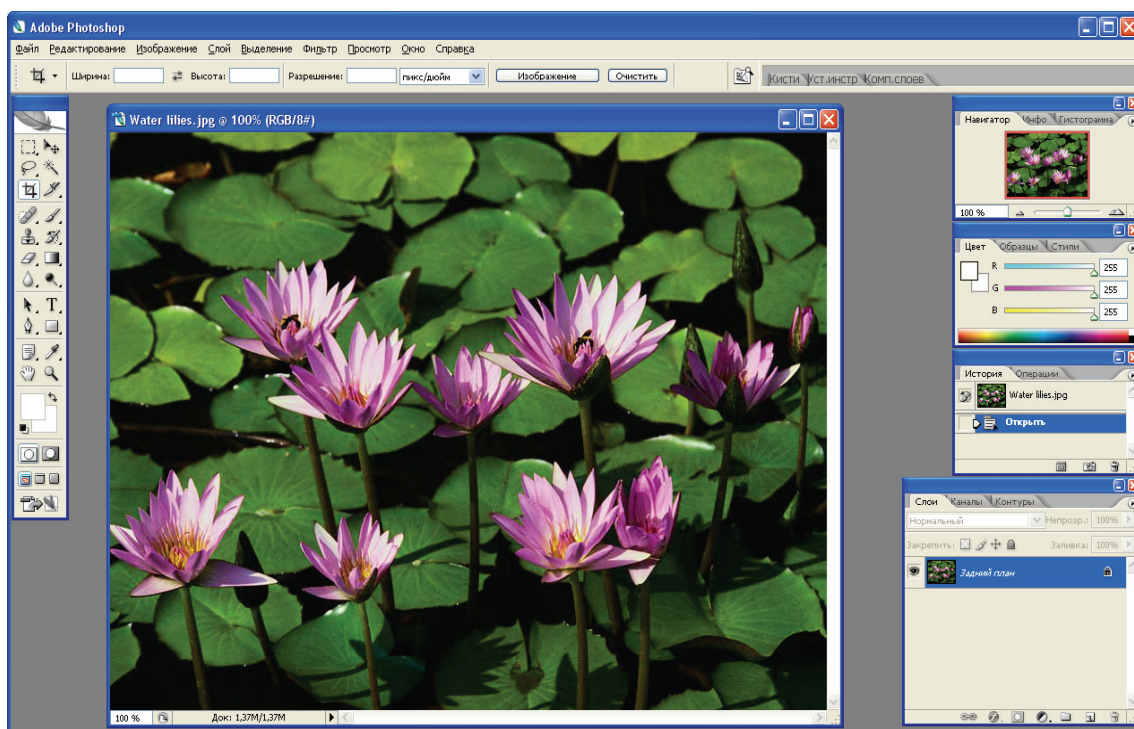
Схема двухступенчатой сероочистки природного газа с использованием полочных адсорберов

На первом этапе работы создать новые пользовательские мастера и новый трафарет, которые будут использованы при рисовании схемы. Определить стиль изображения. В качестве фона выбрать слой с логотипом фирмы.

## 16. ГРАФИЧЕСКИЙ РЕДАКТОР ADOBE PHOTOSHOP

Графический редактор Adobe Photoshop (рисунок) предназначен для обработки и создания изображений растровой графики.

Основные операции выполняются с помощью панели инструментов. Инструмент в Photoshop выбирается щелчком мыши по определенному инструменту.



Окно графического редактора Adobe Photoshop

Рядом с пиктограммами большинства инструментов расположена маленькая стрелочка. Если щелкнуть по такому инструменту правой клавишей мыши, раскроется подменю, в котором можно выбрать инструмент, входящий в группу. Если навести курсор на пиктограмму инструмента и подержать несколько секунд, то всплывет подсказка.

Для каждого инструмента задаются свои собственные параметры (например, режим смешивания, значение непрозрачности и т. п.) на панели опций, расположенной сверху экрана.

Параметры на панели опций изменяются в зависимости от того, какой инструмент выбран.

## 16.1. Основные инструменты и окна

**Группа инструментов выделения.** Инструмент **Rectangular Marquee (Выделение)** служит для выделения прямоугольных, овальных областей, **Move (Перемещение)** – для перемещения выделенной области, **Lasso (Лассо)** – для выделения произвольной области, **Magic Wand (Волшебная палочка)** – для выделения элемента по цвету, **Crop (Кадрирование)** – для выделения и обрезки прямоугольной части изображения, **Slice (Фрагменты)** – для работы с фрагментами изображения.

**Инструменты рисования и редактирования.** Инструмент **Healing Brush (Восстанавливающая кисть)** предназначен для устранения дефектов изображений, **Brush (Кисть)** – для рисования, **Clone Stamp (Штамп)** – для копирования в местах своего прохода некоторого выборочного образца для ретуши, **History Brush (Архивная кисть)** – для работы с историей создания изображения, **Eraser (Ластик)** – для стирания или окрашивания в цвет фона, **Blur (Размытие)** – для размытия изображения, **Dodge (Осветление)** – для осветления изображения, **Path Selection (Выделение контура)** – для прямого выделения, **Rectangle (Прямоугольник)** – для рисования залитых цветом прямоугольников, эллипсов, линий.

**Специализированные инструменты.** Инструмент **Type (Текст)** используется для работы с текстом, **Pen (Перо)** – для работы с контурами, **Eyedropper (Пипетка)** – для взятия цвета переднего плана, при нажатии клавиши <Alt> формирует цвет заднего плана, **Notes (Комментарий)** – для создания примечаний, **Gradient (Градиент)** выполняет градиентную многоцветную заливку.

**Наведение и выбор цвета.** Инструмент **Hand (Рука)** служит для перемещения изображения в окне документа, **Zoom (Масштаб)** изменяет масштаб изображения.

Инструмент **Foreground Color (Основной цвет)** предназначен для вызова палитры **Color Picker** для выбора цвета, **Background Color (Цвет фона)** – для установки цвета фона.

**Плавающие окна инструментов.** Плавающие окна (палитры) вызываются из меню **Window (Окно)**. Каждое окно обладает собственным набором опций. Щелчок по кнопке со стрелкой в правом верхнем углу окна открывает меню команд. С помощью этих команд можно манипулировать опциями окна и задавать установки.

В окне **Navigator (Навигатор)** можно управлять масштабом отображения документа. На вкладке **Info (Информация)** отражена геометрическая и цифровая информация о выбранных пикселях

изображения. На вкладке **Options (Опции)** показан допустимый набор опций для текущего инструмента. Если поменять инструмент, набор опций изменится.

В окне **History (История)** фиксируется история действий пользователя. Любое действие можно отменить щелчком мыши на названии выполненной операции.

В окне **Layers (Слои)** показаны слои изображения. Они располагаются в порядке видимости сверху вниз. Основным является нижний слой **Background (Фон)** и его нельзя удалить. Значок «глаз» показывает видимый слой; значок кисточки – активный слой, который редактируется, а значок связи («звено цепи») – сцепленные слои. **Opacity (Непрозрачность)** означает величину непрозрачности слоя.

Окно **Color (Цвет)** служит для выбора цвета. Цвет можно выбирать ползунками или вводя значения в поля.

## 16.2. Создание изображений

Команда меню **File/New (Файл/Новый)** вызывает появление окна. Следует ввести имя файла в поле **New (Новый)**, размер изображения в пикселях (или в сантиметрах, дюймах и т. д.) в полях **Width (Ширина)** и **Height (Высота)**; установить величину разрешения изображения **Resolution (Разрешение)**, например 72 пикселя для изображений, публикуемых в Интернет, или другое для печати; выбрать цветовой режим (**Mode**):

- **Bitmap** (битовый) – только черный и белый цвет;
- **Grayscale** (градации серого) – 256 оттенков серого;
- **RGB Color** – все цвета;
- **CMYK Color** – все цвета (для печати);
- **Lab Color** – все цвета (для печати).

Тип изображения, определенный при создании нового документа, может изменяться командой меню **Image/Mode (Изображение/Режим)**.

Далее надо выбрать фон в пункте **Contents (Содержимое фона)**:

- **White (Белый цвет)** – фон документа окрашивается в белый цвет;
- **Background Color (Фоновый цвет)** – фон окрашивается в текущий фоновый цвет;
- **Transparent (Прозрачная основа)** – создаются документы, состоящие из одного слоя и не содержащие фоновый цвет.

*Создание простых изображений.* Цвета фона и основного цвета выбираются в окне **Color Picker (Цветовая палитра)**, которое появ-

ляется после щелчка по инструменту **Background Color (Фон)** или **Foreground Color (Основной цвет)** на панели инструментов.

Цвет можно определить и с помощью инструмента **Eyedropper (Пипетка)**, если щелкнуть им по определенной точке изображения.

При выборе инструмента **Brush Tool (Кисть)** в верхней части экрана появляется панель опций. Здесь можно задать тип кисти, уровень прозрачности изображения.

С помощью кисти можно рисовать не только линии различной кривизны, но и прямые. Для этого надо отметить щелчком мыши начало линии, затем нажать клавишу **<Shift>** и, удерживая ее, щелкнуть в другом месте, чтобы завершить линию.

В эту же группу входит инструмент **Pencil Tool (Карандаш)**, которым можно рисовать тонкие четкие линии.

Инструмент **Rectangle Tool (Прямоугольник)** позволяет рисовать прямоугольники простые и скругленные, круги и овалы, линии, пятиугольники. Инструмент **Line (Линия)** служит для изображения прямых линий.

Удалить часть изображения можно инструментом **Eraser (Стирание)**. Для отмены последней операции служит команда **Edit/Undo (Правка/Отменить)**.

**Заливка области.** Инструмент **Paint Bucket (Заливка)**, который располагается во всплывающей палитре инструмента **Gradient (Градиент)**, заливает замкнутую область основным цветом или узором с учетом заданного значения параметра **Tolerance (Допуск)**. На панели опций можно установить цвет и определить режим смешивания, а также прозрачность. Чем выше значение допуска **Tolerance (Допуск)**, тем шире диапазон цветов, которые будет заменять инструмент **Paint Bucket (Заливка)**. Если установлен флажок **Anti-aliased (Сглаживание)**, то сглаживаются границы области заливки, а установка флажка **Contiguous (Смежные)** позволяет залить только те области, которые соединены с областью, по которой делается щелчок. После установки всех параметров надо щелкнуть по нужной области изображения.

Если на панели опций выбрать узор, то можно подобрать подходящий рисунок, которым будет производиться заливка.

**Режимы смешивания.** Режим смешивания (**Mode**) можно выбрать не только на панели опций, но и в окне **Layers (Слои)** или в диалоговых окнах **Fill (Залить)**, **Stroke (Очертить)**, **Fade (Ослабить)** или **Fill Path (Залить контур)**. Режим смешивания определяет, как инструмент будет изменять пиксели, находящиеся под ним. Например, в режиме **Dissolve (Растворение)** создается похожая на известковую, сухая текстура.

### 16.3. Слои в графическом редакторе Adobe Photoshop

Слои можно сравнить с листами прозрачной пленки с нанесенными на них фрагментами изображения. При сложении таких листов получается итоговое изображение.

Для того чтобы создать слой, надо в меню **Layer (Слой)** выполнить команду **New/Layer (Новый/Слой)** либо, нажав клавишу <Alt>, щелкнуть по кнопке **Create New Layer** в нижней части окна **Layers**. Затем следует ввести имя нового слоя. Можно выбрать значение параметров **Mode (Режим смешивания)** и **Opacity (Непрозрачность)**.

В окне **Layers** перечислены все слои изображения. Миниатюра слева от имени слоя в уменьшенном виде отражает его содержимое. Для того чтобы сделать слой активным, надо щелкнуть мышью на имени нужного слоя. Одновременно может быть активен только один слой документа.

Чтобы временно спрятать слой, следует щелкнуть мышью по значку «глаз». Чтобы вновь сделать слой видимым, надо щелкнуть мышью по значку «глаз» еще раз.

Порядок расположения слоев определяет их видимость сверху вниз. Порядок следования слоев можно изменить, для этого надо зацепить мышью один слой и затащить его вниз под другой слой.

Изображение, содержащее слои, может быть сохранено только в файлах формата **\*.psd**.

**Эффекты слоев и операции над ними.** К слою можно применить специальные эффекты: **Drop Shadow (Отбросить тень)**, **Inner Shadow (Внутренняя тень)**, **Outer Glow (Внешнее свечение)**, **Inner Glow (Внутреннее свечение)**, **Bevel and Emboss (Тиснение)**, **Satin (Глянec)**, **Color Overlay (Наложение цвета)**, **Gradient Overlay (Наложение градиента)**, **Pattern Overlay (Наложение узора)** и **Stroke (Кайма)**.

Для того чтобы задействовать эффект слоя, надо дважды щелкнуть на имени слоя. В появившемся окне **Layer Style (Стиль слоя)** следует выбрать эффект и установить параметры эффекта. К одному слою можно применить несколько эффектов.

Для удаления слоя надо щелкнуть правой кнопкой мыши на его имени, из контекстного меню выбрать команду **Delete Layer (Удалить слой)** и щелкнуть по варианту ответа **Yes**.

Чтобы слить два слоя, необходимо активизировать верхний слой, из меню палитры **Layers (Слои)** выбрать команду **Merge Down**

(Слить с нижним). Активный слой будет слит со слоем, расположенным непосредственно под ним.

Для объединения всех слоев служит команда **Layers/Flatten Image (Слои/Выполнить сведение)**, которая объединяет видимые в данный момент слои, располагает результат на месте самого нижнего видимого слоя и удаляет скрытые.

#### 16.4. Выделение областей

Чтобы выделить весь рисунок или весь слой, надо выбрать команду меню **Select/All (Выделение/Все)**.

Для выделения отдельных частей изображения используются инструменты **Marquee (Выделение)**, **Lasso (Лассо)**, **Magic Wand (Волшебная палочка)**, команда меню **Select/Color Range (Выделение/Цветовой диапазон)**. Для любого инструмента выделения, если при повторном выделении удерживать клавишу **<Shift>**, то области складываются друг с другом, если удерживать **<Alt>**, то новая область вычитается из предыдущей.

Группа **Marquee** содержит следующие инструменты: **Rectangular Marquee (Прямоугольная область)** выделяет прямоугольные и квадратные области (для создания пропорционального квадрата при выделении нужно удерживать клавишу **<Shift>**); **Elliptical Marquee (Овальная область)** выделяет круглые и овальные области; **Single Row Marquee (Область: горизонтальная строка)** выделяет один пиксель изображения по горизонтали; **Single Column Marquee (Область: вертикальная строка)** выделяет один пиксель по вертикали.

Для выделения по контуру используется инструмент **Lasso**. В группу **Lasso** входят инструменты выделения областей произвольной формы: собственно **Lasso (Лассо)**, **Polygonal Lasso (Полигонное лассо)** и **Magnetic Lasso (Магнитное лассо)**. Инструмент **Polygonal Lasso** предназначен для выделения многоугольников. При выделении в каждой точке изгиба надо щелкать курсором мыши. **Magnetic Lasso** удобно использовать при выделении объектов неправильной формы.

Иногда выделение объекта удобнее выполнить с помощью инструмента **Magic Wand (Волшебная палочка)**. Принцип действия инструмента заключается в выделении лежащих рядом близких по цвету точек изображения. При этом можно задать степень близости цвета **Tolerance (Допуск)**, необходимую для того, чтобы пиксель попал в выделенную область.

С помощью команды **Select/Color Range (Выделение/Цветовой диапазон)** можно выделить области в зависимости от цвета изображения или задать диапазон яркости или тона. Используя такое выделение, можно ограничить выделяемую область указанием стандартного диапазона цветов, например **Reds (Красные)**, **Yellows (Желтые)**; диапазона яркости **Highlights (Яркие тона)**, **Midtones (Средние тона)** или **Shadows (Темные тона)** или конкретного цвета, выбранного с помощью пипетки.

Для того чтобы смягчить границы выделенной области, можно воспользоваться командой **Select/Feather (Выделение/Растушевка)**.

Чтобы поменять местами выделенную и невыделенную области, нужно выполнить команду **Select/Inverse (Выделение/Инверсия)**. Например, довольно просто выделить фигуру на однотонном фоне, если активизировать инструмент **Magic Wand (Волшебная палочка)**, на панели опций ввести в поле **Tolerance (Допуск)** значение 5 или меньше, щелкнуть по однотонному фону, чтобы выделить его целиком, затем инвертировать области.

Выделенные области можно вырезать, копировать, вставлять, удалять обычным образом. Можно трансформировать (наклонять, искажать, менять размер) с помощью команды **Select/Transform Selection (Выделение/Преобразовать выделение)** или команды **Edit/Transform (Редактирование/Трансформировать)**.

Можно сделать обводку выделенного участка командой **Edit/Stroke (Правка/Выполнить обводку)**. В появившемся окне следует выбрать ширину линии, цвет, определить, где нужна обводка: внутри (**Inside**), в центре (**Center**) или снаружи (**Outside**). После нажатия клавиши <ОК> появится нужная фигура.

Для перемещения выделенного фрагмента служит инструмент **Move Tool (Движение)**.

Чтобы снять любое выделение, надо щелкнуть мышью в любой области вне выделенного изображения.

## 16.5. Сохранение файлов с изображениями

Для сохранения файла служит команда **File/Save As... (Файл/Сохранить как...)**. В появившемся окне указывается тип файла, имя файла и выбирается нужная папка. Если в рисунок внесены изменения и их надо сохранить на прежнем месте, то используется команда **File/Save (Файл/Сохранить)**. Основными форматами для сохранения документа в Photoshop являются форматы **\*.psd**, **\*.tiff**, **\*.jpg**, **\*.gif**.



В формате **\*.psd** сохраняется рабочий документ Photoshop. Размер файла при этом наибольший. Формат **\*.tiff** является универсальным для многих приложений. Если надо перенести изображение из одной программы в другую без потери качества, то лучше сохранять рисунок в этом формате.

Форматы **\*.jpg** и **\*.gif** используются для сохранения картинок для web. Формат **\*.gif** пригоден для ограниченных по цвету рисунков. Формат **\*.jpg** используется для изображений с полноцветной графикой, например фотографий.

Photoshop имеет специальную команду для сохранения файлов формата **\*.jpg** и **\*.gif**, размещаемых в Интернете: **File/Save for Web (Файл/Сохранить для Web)**. Команда предназначена для оптимизации сохраняемого файла. При этом можно выбирать образцы сохраняемого документа с тем, чтобы подобрать формат наименьшего размера и наивысшего качества. Например, можно выбрать в окне **Save for Web** вкладку **4-Up** (4 образца). В правой части окна задать формат **\*.gif** или **\*.jpeg**. Формат **\*.jpeg** имеет три варианта: **High, Medium, Low** (соответственно высокого качества, среднего и низкого).

## 16.6. Задание для выполнения на компьютере

1. Создать изображение на одном слое и другое изображение – на втором. Использовать различные инструменты рисования и режимы смешивания. Опробовать различные эффекты слоев.

2. Создать третий слой, на который скопировать выделенные различным образом части первых двух слоев. Изменить прозрачность слоев. Сохранить рисунок в формате **\*.psd**.

3. Выполнить сведение слоев и сохранить рисунок в формате **\*.jpg**. Сравнить объем полученных файлов.

## 17. ОБРАБОТКА ИЗОБРАЖЕНИЙ В ADOBE PHOTOSHOP

### 17.1. Просмотр рисунка, изменение размеров и поворот

Для увеличения масштаба изображения надо выбрать лупу **Zoom (Масштаб)** и щелкнуть по рисунку или выделенному фрагменту рисунка. Для уменьшения масштаба изображения следует щелкать по лупе при нажатой клавише **<Alt>**. Чтобы увидеть реальный размер изображения, нужно выбрать на панели опций **Actual Pixels (Реальный размер)**.

Удобно пользоваться «горячими» клавишами: для уменьшения масштаба одновременно с окном документа надо нажимать **<Ctrl->**, для увеличения – **<Ctrl+>**.

Инструмент **Hand (Рука)** служит для перемещения изображения, которое не умещается в рабочем окне.

Размер изображения можно изменить, выбрав команду меню **Image/Image Size (Изображение/Размер изображения)**. Если в появившемся окне ввести в поле **Width (Ширина)** другое значение, то параметр **Height (Высота)** установится автоматически в том случае, если стоит галочка у переключателя **Constrain Proportions (Соблюдать пропорции)**. После нажатия **<ОК>** картинка изменит свои размеры пропорционально или, если галочка у переключателя не стоит и значение высоты установлено вручную, не пропорционально.

**Поворот изображений.** Для того чтобы перевернуть изображение справа налево, можно выполнить команду **Image/Rotate Canvas/Flip Canvas Horizontal (Изображение/Повернуть холст/Перевернуть вдоль горизонтали)**. Чтобы перевернуть изображение сверху вниз и создать зеркальное изображение, можно выбрать **Image/Rotate Canvas/Flip Canvas Vertical (Изображение/Повернуть холст/Перевернуть холст вдоль вертикали)**.

Для того чтобы повернуть изображение на predeterminedенный угол, можно использовать команду **Image/Rotate Canvas/180°, 90° CW или 90° CCW (Изображение/Повернуть холст/На 180°, 90° по часовой стрелке или на 90° против часовой стрелки)**.

Для поворота на произвольный угол служит команда **Image/Rotate Canvas/Arbitrary (Изображение/Повернуть холст/Произвольно)**. Затем в поле **Angle (Угол)** надо ввести число в пределах от – 359.99

до 359.99°. Далее следует поставить переключатель в положение °CW (По часовой стрелке) или °CCW (Против часовой стрелки) и щелкнуть по кнопке <ОК>.

## 17.2. Работа с текстом

Текст создается с помощью инструмента **Type (Текст)** на отдельном слое.

Для ввода текста надо активизировать инструмент **Type (Текст)** и очертить мышью рамку для ввода текста. На панели опций можно щелкнуть по кнопке **Type**, чтобы поменять ориентацию текста (по горизонтали/по вертикали), можно выбрать шрифт. При необходимости в поле **Anti-aliased (Сглаживание)** можно указать один из методов сглаживания границ символов: **Sharp (Резкое)**, **Crisp (Жесткое)**, **Strong (Сильное)** или **Smooth (Гладкое)**. При выборе варианта **None (Не применять сглаживание)** у символов будут неровные края. Можно выбрать цвет текста, щелкнув по полю образца на панели опций и выбрав цвет на всплывающей панели. При щелчке по кнопке **Toggle the character and paragraph palettes (Палитры шрифта и абзаца)** открываются палитры **Character (Шрифт)** и **Paragraph (Абзац)**, где можно выбрать необходимые атрибуты. При нажатии кнопки **Create warped text (Деформация текста)** открывается окно, в котором выбирается вид изменения текста: изогнутый, закрученный, в виде волн и т. п. После настройки всех параметров можно вводить текст.

Для того чтобы отредактировать текст, надо его предварительно выделить. Можно изменять атрибуты текста: шрифт, стиль, цвет, межстрочный интервал, выравнивание. Также допускается изменение содержания текста, применение к нему разных эффектов, изменение режимов смешивания и уровня непрозрачности.

Чтобы применить фильтры к тексту, его необходимо преобразовать в растровый формат с помощью команды меню **Layer/Rasterize/Type (Слой/Растрировать/Текст)**. Но надо иметь в виду, что, как только текст будет преобразован в растровый формат, его типографские атрибуты (например, шрифт или стиль) уже нельзя будет изменить.

## 17.3. Корректировка изображений

Осветлить и затемнить фрагменты изображения можно с помощью инструментов корректировки **Dodge (Осветление)** и **Burn**

(**Затемнение**). Для каждого инструмента можно задать вид кисти и другие опции.

Инструмент **Sponge (Губка)** служит для изменения насыщенности цветных областей. При использовании инструмента на панели опций можно выбрать вид кисти, а также установить значение в поле **Mode (Режим): Desaturate (Обесцветить)** или **Saturate (Окрасить)**. Можно задать величину параметра **Pressure (Нажим)** в диапазоне от 1 до 100%.

Инструмент **Smudge (Палец)** используется для размазывания цветов. На панели опций можно выбрать тип кисти. Режим смешивания в поле **Mode (Режим)** позволяет размазывать все оттенки цвета, если выбрано **Normal (Нормальный)**; добавляются темные цвета в более светлые, если выбрано **Darken (Затемнить)**; светлые цвета добавляются в более темные, если выбрано **Lighten (Осветлить)**. Регулируется также параметр **Pressure (Давление)**.

Инструмент **Clone Stamp (Клонирующий штамп)** служит для переноса клона объекта. После активизации инструмента на панели опций инструмента надо щелкнуть по стрелке поля **Brush (Кисть)** и выбрать размер кисти. Можно задать режим смешивания в поле **Mode (Режим)**, а также значение параметра **Opacity (Непрозрачность)** в процентах. В окне рисунка, нажав клавишу <Alt>, надо щелкнуть по области слоя, которая копируется. Затем нажатием левой клавиши мыши в том месте, где должен появиться клон, переносится изображение.

Инструмент **Healing Brush (Корректирующая кисть)** предназначен для устранения дефектов изображений. С его помощью можно с минимальными усилиями восстановить поврежденные фрагменты изображений волосистых покровов или удалить с фотографии родинки, веснушки и другие мелкие детали.

**Команды коррективки.** Использование команд из подменю **Image/Adjustments (Изображение/Коррективки)** позволяет изменить яркость, тон, резкость и тому подобные характеристики рисунка.

Команда **Image/Adjustments/Brightness/Contrast (Изображение/Коррективки/Яркость/Контраст)** служит для корректировки яркости и контраста. В появившемся окне движением ползунка или заданием значения в поле **Brightness (Яркость)** устанавливается нужное значение яркости, а контраста – в поле **Contrast (Контраст)**.

Команда **Image/Adjustments/Color Balance (Изображение/Коррективки/Баланс цвета)** корректирует цветовые оттенки рисунка, отдельного слоя или выделенной области. В диалоговом окне можно

отметить диапазон корректировки: **Shadows** (Темные тона), **Midtones** (Средние тона) или **Highlights** (Светлые тона).

Команда **Image/Adjustments/Hue/Saturation** (Изображение/Корректировки/Цветовой тон/Насыщенность) позволяет изменить цветовую насыщенность рисунка, слоя или выделенной области.

## 17.4. Использование фильтров

Фильтр можно применить как к слою в целом, так и к его выделенной части. Прежде чем применить фильтр к выделенной области, можно растушевать ее, чтобы она не слишком выделялась на изображении после фильтрации.

Некоторые фильтры можно применить непосредственно (выбрав соответствующую команду из подменю). Другие же фильтры можно использовать, только установив определенные значения переменных в диалоговом окне.

Диалоговые окна большинства фильтров имеют окно предварительного просмотра **Preview** (Предварительный просмотр). Изображение в таком окне передвигается мышкой.

С помощью команды **Edit/Fade** (Правка/Ослабить) можно ослабить эффект от применения фильтра.

Для усиления эффекта фильтрации надо выбрать команду меню **Image/Adjustments/Levels** (Изображение/Корректировки/Уровни) и в появившемся диалоговом окне **Levels** (Уровни) переместить ползунки нужным образом.

**Коррекция резкости и устранение мелких дефектов изображения.** Весь рисунок или его выделенную часть с помощью фильтра **Filter/Sharpen** (Фильтр/Резкость) можно сделать более резким.

Для устранения небольших погрешностей изображения служит фильтр **Filter/Noise/Dust & Scratches** (Фильтр/Шум/Пыль и трещины). При использовании фильтра в диалоговом окне в поле **Radius** (Радиус) можно установить значение 1 пиксель, этот параметр определяет область, в которой выявляются отклонения. Чем больше радиус, тем большая область усредняется и обрабатывается фильтром. В поле **Threshold** (Порог) определяется диапазон отклонения от цветовых значений пикселей.

**Пример создания объемного текста.** Пусть на белом фоне написан любой текст крупным шрифтом, например 150 pt. Надо выделить текст, щелкнув по текстовому слою в палитре **Layers** (Слои), скопировать в

буфер обмена, создать новый слой и вставить содержимое буфера. К новому слою можно применить фильтр **Filter/Sketch/Plaster** (Фильтр/Эскиз/Рельеф), подобрав желаемые настройки. Для придания буквам большего объема надо применить фильтр **Filter/Sketch/Bas Relief** (Фильтр/Эскиз/Барельеф) и создать внутреннюю тень **Layer/Layer Style/Inner Shadow** (Слой/Стиль слоя/Внутренняя тень). Текст будет смотреться эффектней, если создать для него обычную тень: **Layer/Layer Style/Drop Shadow** (Слой/Стиль слоя/Тень).

*Пример окантовки картинкой рамкой с рваными краями.* Первоначально на холсте белого цвета надо создать изображение и задать цвет окантовки в поле **Background** (Фон) на панели инструментов. Чтобы создать саму окантовку, следует выполнить команду **Image/Canvas Size** (Изображение/Размер холста). В диалоговом окне команды **Canvas Size** поставить флажок у переключателя **Relative** (Относительно) и задать ширину и высоту окантовки соответственно в полях **Width** и **Height** (например по 1). Затем надо выбрать инструмент **Rectangular Marquee** (Прямоугольная область). На панели опций этого инструмента следует ввести значение (например 8) в поле **Feather** (Растушевка) и нарисовать пунктирный контур вокруг примерно трех четвертей изображения (не включая его окантовки). Из меню **Select** (Выделение) выбрать команду **Inverse** (Инвертировать).

Далее надо выполнить команду меню **Filter/Distort/Ripple** (Фильтр/Деформация/Рябь), переместить изображение в окне предварительного просмотра так, чтобы была видна граница изображения, и подобрать нужный вид, изменяя ползунком параметры. Затем можно выполнить команду **Filter/Brush Strokes/Spatter** (Фильтр/Мазки кисти/Разбрызгивание).

*Пример создания надписи «под водой».* На холсте темно-серого цвета с использованием светло-серого основного цвета необходимо создать изображение воды, применив фильтры **Filter/Render/Clouds** (Фильтр/Рендеринг/Облака) и **Filter/Sketch/Chrome** (Фильтр/Эскиз/Хром). С использованием команды **Image/Adjustment/Color Balance** (Изображение/Корректировки/Баланс цвета) следует раскрасить воду.

Надо создать новый слой, установить цвет фона – черный, основной цвет – белый, применить фильтры **Clouds** (Облака) и **Chrome** (Хром), раскрасить слой, сделав его чуть темнее предыдущего. Затем следует сделать этот слой прозрачным, установив значение параметра **Opacity** (Непрозрачность) примерно 50%.

На нижнем слое необходимо напечатать текст, перевести его в растровый формат, для чего щелкнуть правой клавишей мыши по слою текста в окне **Layers (Слои)** и в контекстном меню выполнить команду **Rasterize Layer (Растрировать слой)**. Также сделать слой полупрозрачным, установив значение **Opacity (Непрозрачность)**, равное 60%. Применить к тексту фильтр **Filter/Distort/Wave (Фильтр/Искажение/Волна)**.

### 17.5. Создание коллажа

Пусть имеется два рисунка: первый – с изображением дерева, второй – с изображением птички. Надо создать коллаж, в котором птичка сидит на дереве.

Чтобы выделить птичку, надо выбрать инструмент **Magic Wand (Волшебная палочка)**, установить на панели опций **Tolerance (Допуск)**, например 10. Щелкнуть по фону вокруг птички и выполнить команду **Select/Inverse (Выделение/Инверсия)**. Птичка выделится. Изменить размеры птички следует командой **Edit/Free Transform (Редактирование/Трансформировать)**. Затем выбрать любой инструмент и в появившемся окне нажать кнопку **Apply (Применить)**.

Инструментом **Move (Движение)** изображение птички нужно перетащить на картинку с деревом. В файле с этой картинкой добавится новый слой. Можно еще подкорректировать размеры и наклон птички командой **Edit/Free Transform (Редактирование/Трансформировать)**. Если изображение птички должно выглядеть более бледно, то можно в окне **Layers (Слои)** для слоя с птичкой установить величину **Opacity (Непрозрачность)**, например, 80%. Там же можно изменить режим смешивания.

Для создания прозрачного цветового перехода (градиента) надо создать новый слой над фоновым. Нажать на панели инструмент **Gradient (Градиент)**. На панели опций задать значение **Opacity (Непрозрачность)**, равное 60%, выбрать из списка пункт **Gradient Foreground to Transparent (От основного к прозрачному)**, установить флажок **Transparency (Прозрачность)**. Выбрать нужный цвет пипеткой и провести мышью от правого края до левого посередине картинки.

Теперь надо сделать надпись: выбрать инструмент **Type (Текст)**, очертить окно для текста, на панели опций выбрать цвет, определить параметры шрифта и напечатать текст. Можно изменить атрибуты

текста, предварительно его выделив. К надписи можно применить эффект, выполнив команду **Layer/Layer Style/Bevel and Emboss** (Слой/Стиль слоя/Тиснение). В окне **Layers** (Слои) в слое с текстом, рядом с *T* появилось *f* в кружочке. Двойной щелчок по *f* позволяет редактировать эффекты, наложенные на текст.

Если коллаж готов, то можно выполнить сведение слоев.

### **17.6. Задание для выполнения на компьютере**

1. Опробовать вышеприведенные примеры. На их основе создать свои изображения с использованием различных фильтров.
2. Создать надпись на рисунке и применить к ней различные эффекты.
3. Разработать коллаж, используя различные изображения.



## 18. ПРИЛОЖЕНИЕ MATHCAD

Приложение Mathcad предназначено для выполнения различных расчетов, для работы с графическими изображениями.

Объединение текстового, формульного и графического редакторов с вычислительным ядром позволяет готовить электронные документы с высоким качеством оформления, способные выполнять расчеты с наглядной демонстрацией результатов.

Для удобства работы с математическими выражениями в приложении имеется панель **Математика**, которая служит для вывода на экран еще 9 панелей: **Калькулятор** – для вставки в формулы цифр и основных математических операторов; **График** – для построения графиков; **Матрица** – для работы с матрицами; **Вычисления** – для вставки операторов управления вычислениями; **Матанализ** – для использования операторов дифференцирования, интегрирования, пределов, сумм и произведений; **Логический (Булево)** – для вставки логических операторов; **Программирование** – для программирования средствами Mathcad; **Грек** – для вставки греческих символов; **Символы** – для вставки ключевых слов и операторов символьных вычислений.

Вывод на экран и отключение панелей производится при помощи пункта меню **Просмотр/Панели инструментов**. Многие операторы, имеющиеся на этих панелях, можно также ввести и с клавиатуры. Комбинации клавиш отображаются во всплывающих подсказках при наведении на соответствующие кнопки панелей указателя мыши.

### 18.1. Ввод и форматирование информации

**Текст.** Для работы с текстом надо ввести знак двойной кавычки или выбрать пункт меню **Вставка/Область текста**, появится прямоугольник с курсором ввода в виде красной вертикальной черты. При вводе русскоязычного текста надо использовать шрифт, заканчивающийся словом «Суг». Для перехода на новую строку служит клавиша **<Enter>**. Завершается ввод щелчком мыши вне рамки.

Форматирование текста возможно тогда, когда курсор ввода находится в рамке текстового поля. Для установки параметров форматирования можно использовать панель инструментов или команду меню **Формат/Текст**.

Выделенные рамкой текстовые блоки можно переносить на другое место.

**Ввод и вычисление математических выражений.** Ввод информации можно производить с помощью кнопок на панели инструментов и с помощью клавиатуры. В качестве разделителя у десятичных чисел при вводе используется точка.

При наборе выражений курсор ввода превращается в синий уголок, указывающий направление и место ввода. Для расширения охваченной уголком области можно пользоваться клавишей <Пробел> и/или левой кнопкой мыши и клавишами перемещения курсора.

Mathcad различает регистр букв, например, переменные  $x$  и  $X$  – это две разные переменные.

Для присвоения переменной некоторого значения используется оператор присваивания ( $:=$ ), который с клавиатуры вводится двоеточием.

При перемножении двух переменных знак умножения опускать нельзя.

Если необходимо записать способ изменения переменной, например:  $x$  меняется от 1 до 2 с шагом 0.1, то это будет выглядеть следующим образом:  $x := 1, 1.1..2$ . Здесь символ «..» появляется после нажатия клавиши (;) на клавиатуре или берется на соответствующей панели.

Важно, чтобы значение переменной в документе было определено *левее или выше того выражения*, в котором переменная используется.

**Встроенные функции.** Mathcad поддерживает множество встроенных функций. Например: **exp(x)** – экспонента (соответствует  $e^x$ ); **ln(x)**, **log(x)** – натуральный и десятичный логарифмы; **log(x, n)** – логарифм  $x$  по основанию  $n$ ; **sin(x)**, **cos(x)**, **tan(x)**, **cot(x)**, **sec(x)**, **csc(x)** – тригонометрические функции: соответственно синус, косинус, тангенс, котангенс, секанс и косеканс; **asin(x)**, **acos(x)**, **atan(x)**, **acot(x)**, **asec(x)**, **acsc(x)** – обратные тригонометрические функции.

Для организации разветвлений используется встроенная функция **if(cond, x, y)**. Если условие **cond** выполняется, то функция принимает значение  $x$ , в противном случае – значение  $y$ .

Чтобы вставить встроенную функцию, надо выбрать кнопку **f(x)** на панели инструментов или выполнить команду **Вставка/Функция** и в открывшемся диалоговом окне выбрать необходимую категорию, а затем функцию. Щелкнуть по кнопке **Вставить**. После этого в окне редактирования появится шаблон функции, в который надо ввести аргументы. Функция может быть также набрана и непосредственно с клавиатуры.

**Создание функций пользователя.** Для того чтобы определить функцию пользователя, необходимо ввести имя функции и в скобках через запятую указать имена переменных. Например:  $f(x, y) := x^2 + y^2$ .

После определения функции ее можно использовать точно так же, как и встроенные функции.

**Форматирование математических выражений и числовых результатов.** Форматирование числовых результатов вычислений возможно при выполнении команды меню **Формат/Результат**. В открывшемся диалоговом окне надо установить требуемые параметры.

Форматирование математических выражений может относиться к переменным (**Variables**) или константам (**Constants**). Выбор объекта форматирования производится с помощью команды **Формат/Равенство**, в диалоговом окне задается стиль **Variables** или **Constants**. Затем устанавливаются желаемые параметры форматирования.

При необходимости можно изменить цвет листов документов. Для этого надо выбрать пункт меню **Просмотр/Области**. Для выбора произвольного цвета заливки следует выполнить команду **Формат/Цвет/Фон**.

## 18.2. Вычисления в Mathcad

**Вычисление сумм и произведений.** Для вычисления необходимо ввести соответствующее выражение с помощью панели **Матанализ** и выбрать пункт меню **Символика/Упрощение** или ввести оператор символьного вывода ( $\rightarrow$ ) на панели **Символы**.

**Дифференцирование и интегрирование.** Следует ввести выражение, стоящее под знаком дифференциала, или подынтегральное выражение, выделить переменную дифференцирования или интегрирования и выполнить команду **Символика/Переменная/Дифференциация** или **Интеграция**.

Можно ввести нужное выражение производной или интеграла с помощью панели **Матанализ** и записать оператор символьного вывода  $\rightarrow$ .

**Матричные вычисления.** Чтобы создать матрицу, надо ввести ее имя, знак присваивания и выполнить команду **Вставка/Матрица** либо нажать кнопку **Матрица или вектор** на панели **Матрица** (либо «горячие» клавиши –  $\langle \text{Ctrl} \rangle + \langle \text{M} \rangle$ ).

В появившемся окне задать количество строк и столбцов, затем ввести конкретные значения элементов матрицы.

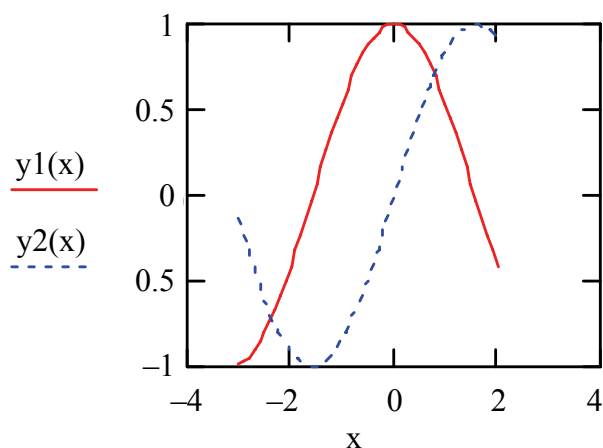
Нумерация элементов массива начинается с нуля. Mathcad допускает обращение к отдельным элементам матриц и векторов с помощью нижних индексов и к столбцам матриц с помощью верхних индексов. Нижние индексы вводятся с помощью кнопки **X<sub>n</sub>** на панели **Матрица** или клавишей открывающей квадратной скобки (**[**). Нижние индексы, если их несколько, отделяются друг от друга запятой. Верхние индексы вводятся кнопкой **M<sup>◇</sup>** на панели **Матрица** или комбинацией клавиш **<Ctrl> + <6>**.

### 18.3. Построение графиков

Mathcad позволяет строить двумерные (**X<sub>Y</sub>** график в декартовой системе координат, полярные графики в полярной системе координат) и трехмерные графики (график трехмерной поверхности, график линий уровня, трехмерная гистограмма, трехмерное множество точек, векторное поле).

**Двумерные графики.** На одном графике можно построить до 16 различных зависимостей.

Сначала нужно определить значения аргумента (или нескольких аргументов) и вид функции (или нескольких функций). Затем поместить курсор в то место, куда требуется вставить график, и при помощи меню **Вставка/График** или панели **График** выбрать **X-<sub>Y</sub> график** или **Полярный график**. Можно использовать также клавишу **@**.



Графики в приложении Mathcad

В появившейся пустой области графика в местозаполнители возле осей надо ввести имена аргумента и функции (для нескольких – через запятую). Например, на рисунке представлены два графика для значений **x**, меняющихся от **-3** до **2** с шагом **0.1**:

$x := -3, -2.9..2$

$y1(x) := \cos(x)$      $y2(x) := \sin(x)$

На вкладке **Метки** можно задать название графика, его расположение и отображение, названия и отображение меток осей.

На вкладке **Умолчание** можно указать использование формата графика как образца формата графиков по умолчанию для данного документа.

#### 18.4. Задание для выполнения на компьютере

1. Ввести текстовый заголовок к работе.

2. Произвести символьные вычисления, упростить сложные результаты:

а)  $\sum_{i=1}^{\infty} \frac{1}{i^3}$ ;                      б)  $\frac{d}{dx} \left( \frac{x^3}{3} + a^2x + a^2 \operatorname{arctg} \left( \frac{x}{a} \right) \right)$ ;

в)  $\int \frac{x+5}{\sqrt{x^2-x-1}} dx$ ;              г)  $\lim_{x \rightarrow \infty} \left( \frac{x^2+1}{x^2} \right)^{x^2+1}$ .

3. Записать пользовательскую функцию и вычислить ее значения при  $x = -5, x = 0.3, x = 4$ :

$$f(x) = \begin{cases} -\pi/2 & \text{при } x \leq -1, \\ \operatorname{arctg} \frac{x}{\sqrt{1-x^2}} & \text{при } -1 < x < 1, \\ \pi/2 & \text{при } x \geq 1. \end{cases}$$

4. Пусть заданы матрицы:

$$A = \begin{pmatrix} 1.4 & 0.2 & -10 \\ -5 & 2.5 & 0.3 \\ 3.1 & 8 & -1 \end{pmatrix}; \quad B = \begin{pmatrix} -7 & 0.6 & 4 \\ 12 & 1.2 & -9 \\ 0.5 & -6 & 3.8 \end{pmatrix}; \quad C = \begin{pmatrix} 1.5 \\ 0.6 \\ 7.7 \end{pmatrix}.$$

Вычислить  $D = AC, F = A^{-1}B, A^T$ . Найти определитель матрицы  $B$ .

5. Построить графики зависимостей  $y = kx^3 + 2x^2 - 5x - 2$  в декартовой системе координат и  $r = 3 - (k+1)\sin 3\varphi$  в полярной системе координат. Значение  $k$  выбрать произвольно. Произвести форматирование полученных графиков.

6. Построить график поверхности  $F(x, y) = \cos x \cos y$ , где  $x$  и  $y$  меняются от  $-4$  до  $4$ .

## 19. ВЫЧИСЛЕНИЕ ИНТЕГРАЛОВ И РЕШЕНИЕ УРАВНЕНИЙ

### 19.1. Методы вычисления определенных интегралов

Приближенное вычисление определенного интеграла основано на геометрическом смысле интеграла и сводится к приближенному вычислению площади, ограниченной графиком подынтегральной функции  $f(x)$ , прямыми  $x = a = x_0$ ,  $x = b = x_n$  и осью  $OX$  (рис. 19.1).

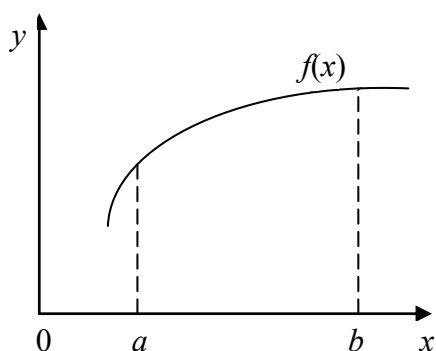


Рис. 19.1. График подынтегральной функции

Интервал  $[a, b]$  делится на  $n$  равных частей длиной  $h = (b - a) / n$ .

Тогда значениям  $x_i = x_{i-1} + h$ ,  $i = 1, 2, \dots, n$  соответствуют значения  $y_i = f(x_i)$ .

**Метод прямоугольников.** Согласно методу левых прямоугольников, искомая площадь вычисляется как сумма площадей прямоугольников, основание которых равно  $h$ , а высота равна соответственно  $y_0$  для первого прямоугольника,  $y_1$  — для второго и т. д.

вплоть до последнего с высотой  $y_{n-1}$ . Тогда

$$\int_a^b f(x) dx = hy_0 + hy_1 + \dots + hy_{n-1} = h(y_0 + y_1 + \dots + y_{n-1}).$$

Для метода правых прямоугольников аналогично

$$\int_a^b f(x) dx = hy_1 + hy_2 + \dots + hy_n = h(y_1 + y_2 + \dots + y_n).$$

**Метод трапеций.** По методу трапеций определяется сумма площадей трапеций, основаниями которых являются ординаты  $y_0, y_1$  и т. д., а высоты равны  $h$ :

$$\begin{aligned} \int_a^b f(x) dx &= h \frac{y_0 + y_1}{2} + h \frac{y_1 + y_2}{2} + \dots + h \frac{y_{n-1} + y_n}{2} = \\ &= \frac{h}{2} (y_0 + 2(y_1 + y_2 + \dots + y_{n-1}) + y_n). \end{aligned}$$

Погрешность метода оценивается как  $R_n \leq \frac{(b-a)^3 \cdot M}{12nM}$ , где  $M$  –

максимальное значение второй производной  $f(x)$  на отрезке  $[a, b]$ . Используя это соотношение, можно определить количество точек, на которое делится отрезок, исходя из заданной погрешности.

Значение интеграла, которое рассчитано по формуле трапеций, равно среднему арифметическому от значений интеграла, вычисленных по формулам левых и правых прямоугольников при том же разбиении.

Рассмотрим алгоритм метода трапеций (рис. 19.2):

1. Ввод  $a, b, n$ .
2. Вычисление  $h = (b-a)/n$ ,  $x = a + h$ ,  $s = 0$ .
3. Расчет  $s = s + f(x)$ ,  $x = x + h$ .
4. Если  $x > b - h$ , то переход к п. 5, иначе – переход к п. 3.
5. Вычисление значения интеграла

$$z = \frac{h}{2}(f(a) + 2s + f(b)).$$

6. Вывод  $z$ .

**Метод парабол (Симпсона).** Согласно методу парабол, интервал  $[a, b]$  делится на четное количество частей –  $2n$ . Тогда

$$h = \frac{b-a}{2n}, x_i = x_{i-1} + h, i = 1, 2, 3, \dots, 2n,$$

$$\int_a^b f(x)dx = \frac{h}{3}(y_0 + 4(y_1 + y_3 + \dots + y_{2n-1}) + 2(y_2 + y_4 + \dots + y_{2n-2}) + y_{2n}).$$

**Алгоритм метода парабол:**

1. Ввод  $a, b, n$ .
2. Вычисление  $h = (b-a)/2n$ ,  $x = a + 2h$ ,  $s1 = 0$ ,  $s2 = 0$ ,  $i = 1$ .
3. Расчет  $s2 = s2 + f(x)$ ,  $x = x + h$ ,  $s1 = s1 + f(x)$ ,  $x = x + h$ ,  $i = i + 1$ .
4. Если  $i < n - 1$ , то переход к п. 3, иначе – переход к следующему пункту.

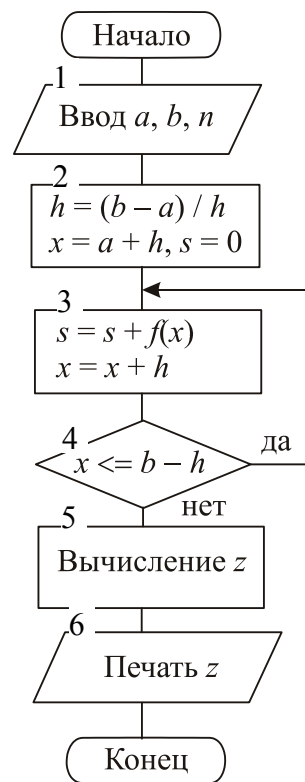


Рис. 19.2. Схема алгоритма метода трапеций

5. Вычисление значения интеграла

$$z = \frac{h}{3} (f(a) + 4f(a+h) + 4s1 + 2s2 + f(b)).$$

6. Вывод z.

Здесь  $s1 = y_3 + y_5 + \dots + y_{2n-1}$ , а  $s2 = y_2 + y_4 + \dots + y_{2n-2}$ .

## 19.2. Приближенное вычисление интеграла в приложениях Mathcad и Excel

Чтобы вычислить определенный интеграл в приложении Mathcad нужно записать интеграл, подынтегральную функцию и пределы интегрирования. Например:

$$z := \int_1^2 (\sin(x) + 3) dx$$

Для получения численного значения записывается следующее выражение: **z =**

В приложении Excel можно составить программы по алгоритмам, приведенным выше, на языке VBA. Например, пусть на рабочем листе Excel в ячейках **A1**, **A2**, **A3** располагаются значения **a**, **b** и **n**. Для вычисления интеграла по методу левых прямоугольников можно создать кнопку и составить для нее следующую программу:

```
Sub CommandButton1_Click()  
a = Range("A1")  
b = Range("A2")  
n = Range("A3")  
h = (b - a) / n  
For x = a To b - h Step h  
S = S + Fp(x)  
Next  
z = S * h  
Range("A4") = z  
End Sub  
  
Function Fp(x)  
Fp = Sin(x) + 3  
End Function
```



### 19.3. Численные методы решения уравнений

Пусть имеется уравнение  $f(x) = 0$ .

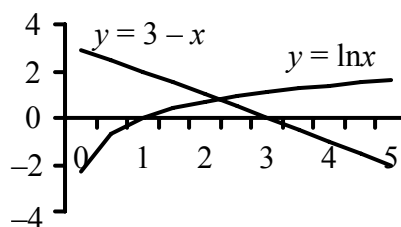
Решение уравнения численными методами состоит из двух этапов:

- отделение корней, т. е. нахождение таких отрезков  $[a, b]$  на оси  $OX$ , внутри которых имеется один корень;
- вычисление корней с заданной точностью.

Одним из способов отделения корней является графический способ. Рассмотрим его на примере.

Пусть требуется отделить корни уравнения  $3 - x - \ln x = 0$ .

Перепишем исходное уравнение в виде  $3 - x = \ln x$  и построим графики функций  $y = 3 - x$  и  $y = \ln x$  (рис. 19.3). Из чертежа видно, что графики пересекаются в единственной точке, абсцисса которой находится внутри отрезка  $[1, 3]$ . Знаки функции на концах отрезка разные:  $f(1) = 3 - 1 - \ln(1) > 0$ ,  $f(3) = 3 - 3 - \ln(3) < 0$ . Значит, данное уравнение имеет один действительный корень, лежащий внутри отрезка  $[1, 3]$ , т. е.  $a = 1$ ,  $b = 3$ .



Можно также отделить корни, построив график функции  $f(x)$  в приложениях Mathcad или Excel.

Рис. 19.3. Графики функций

После того, как определен отрезок (или отрезки), внутри которого имеется один корень, можно рассчитать его с заданной точностью одним из методов.

**Метод касательных.** При использовании данного метода для вычисления корня уравнения необходимо определить начальное приближение корня  $x_0$ :  $x_0 = a$ , если знаки  $f(a)$  и  $f'(a)$  совпадают, и  $x_0 = b$ , если знаки  $f(b)$  и  $f'(b)$  совпадают. Последовательные приближения корня рассчитываются по формуле

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, 2, \dots$$

Вычисления продолжаются до тех пор, пока не будет выполнено условие  $|x_{n+1} - x_n| \leq e$ , где  $e$  – требуемая точность определения корня.

Рассмотрим алгоритм метода касательных:

1. Ввод значений  $a$ ,  $b$ ,  $e$ .
2. Вычисление начального приближения корня  $x_{n1} = a$ , если  $f(a)f'(a) > 0$  или  $x_{n1} = b$  в противном случае.
3. Расчет  $x_n = x_{n1}$ .

4. Определение очередного приближения корня по формуле

$$x_{n1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

5. Если  $|x_{n1} - x_n| > e$ , то переход к п. 3, в противном случае – переход к п. 6.

6. Вывод значения корня  $x_{n1}$ .

**Метод дихотомии (деления отрезка пополам).** При использовании метода дихотомии отрезок  $[a, b]$  делится пополам. Из полученных двух отрезков для дальнейших вычислений выбирается тот, на концах которого функция  $f(x)$  имеет разные знаки. Выбранный отрезок вновь делится пополам. Вычисления продолжаются до тех пор, пока величина последнего из полученных отрезков не станет меньше  $2e$ .

Рассмотрим *алгоритм метода дихотомии*:

1. Ввод значений  $a, b, e$ .

2. Вычисление  $x = (a + b)/2$ .

3. Если  $f(x) = 0$ , то переход к п. 6, иначе – переход к п. 4.

4. Если  $f(x)f(a) \leq 0$ , то  $b = x$ , в противном случае –  $a = x$ .

5. Если  $|a - b| > 2e$ , то переход к п. 2, иначе – переход к следующему пункту.

6. Вывод значения корня  $x$ .

#### 19.4. Решение уравнений в приложениях Mathcad и Excel

Решить уравнение с помощью Mathcad можно разными способами.

**С помощью меню.** Следует записать уравнение, в котором логический знак равенства (на экране он будет жирным) вводится с панели **Логический**. Если уравнение приведено к виду  $f(x) = 0$ , то можно ввести только левую часть уравнения без знака равенства и нуля.

Затем надо выделить в уравнении переменную, относительно которой оно решается, и выполнить **Символика/Переменная/Решение**.

**С помощью ключевого слова solve.** Надо ввести уравнение и ключевое слово **solve** с панели **Символика**, в появившемся место-заполнителе записать имя переменной, относительно которой решается уравнение.

Для упрощения сложного решения можно после имени введенной переменной ввести ключевое слово **simplify** с панели **Символика**. Ключевые слова при этом отобразятся записанными в столбик.

**С помощью встроенной функции root.** Следует задать начальное приближение корня и записать само уравнение:

$$x := 1$$

$$f(x) := 3 - x - \ln(x)$$

Для получения значения корня нужно использовать встроенную функцию:

$$\text{root}(f(x), x) =$$

В приложении Excel можно составить программы по алгоритмам, приведенным выше, на языке VBA и произвести нужные вычисления.

Кроме того, для решения уравнения в приложении Excel имеется команда **Сервис/Подбор параметра**. Чтобы решить уравнение, надо на рабочем листе, например, в ячейке **A1** записать начальное приближение корня, в ячейке **B1** – само уравнение:

$$=3-A1-\text{Log}(A1)$$

Далее следует выполнить команду **Сервис/Подбор параметра**. В появившемся окне задать следующие значения: в поле **Установить в ячейке** выбрать **B1**, в поле **Значение** ввести **0**, в поле **Изменяя значение ячейки** – **A1**. После нажатия **<ОК>** в ячейке **A1** будет выведен корень уравнения.

## 19.5. Задание для выполнения на компьютере

1. Написать программу для вычисления определенного интеграла из таблицы на языке VBA. Номер варианта определяет преподаватель.

Исходные данные для расчета

Номер варианта	Функция $f(x)$	Пределы интегрирования	Номер варианта	Функция $f(x)$	Пределы интегрирования
1	$x^3 + x - 3$	$a = 1, b = 2$	9	$x^3 + 3x - 1$	$a = 4, b = 8$
2	$\ln x + x - 3$	$a = 3, b = 4$	10	$x^3 + x - 4$	$a = 3, b = 4$
3	$x^3 + 2x - 1$	$a = 6, b = 7$	11	$\ln x + x^3$	$a = 3, b = 7$
4	$2\ln x - 1/x$	$a = 8, b = 9$	12	$e^x - 2x^2 - 1$	$a = 2, b = 9$
5	$2 - x^2 + x$	$a = 1, b = 3$	13	$2x + \ln x - 7$	$a = 2, b = 4$
6	$5x - 1 + x^3$	$a = 2, b = 5$	14	$x^3 + 2x - 4$	$a = 1, b = 5$
7	$1 + e^x + x$	$a = 3, b = 8$	15	$2 + x + \ln x$	$a = 3, b = 7$
8	$x^3 + x - 2$	$a = 6, b = 9$	16	$x^2 + 4x - 2$	$a = 6, b = 8$

Для нечетных по номеру вариантов использовать метод трапеций, для четных – метод парабол. Для всех вариантов принять  $n = 20$ .

2. Выполнить вычисления в приложении Mathcad. Результаты сравнить между собой.

3. Отделить корни уравнения  $f(x) = 0$ , где  $f(x)$  берется из той же таблицы.

4. Написать программу вычисления корня. Для нечетных по номеру вариантов использовать метод касательных, для четных – метод дихотомии. Точность вычислений принять равной  $e = 0.0001$  для всех вариантов.

5. Выполнить вычисления в приложении Excel с помощью команды **Подбор параметра** и в приложении Mathcad. Результаты сравнить между собой.

## 20. РЕШЕНИЕ СИСТЕМ УРАВНЕНИЙ

### 20.1. Системы линейных уравнений

Система линейных уравнений  $n$ -го порядка имеет следующий вид:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases}$$

или в матричном виде:

$$AX = B,$$

где

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}, \quad X = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}, \quad B = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{bmatrix}.$$

Корнями системы являются такие значения  $x_1, x_2, \dots, x_n$ , подстановка которых в систему превращает уравнения в тождества.

**Метод Гаусса.** Метод Гаусса состоит в последовательном исключении неизвестных  $x_1, x_2, \dots, x_n$  путем преобразования системы уравнений таким образом, чтобы под главной диагональю располагались нули. В полученной системе определяется корень  $x_n$  из последнего уравнения, корень  $x_{n-1}$  – из предпоследнего и т. д.

Рассмотрим *алгоритм метода Гаусса*:

1. Ввод числа  $n$ , обозначающего порядок системы, матрицы  $A$  и вектора  $B$ .

2. Выполнение п. 3–7 данного алгоритма с изменением номера вычитаемого уравнения  $k$  с 1 до  $n - 1$ .

3. Выполнение п. 4–7 с изменением номера уравнения  $i$ , из которого производится вычитание, с  $k + 1$  до  $n$ .

4. Вычисление  $c = a_{ik} / a_{kk}$ ,  $a_{ik} = 0$ .

5. Выполнение п. 6 с изменением номера столбца  $j$  с  $k + 1$  до  $n$ .

6. Расчет  $a_{ij} = a_{ij} - ca_{kj}$ .

7. Вычисление  $b_i = b_i - cb_k$ .

8. Определение корня  $x_n = b_n / a_{nn}$ .

9. Выполнение п. 10–13 с изменением номера уравнения  $i$  с  $n - 1$  до 1.

10. Подготовка переменной для вычисления суммы  $s = 0$ .
11. Выполнение п. 12 с изменением номера столбца  $j$  с  $i + 1$  до  $n$ .
12. Вычисление  $s = s + a_j x_j$ .
13. Определение  $x_i = (b_i - s) / a_{ii}$ .
14. Вывод значений  $x_1, x_2, \dots, x_n$ .

В данном алгоритме п. 2–7 обеспечивают преобразование матрицы  $A$  к треугольному виду (прямой ход метода), а выполнение п. 8–13 позволяет определить корни системы линейных уравнений (обратный ход метода).

**Матричный метод.** Зная матрицу  $A$ , можно вычислить обратную матрицу  $A^{-1}$ , затем умножить ее на систему:  $A^{-1}AX = A^{-1}B$ . Получится:  $X = A^{-1}B$ . Элементы вектора  $X$  и являются корнями системы линейных уравнений.

## 20.2. Решение систем линейных уравнений в приложениях Mathcad и Excel

Рассмотрим решение систем линейных уравнений в приложении Mathcad матричным методом. Сначала записываются коэффициенты системы в матрицу  $A$ . Далее задается вектор  $B$  и записывается формула для определения корней

$$X := A^{-1} * B$$

Корни вычисляются после набора выражения:  $X =$

В приложении Excel также можно использовать матричный метод. Пусть имеется система линейных уравнений третьего порядка. Первоначально необходимо ввести элементы матрицы  $A$ , например, в ячейки **A1:C3**. Затем – вектор  $B$ , например, в ячейки **E1:E3**.

Далее следует выделить диапазон ячеек для вычисления корней, например **G1:G3**, и в строке формул набрать:

$$=МУМНОЖ(МОБР(A1:C3);E1:E3)$$

После ее набора нажать не одну клавишу ввода, а вместе три клавиши: **<Shift> + <Ctrl> + <Enter>**. В ячейках **G1:G3** появятся вычисленные корни системы линейных уравнений.

## 20.3. Решение систем нелинейных уравнений в приложении Mathcad

Системы нелинейных уравнений могут иметь разнообразный вид. Рассмотрим способ решения системы нелинейных уравнений на примере. Пусть имеется система:

$$\begin{cases} 5x_1x_2 + 0.2x_2 = -6, \\ -6x_1 + 4x_1x_2 = 0.8. \end{cases}$$

В приложении Mathcad надо записать начальные приближения корней и систему уравнений в блоке **given**:

```
x1 := 1   x2 := 1
given
5 * x1 * x2 + 0.2 * x2 = -6
-6 * x1 + 4 * x1 * x2 = 0.8
```

При записи системы используется не знак равенства, а знак логического равенства (=), который имеется на панели **Булево**. Затем вводится встроенная функция: **r := find(x1, x2)**.

Чтобы получить значения корней, надо записать: **r =**

## 20.4. Задание для выполнения на компьютере

1. Составить программу решения системы линейных уравнений по методу Гаусса. Коэффициенты системы даны в таблице.

2. Решить эту же систему с помощью приложений Mathcad и Excel. Результаты сравнить между собой.

3. Решить систему нелинейных уравнений, приведенную выше, в приложении Mathcad.

**Исходные данные для расчета**

№ п/п	$a_{11}$	$a_{12}$	$a_{13}$	$a_{21}$	$a_{22}$	$a_{23}$	$a_{31}$	$a_{32}$	$a_{33}$	$b_1$	$b_2$	$b_3$
1	6	-3	1	4	8	-2	7	-1	9	1.2	1.3	1.4
2	5	-2	1	1	-4	-1	2	-1	4	1.7	1.6	1.9
3	4	1	-1	1	-5	1	2	-3	6	6.2	2.4	1.8
4	8	2	-1	1	10	5	2	-1	-4	2.3	1.4	1.7
5	5	0.3	0.1	0.2	2	0.8	0.4	0.4	2	7.2	8.3	6.6
6	3	0.6	-1	0.8	3	1.2	1	0.8	2.6	0.2	-3	2.7
7	1	0.1	0.1	0.2	1	0.3	0.3	0.4	1	1.5	1.6	1.7
8	4	1.2	1	1.5	2.8	0.7	1	1.5	3.5	7.1	7.7	4.2
9	2.1	-2	0.3	0.7	1.9	0.6	0.3	0.7	1.5	1.6	0.9	3.5
10	3.1	-2	0.5	0.8	2.7	-1	0.5	-1	2.1	1.3	4	2.3
11	0.1	0.3	0.4	0.2	0.1	0.6	0.4	0.2	0.3	0.7	0.5	0.6
12	4	-1	2	1	-5	3	2	1	-8	1.8	2.5	0.3
13	7.1	-2	0.7	0.7	3.5	8.4	5.5	5.6	1.3	7.7	0.1	4.5
14	2.5	0.2	0.1	0.1	1	0.4	0.2	0.2	1	3.6	4.2	3.3
15	1.6	0.7	0.2	0.1	-2	0.5	1	0.2	2.5	1.9	2.6	2.3

## 21. МАТЕМАТИЧЕСКИЕ МОДЕЛИ И ОПТИМИЗАЦИЯ ПРОЦЕССОВ

### 21.1. Модель одномерного объекта

Пусть в результате проведения эксперимента получена табличная зависимость значений выходного параметра процесса  $y$  от значений входного параметра  $x$  (рис. 21.1).

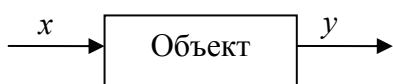


Рис. 21.1. Одномерный объект

Требуется получить эмпирическую формулу, описывающую зависимость  $y$  от  $x$ . Решение такой задачи состоит из двух этапов.

На первом этапе выбирается общий вид формулы, исходя из теоретических представлений о характере изучаемого процесса. Это может быть, например, полином  $m$ -степени:

$$y = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \dots + a_m \cdot x^m.$$

Формула может содержать тригонометрические, экспоненциальные, логарифмические функции и т. п.

На втором этапе определяются значения параметров  $a_0, a_1, \dots, a_m$  эмпирической формулы  $f(x, a_0, a_1, \dots, a_m)$ , которые обеспечивали бы соответствие этой формулы экспериментальным данным.

Согласно методу наименьших квадратов, параметры  $a_0, a_1, \dots, a_m$  выбираются так, чтобы была минимальной сумма квадратов:

$$S = \sum_{i=1}^n \delta_i^2 = \sum_{i=1}^n (y_i - f(x_i, a_0, a_1, \dots, a_m))^2.$$

Чтобы найти нужные параметры, следует взять частные производные от правой части по  $a_0, a_1, \dots, a_m$  и приравнять их к нулю. Полученную систему уравнений можно решить одним из известных методов.

*Пример.* Пусть требуется определить параметры  $a_0, a_1, a_2$  полинома второй степени:

$$y = a_0 + a_1 \cdot x + a_2 \cdot x^2.$$

Надо взять частные производные от выражения

$$S = \sum_{i=1}^n (y_i - (a_0 + a_1 \cdot x + a_2 \cdot x^2))^2$$

и приравнять их к нулю:



$$\begin{cases} \frac{\partial S}{\partial a_0} = 2 \cdot \sum_{i=1}^n (y_i - (a_0 + a_1 \cdot x_i + a_2 \cdot x_i^2)) \cdot (-1) = 0, \\ \frac{\partial S}{\partial a_1} = 2 \cdot \sum_{i=1}^n (y_i - (a_0 + a_1 \cdot x_i + a_2 \cdot x_i^2)) \cdot (-x_i) = 0, \\ \frac{\partial S}{\partial a_2} = 2 \cdot \sum_{i=1}^n (y_i - (a_0 + a_1 \cdot x_i + a_2 \cdot x_i^2)) \cdot (-x_i^2) = 0. \end{cases}$$

Отсюда

$$\begin{cases} a_0 \cdot n + a_1 \cdot \sum_{i=1}^n x_i + a_2 \cdot \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i, \\ a_0 \cdot \sum_{i=1}^n x_i + a_1 \cdot \sum_{i=1}^n x_i^2 + a_2 \cdot \sum_{i=1}^n x_i^3 = \sum_{i=1}^n y_i \cdot x_i, \\ a_0 \cdot \sum_{i=1}^n x_i^2 + a_1 \cdot \sum_{i=1}^n x_i^3 + a_2 \cdot \sum_{i=1}^n x_i^4 = \sum_{i=1}^n y_i \cdot x_i^2. \end{cases}$$

Решив эту систему линейных уравнений, можно определить искомые величины  $a_0$ ,  $a_1$ ,  $a_2$ .

Рассмотрим *алгоритм метода наименьших квадратов* для вычисления коэффициентов полинома второй степени:

1. Ввод количества опытов  $n$ , значений  $x_1, x_2, \dots, x_n; y_1, y_2, \dots, y_n$ .
2. Определение коэффициентов системы линейных уравнений:

$$a_{1,2} = \sum_{i=1}^n x_i, \quad a_{1,3} = \sum_{i=1}^n x_i^2, \quad a_{2,3} = \sum_{i=1}^n x_i^3, \quad a_{3,3} = \sum_{i=1}^n x_i^4;$$

$$b_1 = \sum_{i=1}^n y_i, \quad b_2 = \sum_{i=1}^n y_i x_i, \quad b_3 = \sum_{i=1}^n y_i x_i^2;$$

$$a_{1,1} = n, \quad a_{2,1} = a_{1,2}, \quad a_{2,2} = a_{1,3}, \quad a_{3,1} = a_{1,3}, \quad a_{3,2} = a_{2,3}.$$

3. Решение системы  $AZ = B$ , где  $A$  – матрица коэффициентов;  $Z$  – вектор, в котором определяются корни  $z_1 = a_0$ ,  $z_2 = a_1$ ,  $z_3 = a_2$ ;  $B$  – вектор свободных членов системы.

4. Вывод искомых коэффициентов  $a_0$ ,  $a_1$ ,  $a_2$ .

5. Определение и вывод разностей  $\delta_1, \delta_2, \dots, \delta_n$ .

## 21.2. Получение математических моделей в приложениях Mathcad и Excel

В приложении Mathcad можно записать формулы приведенного выше алгоритма и вычислить коэффициенты функциональной зависимости.

Существуют также и встроенные функции для определения математических моделей. Например, пусть имеются значения  $x$  и  $y$ , полученные в результате проведения опытов. Надо найти математическую модель в виде полинома второй степени:

$$y = a_0 + a_1 \cdot x + a_2 \cdot x^2.$$

Можно использовать для решения задачи встроенную функцию **linfit**. На листе Mathcad тогда нужно записать:

$$X := \begin{bmatrix} 0.1 \\ 0.3 \\ 0.5 \\ 0.7 \\ 0.9 \\ 1 \end{bmatrix} \quad Y := \begin{bmatrix} 5.1 \\ 9.8 \\ 15.5 \\ 8.4 \\ 6.1 \\ 4.9 \end{bmatrix} \quad F(X) := \begin{bmatrix} 1 \\ x \\ x^2 \end{bmatrix}$$

$$A := \text{linfit}(X, Y, F)$$

Вычисленные значения  $a_0$ ,  $a_1$ ,  $a_2$  будут записаны в векторе  $A$ , который появится после ввода текста: **A =**

Для построения графика теперь можно определить значения:

$$i := 0..5 \quad t := 0, 0.01..1 \quad Z(t) := F(t) * A$$

Здесь  $Z(t)$  – искомая математическая модель. Если построить на одном графике зависимость  $Z(t)$  от  $t$  и зависимость  $Y_i$  от  $X_i$ , то можно сравнить, насколько хорошо полученный полином описывает данные опытов.

В приложении Excel предоставляется интересная возможность получения математических моделей через построение графиков функций.

Пусть имеются значения  $x_1, x_2, \dots, x_n$  и соответствующие им значения  $y_1, y_2, \dots, y_n$ . Надо для этих данных построить точечный график и выполнить команду **Диаграмма/Добавить линию тренда**. В появившемся окне на вкладке **Тип** определить вид математической модели, а на вкладке **Параметры** отметить флажок **Показывать уравнение на диаграмме**. После нажатия **<ОК>** искомое уравнение появится на графике.

### 21.3. Модель многомерного объекта

Предположим, что технологический процесс можно описать математической моделью вида

$$y = b_0 + b_1x_1 + \dots + b_nx_n + b_{12}x_1x_2 + \dots + b_{n-1,n}x_{n-1}x_n,$$

где  $y$  – выходной параметр процесса;  $b_0, b_1, \dots, b_{n-1,n}$  – искомые неизвестные коэффициенты процесса;  $x_1, \dots, x_n$  – входные параметры процесса.

Соотношения такого вида называются уравнениями регрессии.

Например, для процесса (рис. 21.2), имеющего три входных параметра (фактора), математическая модель примет вид

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_{12}x_1x_2 + b_{13}x_1x_3 + b_{23}x_2x_3.$$

Чтобы определить коэффициенты математической модели процесса, необходимо провести эксперимент по соответствующему плану, например по плану полного факторного эксперимента.

Количество опытов в эксперименте рассчитывается по формуле  $N = 2^n$ , где  $n$  – количество факторов. Входные воздействия принимают минимальные и максимальные значения. Для упрощения вычислений нужно перейти от физических переменных  $x_1, \dots, x_n$  к кодированным по следующей формуле:



Рис. 21.2. Многомерный объект

Для упрощения вычислений нужно перейти от физических переменных  $x_1, \dots, x_n$  к кодированным по следующей формуле:

$$z_i = \frac{x_i - x_{i0}}{\Delta x_i}, \quad i = 1, 2, 3.$$

Здесь  $x_{i0}$  – значение фактора на базовом (нулевом) уровне, равное среднему значению между минимальным и максимальным значениями;  $\Delta x_i$  – интервал варьирования по данному фактору.

В случае трех входных параметров план проведения эксперимента имеет вид, представленный на рис. 21.3.

№ п/п	z1	z2	z3	yэ
1	-1	-1	-1	yэ1
2	+1	-1	-1	yэ2
3	-1	+1	-1	yэ3
4	+1	+1	-1	yэ4
5	-1	-1	+1	yэ5
6	+1	-1	+1	yэ6
7	-1	+1	+1	yэ7
8	+1	+1	+1	yэ8

Рис. 21.3. План эксперимента

Значение  $-1$  соответствует минимальному значению входного параметра,  $+1$  – максимальному значению входного параметра.

В соответствии с методом наименьших квадратов производится вычисление коэффициентов:

$$b_0 = \frac{1}{N} \sum_{j=1}^N y_{эj}, \quad b_1 = \frac{1}{N} \sum_{j=1}^N z1_j \cdot y_{эj}, \quad b_2 = \frac{1}{N} \sum_{j=1}^N z2_j \cdot y_{эj}, \quad b_3 = \frac{1}{N} \sum_{j=1}^N z3_j \cdot y_{эj},$$

$$b_{12} = \frac{1}{N} \sum_{j=1}^N z1_j \cdot z2_j \cdot y_{эj}, \quad b_{13} = \frac{1}{N} \sum_{j=1}^N z1_j \cdot z3_j \cdot y_{эj},$$

$$b_{23} = \frac{1}{N} \sum_{j=1}^N z2_j \cdot z3_j \cdot y_{эj}, \quad i = 1, 2, \dots, N.$$

Коэффициент регрессии  $b$  ( $b_0, b_1, \dots, b_{n-1,n}$ ) считается значимым, если выполняется условие

$$\frac{|b|}{Sb} \geq t_T, \quad Sb^2 = \frac{Sy^2}{N},$$

где  $Sb$  – среднеквадратичная ошибка в определении коэффициентов регрессии;  $t_T$  – табличное значение критерия Стьюдента, которое выбирается для числа степеней свободы  $f1 = m - 1$ .

Для расчета дисперсии воспроизводимости нужно выполнить дополнительно  $m$  опытов ( $m < N$ ) по любой строчке плана, например, при значениях входных факторов на базовом уровне.

В результате получают дополнительные значения экспериментальных данных  $yd_1, yd_2, \dots, yd_m$ .

Тогда

$$Sy^2 = \sum_{k=1}^m \frac{(yd_k - yoc)^2}{m-1}, \quad yoc = \frac{yd_1 + yd_2 + \dots + yd_k}{m}, \quad k = 1, \dots, m.$$

В табл. 21.1 приведены значения критерия Стьюдента.

Таблица 21.1

**Значения критерия Стьюдента**

Число степеней свободы $f1$	1	2	3	4	5	6	7
Значение коэффициента	12.71	4.30	3.18	2.78	2.57	2.45	2.36

Если коэффициент не удовлетворяет критерию Стьюдента, то он считается незначимым и приравнивается к нулю.

Проверка адекватности (соответствия) полученного уравнения регрессии экспериментальным данным проводится с помощью критерия Фишера. Для этого вычисляются

$$S\alpha\partial^2 = \frac{1}{N-B} \sum_{j=1}^N (y_{\text{э}j} - y_{\text{р}j})^2, \quad F = \frac{S\alpha\partial^2}{S_y^2},$$

где  $S\alpha\partial^2$  – оценка дисперсии адекватности;  $B$  – число значимых коэффициентов уравнения регрессии;  $y_{\text{э}j}$ ,  $y_{\text{р}j}$  – соответственно экспериментальное и рассчитанное по найденной математической модели значения  $y$  в  $j$ -м опыте.

Определяется также табличное значение критерия Фишера  $F_T$  по числу степеней свободы  $f_1$  и числу степеней свободы  $f_2 = N - B$  (табл. 21.2).

Если  $F < F_T$ , то уравнение регрессии рассматривается как модель исследуемого процесса.

Таблица 21.2

**Коэффициенты критерия Фишера**

Число степеней свободы $f_1$	Число степеней свободы $f_2$					
	1	2	3	4	5	6
1	161.40	199.50	215.70	224.60	320.20	234.00
2	18.51	19.00	19.16	19.25	19.30	19.33
3	10.13	9.55	9.28	9.12	9.01	8.94
4	7.71	6.94	6.59	6.39	6.26	6.16
5	6.61	5.79	5.41	5.19	5.05	4.95
6	5.99	5.14	4.76	4.53	3.39	4.28
7	5.99	4.74	4.35	4.12	3.97	3.97

Если полученное уравнение не адекватно процессу, то нужно перейти к более сложному виду математической модели, вновь провести опыты и обработать их результаты.

Если уравнение адекватно процессу, то нужно от кодированных переменных перейти к физическим.

#### 21.4. Задание для выполнения на компьютере

1. В табл. 21.3 приведены значения входных и выходных параметров некоторого процесса. В качестве эмпирической формулы выбрать полином второй степени и составить программу получения его коэффициентов. Номер варианта определяет преподаватель.

Таблица 21.3

## Исходные данные для расчета

№ п/п	Переменные	Значения переменных								
1	x	2.1	2.7	3.3	3.8	4.2	4.9	5.6	6.1	6.8
	y	1.2	1.6	2.1	2.4	2.5	2.8	3.4	3.8	4.0
2	x	0.2	0.7	1.1	1.6	2.2	2.3	3.0	3.9	4.3
	y	6.3	10.6	14.2	15.7	15.9	15.5	12.5	5.0	0.2
3	x	-5.0	-4.2	-3.5	-2.8	-1.9	-1.2	-0.3	0.8	1.3
	y	8.8	4.3	1.8	-0.2	-0.8	-0.5	1.8	7.6	12.2
4	x	0.0	0.6	1.3	1.8	2.7	3.1	3.9	4.2	5.1
	y	10.2	8.2	6.0	5.1	1.5	0.8	-1.6	-2.8	-5.5
5	x	-0.4	-3.5	-2.4	-2.0	-0.8	0.5	1.4	2.5	3.8
	y	-1.6	-1.4	-1.1	-0.9	-0.7	-0.5	-0.4	-0.2	0.1
6	x	3.1	3.7	4.3	4.8	5.2	5.9	6.6	7.1	7.8
	y	1.2	1.6	2.1	2.4	2.5	2.8	3.4	3.8	4.0
7	x	0.2	0.7	1.1	1.6	2.2	2.3	3.0	3.9	4.7
	y	6.3	9.6	13.2	14.7	14.9	14.5	11.5	4.0	2.1
8	x	-3.0	-2.2	-1.5	-1.1	-0.9	-0.2	0.3	0.8	1.3
	y	8.8	4.3	1.8	-0.2	-0.8	-0.5	1.8	7.6	12.2
9	x	0.0	0.6	1.3	1.8	2.7	3.1	3.9	4.2	5.1
	y	12.2	9.2	8.0	7.1	0.5	0.8	-2.6	-3.8	-6.5
10	x	-5.4	-3.5	-2.4	-2.0	-0.8	0.5	1.4	2.5	3.4
	y	-1.6	-1.1	-0.8	-0.7	0.7	1.4	2.1	3.2	3.9
11	x	4.1	5.7	6.3	6.8	7.2	7.9	8.6	9.1	9.8
	y	1.2	1.6	2.1	2.4	2.5	2.8	3.4	3.8	4.0
12	x	0.2	0.7	1.1	1.6	2.2	2.3	3.0	3.9	4.3
	y	7.3	11.6	16.2	17.7	17.9	15.5	12.5	5.0	0.2
13	x	-3.0	-2.2	-1.5	-0.8	0.2	0.3	1.3	1.8	2.3
	y	8.8	4.3	1.8	-0.2	-0.8	-0.5	1.8	7.6	12.2
14	x	0.0	0.6	1.3	1.8	2.7	3.1	3.9	4.2	5.1
	y	20.2	28.2	26.0	25.1	21.5	20.8	11.6	12.8	15.5
15	x	0.4	1.5	2.4	2.8	3.1	4.5	5.4	5.5	6.8
	y	-1.6	-1.3	-1.0	-0.8	-0.7	-0.4	-0.2	-0.2	0.3

2. Определить коэффициенты математической модели процесса в виде полинома второй степени с помощью приложений Mathcad и Excel. Результаты сравнить между собой.

## 22. МЕТОДЫ ОПТИМИЗАЦИИ

Задачу оптимизации в общем виде можно сформулировать так: определить значения входных параметров  $x_1, x_2, \dots, x_n$  некоторого процесса, которые обеспечивают максимум или минимум целевой функции  $f(x_1, x_2, \dots, x_n)$ , характеризующей показатели процесса, и удовлетворяют ограничениям, если они присутствуют.

### 22.1. Метод сканирования

Рассмотрим использование метода сканирования на примере оптимизации процесса, имеющего два входных параметра  $x_1, x_2$  и выходной параметр  $y$ . Пусть требуется определить оптимальные значения  $x_1$  и  $x_2$ , которые обеспечивали бы минимум целевой функции

$$y = f(x_1, x_2)$$

и удовлетворяли ограничениям:

$$a1 \leq x_1 \leq b1,$$

$$a2 \leq x_2 \leq b2,$$

$$g(x_1, x_2) > 0$$

(последнее ограничение может отсутствовать).

Метод сканирования заключается в нахождении значений  $x_1$  из интервала  $[a1, b1]$ , начиная с  $a1$  и до  $b1$  с шагом  $h1$ , и определении значений  $x_2$  из интервала  $[a2, b2]$ , начиная с  $a2$  и до  $b2$  с шагом  $h2$ . Для всех значений  $x_1$  и  $x_2$ , удовлетворяющих ограничениям  $g(x_1, x_2) > 0$ , нужно вычислить значения целевой функции  $y = f(x_1, x_2)$ .

Те значения  $x_1$  и  $x_2$ , для которых значение целевой функции минимально, являются искомым решением.

Рассмотрим *алгоритм метода сканирования*:

1. Ввод исходных данных:  $a1, b1, h1, a2, b2, h2$  и некоторого числа  $A$ , заведомо большего, чем значение целевой функции.

2. Вычисление  $y_{opt} = A, x1_{opt} = a1, x2_{opt} = a2$ .

3.  $x_1 = a1$ .

4.  $x_2 = a2$ .

5. Проверка ограничения: если ограничение не выполняется, т. е.  $g(x_1, x_2) \leq 0$ , то переход к п. 8, иначе – переход к следующему пункту.
6. Вычисление целевой функции  $y = f(x_1, x_2)$ .
7. Если  $y < y_{opt}$ , то  $y_{opt} = y$ ,  $x1_{opt} = x_1$ ,  $x2_{opt} = x_2$ , в противном случае – переход к следующему пункту.
8. Вычисление  $x_2 = x_2 + h_2$ .
9. Если  $x_2 \leq b_2$ , то переход к п. 5, иначе – переход к следующему пункту.
10. Вычисление  $x_1 = x_1 + h_1$ .
11. Если  $x_1 \leq b_1$ , то переход к п. 4, в противном случае – переход к следующему пункту.
12. Вывод оптимальных значений  $x1_{opt}$ ,  $x2_{opt}$  и минимального значения целевой функции  $y_{opt}$ .

## 22.2. Метод случайного поиска

Рассмотрим применение метода случайного поиска для оптимизации процесса на примере, приведенном выше. Идея метода основана на многократном ( $N$  раз) вычислении целевой функции  $y$  для значений  $x_1$  и  $x_2$ , выбранных из отрезков  $[a_1, b_1]$  и  $[a_2, b_2]$  случайным образом. Те значения  $x_1$  и  $x_2$ , при которых целевая функция минимальна и удовлетворяются ограничения, и являются решением.

Для определения случайного числа  $x$  на отрезке  $[a, b]$  можно использовать встроенную функцию **Rnd**. Тогда  $x = (b - a) \cdot \text{Rnd}(1) + a$ .

*Алгоритм метода случайного поиска:*

1. Ввод исходных данных:  $a_1, b_1, a_2, b_2$ , количества опытов  $N$  и числа  $A$ , заведомо большего, чем значение целевой функции.
2. Вычисление  $y_{opt} = A$ ,  $x1_{opt} = a_1$ ,  $x2_{opt} = a_2$ .
3.  $i = 1$ .
4. Вычисление  $x_1 = (b_1 - a_1) \cdot \text{Rnd}(1) + a_1$ ,  $x_2 = (b_2 - a_2) \cdot \text{Rnd}(1) + a_2$ .
5. Проверка ограничения: если  $g(x_1, x_2) \leq 0$ , то переход к п. 8, иначе – переход к следующему пункту.
6. Вычисление целевой функции  $y = f(x_1, x_2)$ .
7. Если  $y < y_{opt}$ , то  $y_{opt} = y$ ,  $x1_{opt} = x_1$ ,  $x2_{opt} = x_2$ , в противном случае – переход к следующему пункту.
8.  $i = i + 1$ .
9. Если  $i \leq N$ , то переход к п. 4, иначе – переход к п. 10.
10. Вывод оптимальных значений  $x1_{opt}$ ,  $x2_{opt}$  и минимального значения целевой функции  $y_{opt}$ .



### 22.3. Решение задач оптимизации в приложениях Mathcad и Excel

В приложении Mathcad имеются встроенные функции, с помощью которых можно решать задачи оптимизации. Рассмотрим пример.

Пусть требуется определить оптимальные значения  $x_1$  и  $x_2$ , которые обеспечивали бы максимум целевой функции

$$y = 1.7 + 4.56x_1 - 3x_2 - 0.69x_1x_2 - 0.44x_2^2$$

и удовлетворяли ограничениям:

$$3 \leq x_1 \leq 4, \quad 0.1 \leq x_2 \leq 0.9.$$

В рабочей области приложения Mathcad требуется записать:

$$f(x_1, x_2) := 1.7 + 4.56 * x_1 - 3 * x_2 - 0.69 * x_1 * x_2 - 0.44 * x_2^2$$

$$x_1 := 3 \quad x_2 := 0.1$$

given

$$4 \geq x_1 \geq 3$$

$$0.9 \geq x_2 \geq 0.1$$

$$R := \text{maximize}(f, x_1, x_2) \quad R =$$

Встроенная функция **minimize** позволяет решать задачи оптимизации, в которых нужно определить минимум целевой функции.

В приложении Excel имеется специальная команда, с помощью которой можно решать задачи оптимизации. Например, чтобы решить предыдущий пример, нужно произвести следующие действия:

– на рабочем листе, например, в ячейке **A1** записать значение левой границы для первого ограничения (число 3);

– в ячейке **B1** записать значение левой границы для второго ограничения (число 0);

– в ячейке **C1** записать целевую функцию:

$$=1,7+4,56*A1-3*B1-0,69*A1*B1-0,44*B1^2$$

– выполнить команду **Сервис/Поиск решения**. В появившемся окне задать имя ячейки с целевой функцией (для данного примера **C1**), определить, что в задаче целевая функция стремится к максимуму, ввести соответствующие ограничения для содержимого ячеек **A1** и **B1**.

Решение задачи можно посмотреть и проанализировать на отдельном листе.

Если в пункте меню **Сервис** команда **Поиск решения** отсутствует, то ее можно добавить, используя команду **Сервис/Настройка**.

## 22.4. Задание для выполнения на компьютере

В табл. 22.1 представлены данные для оптимизации процессов, модели которых в виде уравнений регрессии приведены в табл. 22.2. Написать программу для решения задач оптимизации методом сканирования и методом случайного поиска. Шаги  $h_1$  и  $h_2$  для метода сканирования выбрать самостоятельно. Проанализировать результаты.

Таблица 22.1

### Исходные данные для оптимизации процессов

№ п/п	Критерий	$x_1$	$x_2$	$x_3$	Ограничения
1	$y \rightarrow \max$	[15; 70]	[100; 170]	[10; 20]	$x_1 + x_2 < 150$
2	$y \rightarrow \min$	[20; 50]	[80; 140]	[30; 50]	$x_2 + x_3 > 135$
3	$y \rightarrow \max$	[30; 70]	[100; 150]	[12; 20]	
4	$y \rightarrow \max$	[50; 80]	[150; 180]	[15; 30]	$x_1 + x_2 - x_3 > 200$
5	$y \rightarrow \min$	[20; 50]	[100; 170]	[20; 30]	
6	$y \rightarrow \max$	[40; 60]	[10; 30]	[2.1; 2.4]	$x_1 + x_2 < 80$
7	$y \rightarrow \min$	[30; 70]	[100; 150]	[15; 25]	$x_2 + x_3 < 155$
8	$y \rightarrow \max$	[2; 4]	[0.1; 0.8]	[40; 70]	
9	$y \rightarrow \max$	[330; 350]	[10; 20]	[15; 16]	$x_1 + x_3 > 350$
10	$y \rightarrow \min$	[20; 50]	[90; 170]	[10; 30]	
11	$y \rightarrow \max$	[40; 70]	[100; 170]	[20; 40]	$x_1 + x_2 < 220$
12	$y \rightarrow \min$	[30; 50]	[80; 140]	[15; 45]	$x_2 + x_3 > 100$
13	$y \rightarrow \max$	[20; 70]	[90; 150]	[2; 26]	
14	$y \rightarrow \max$	[50; 80]	[100; 140]	[15; 35]	$x_1 + x_2 + x_3 > 175$
15	$y \rightarrow \min$	[0.8; 1.6]	[0.3; 0.8]	[1.6; 2.5]	

Таблица 22.2

### Модели в виде уравнений регрессии

№ п/п	Математическая модель
1	$y = 48.94 + 9.29x_1 - 3.26x_2 + 8.41x_3 - 0.76x_1x_2 - 2.44x_1x_3 - 8.24x_2x_3$
2	$y = 21.93 + 4.56x_1 + 7.76x_3 - 0.69x_1x_2 - 0.44x_2x_3$
3	$y = 23.63 + 4.65x_1 - 2.15x_2 + 5.68x_3 - 1.03x_1x_2 - 5.6x_2x_3$
4	$y = 75.78 + 5.13x_1 + 7.15x_3 - 0.65x_1x_3 - 3.58x_2x_3$
5	$y = 21.77 + 2.97x_1 + 4.23x_3 - 0.51x_1x_2 - 2.1x_2x_3$
6	$y = 3.59 + 0.69x_1 - 1.36x_2 + 0.66x_3 + 0.26x_1x_3$
7	$y = 40.28 + 8.98x_1 - 3.27x_2 + 10.95x_3 - 0.68x_1x_2 - 1.71x_1x_3 - 8.65x_2x_3$
8	$y = 2100 - 665.4x_1 - 242.9x_1x_2 + 115.4x_1x_3 - 67.1x_2x_3$
9	$y = 26.36 + 0.89x_1 - 2.01x_2 + 1.89x_3 - 1.19x_1x_2 + 1.01x_2x_3$
10	$y = 48.43 + 8.98x_1 - 3.3x_2 + 8.7x_3 - 1.35x_1x_2 - 1.85x_1x_3 - 8.13x_2x_3$
11	$y = 22.73 + 4.65x_1 - 1.28x_2 + 10.08x_3 - 0.95x_1x_2 + 0.55x_1x_3 - 1.48x_2x_3$
12	$y = 34.67 + 6.18x_1 + x_2 + 10.45x_3 + 1.95x_1x_2 - 2.5x_1x_3 - 5.28x_2x_3$
13	$y = 23.8 + 4.68x_1 - 2.13x_2 + 5.68x_3 - 0.95x_1x_2 - 0.45x_1x_3 - 5.6x_2x_3$
14	$y = 18.14 + 3.69x_1 - 1.09x_2 + 5.44x_3 - 0.14x_1x_2 - 1.16x_1x_3 - 3.89x_2x_3$
15	$y = 16.79 + 0.61x_1 - 3.2x_2 + 0.41x_3 - 0.6x_1x_2 - 0.2x_1x_3 - 0.24x_2x_3$

## 23. РЕШЕНИЕ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

### 23.1. Решение уравнений первого порядка

Рассмотрим дифференциальное уравнение первого порядка:

$$\frac{dy}{dx} = f(x, y).$$

Требуется найти решение на интервале  $[x_0, x_n]$ , удовлетворяющее начальному условию  $y(x_0) = y_0$ .

Для приближенного решения дифференциального уравнения интервал  $[x_0, x_n]$  разбивается на  $n$  частей с шагом  $h$ :

$$x_{i+1} = x_i + h, \quad i = 0, 1, 2, \dots, n - 1.$$

В полученных точках вычисляются значения  $y_i$ .

**Метод Эйлера.** Согласно методу Эйлера, значения  $y_i$  определяются по формуле

$$y_{i+1} = y_i + hf(x_i, y_i).$$

*Алгоритм метода Эйлера:*

1. Ввод  $n$ , конечного значения  $x_n$ , начального значения  $x_0$  (в переменную  $x$ ), ввод  $y_0$  (в переменную  $y$ ).
2. Вычисление  $h = (x_n - x) / n$ ,  $x = x_0$ ,  $y = y_0$ .
3. Вывод  $x$ ,  $y$ .
4. Вычисление  $y = y + hf(x, y)$ ,  $x = x + h$ .
5. Если  $x > x_n$ , то переход к п. 6, иначе – переход к п. 3.
6. Конец вычислений.

Для получения достоверных результатов значение  $h$  должно быть достаточно мало, при этом можно не выводить все получающиеся значения  $x$  и  $y$ . Целесообразно внести изменения в алгоритм программы так, чтобы вычисления проводились с малым шагом, а вывод результатов – с большим.

**Метод Рунге – Кутты.** Расчетные формулы метода Рунге – Кутты четвертого порядка имеют следующий вид:

$$k_1 = hf(x_i, y_i),$$
$$k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right),$$

$$\begin{aligned}
k_3 &= hf \left( x_i + \frac{h}{2}, y_i + \frac{k_2}{2} \right), \\
k_4 &= hf(x_i + h, y_i + k_3), \\
y_{i+1} &= y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \\
x_{i+1} &= x_i + h, \quad i = 0, 1, 2, \dots, n-1.
\end{aligned}$$

Для разработки программы, реализующей метод Рунге – Кутты, можно использовать тот же алгоритм, что и для метода Эйлера, внося в него соответствующие изменения.

### 23.2. Системы дифференциальных уравнений и уравнения высших порядков

Система дифференциальных уравнений  $m$ -го порядка имеет вид

$$\begin{cases}
\frac{dy_1}{dx} = f_1(x, y_1, y_2, \dots, y_m) \\
\frac{dy_2}{dx} = f_2(x, y_1, y_2, \dots, y_m) \\
\quad \quad \quad \dots \\
\frac{dy_m}{dx} = f_m(x, y_1, y_2, \dots, y_m)
\end{cases}$$

Для решения системы на отрезке  $[x_0, x_n]$  должны быть заданы начальные условия:  $y_1(x_0) = y_1^0, y_2(x_0) = y_2^0, \dots, y_m(x_0) = y_m^0$ . Решением системы  $m$ -го порядка будут  $m$  функций, удовлетворяющих начальным условиям. Чтобы определить эти функции, можно использовать метод Эйлера или Рунге – Кутта (или любой другой метод), применяя их к каждому уравнению последовательно.

Уравнения высших порядков сводятся к системам дифференциальных уравнений путем введения новых переменных.

*Пример.* Требуется решить уравнение  $y'' + 2y' - y + 4x = 5$  на отрезке  $[1; 1.3]$ . Начальные условия:  $y(1) = 2, y'(1) = 0$ . Шаг  $h = 0.1$ . Здесь шаг выбран большим, чтобы было проще продемонстрировать вычисления, сделанные вручную.

Введем новую переменную  $z = y'$ . Тогда исходное уравнение записывается в виде системы двух уравнений первого порядка:

$$y' = z,$$

$$z' = -2z + y - 4x + 5.$$

Начальные условия:  $y(1) = 1, z(1) = 0$ . Решим данную систему методом Эйлера:

$$y(1.1) = 2 + 0.1 \cdot 0 = 2,$$

$$z(1.1) = 0 + 0.1 \cdot (-2 \cdot 0 + 2 - 4 \cdot 1 + 5) = 0.3,$$

$$x = 1 + 0.1 = 1.1,$$

$$y(1.2) = 2 + 0.1 \cdot 0.3 = 2.03,$$

$$z(1.2) = 0.3 + 0.1 \cdot (-2 \cdot 0.3 + 2 - 4 \cdot 1.1 + 5) = 0.5,$$

$$x = 1.1 + 0.1 = 1.2,$$

$$y(1.3) = 2.03 + 0.1 \cdot 0.5 = 2.08,$$

$$z(1.3) = 0.5 + 0.1 \cdot (-2 \cdot 0.5 + 2.03 - 4 \cdot 1.2 + 5) = 0.623,$$

$$x = 1.2 + 0.1 = 1.3.$$

Решение:

$$x = 1, \quad y = 2, \quad z = 0.$$

$$x = 1.1, \quad y = 2, \quad z = 0.3.$$

$$x = 1.2, \quad y = 2.03, \quad z = 0.5.$$

$$x = 1.3, \quad y = 2.08, \quad z = 0.623.$$

### 23.3. Решение дифференциальных уравнений в приложении Mathcad

В приложении Mathcad решить дифференциальное уравнение можно, записав формулы выбранного метода. Например, пусть имеется дифференциальное уравнение:

$$\frac{dy}{dx} = 0.2y + x, \quad x_0 = 0, \quad y_0 = 1, \quad h = 0.1.$$

Для решения уравнения методом Эйлера надо на рабочем поле Mathcad записать:

$$h := 0.1 \quad n := 3 \quad i := 0..n$$

$$x_0 := 0 \quad y_0 := 1$$

$$x_{i+1} := x_i + h$$

$$y_{i+1} := y_i + h * (0.2 * y_i + x_i)$$

Для получения численных значений записываются выражения:  
**x =** и **y =**

Имеются и встроенные функции для решения дифференциальных уравнений, например встроенная функция **rkfixed**.

Чтобы решить систему дифференциальных уравнений, приведенную выше, можно записать:

$$x1 := 1 \quad x2 := 1.3 \quad Np := 20$$

$$y_0 := 1 \quad y_1 := 0$$

$$y := \begin{bmatrix} y_0 \\ y_1 \end{bmatrix} \quad D(x, y) := \begin{bmatrix} y_1 \\ -2 * y_1 + y_0 - 4 * x + 5 \end{bmatrix}$$

$$R := rkfixed(y, x1, x2, Np, D)$$

Здесь  $x1$ ,  $x2$  – соответственно левая и правая границы интервала, на котором ищется решение;  $Np$  – число точек, в которых определяется решение;  $y_0$ ,  $y_1$  – начальные условия;  $D(x, y)$  – вектор правых частей системы. Для определения матрицы с решениями надо набрать:  $R =$

Можно также построить графики решения для различных значений  $i$ , характеризующие зависимость  $R(1, i)$  от  $R(0, i)$  и зависимость  $R(2, i)$  от  $R(0, i)$ , т. е. зависимость  $y$  от  $x$  и зависимость  $z$  от  $x$ .

### 23.4. Задание для выполнения на компьютере

1. Решить дифференциальные уравнения, представленные в таблице. В приложении Excel для дифференциального уравнения первого порядка на языке VBA разработать программу решения методом Рунге – Кутты, а для системы дифференциальных уравнений – методом Эйлера.

2. Выполнить вычисления в пакете Mathcad. Результаты сравнить между собой.

#### Исходные данные для расчета

№ п/п	Уравнения	Интервал и шаг	Начальные условия
1	2	3	4
1	$\frac{dy}{dx} = -xy + 2y$	$[0; 1],$ $h = 0.05$	$y(0) = 1$
	$\frac{dz1}{dx} = 3z1 + z2; \quad \frac{dz2}{dx} = -z1 + 4z2 - x$	$[2; 4],$ $h = 0.1$	$z1(2) = 3,$ $z2(2) = 0$
2	$\frac{dc}{dt} = 0.2c - t$	$[1; 2],$ $h = 0.05$	$c(1) = 1$
	$\frac{dy1}{dx} = y1 - y2 + x; \quad \frac{dy2}{dx} = 0.5y1 + y2$	$[0; 1],$ $h = 0.1$	$y1(0) = 2,$ $y2(0) = 2$

Продолжение таблицы

1	2	3	4
3	$\frac{dx_1}{dt} = 0.3x_1 + x_2; \frac{dx_2}{dt} = x_1 - x_2 + 2t$	[1; 3], $h = 0.1$	$x_1(1) = 1,$ $x_2(1) = 0$
	$\frac{dy}{dx} = 0.5y + x$	[0; 1], $h = 0.05$	$y(0) = 4$
4	$\frac{dp_1}{dx} = p_1 + p_2 - x; \frac{dp_2}{dx} = 0.2p_1 - p_2$	[10; 12], $h = 0.1$	$p_1(10) = 3,$ $p_2(10) = 0$
	$\frac{dy}{dx} = 2 \cos x - y$	[1; 2], $h = 0.05$	$y(1) = 10$
5	$\frac{dz}{dt} = 3.4z - tz$	[0; 2], $h = 0.2$	$z(0) = 1$
	$\frac{dy_1}{dx} = y_1 - y_2; \frac{dy_2}{dx} = 0.4y_1 + y_2$	[1; 2], $h = 0.05$	$y_1(1) = 5,$ $y_2(1) = 0$
6	$\frac{dy}{dx} = xy + 5y$	[0; 1], $h = 0.05$	$y(0) = 8$
	$\frac{dz_1}{dx} = 8z_1 + z_2; \frac{dz_2}{dz} = -z_1 + z_2 - 0.2x$	[2; 3], $h = 0.1$	$z_1(2) = 3,$ $z_2(2) = 1$
7	$\frac{dc}{dt} = 2c - t$	[1; 2], $h = 0.05$	$c(1) = 1$
	$\frac{dy_1}{dx} = y_1 - y_2 + x; \frac{dy_2}{dx} = 0.6y_1 + y_2$	[0; 1], $h = 0.1$	$y_1(0) = 2,$ $y_2(0) = 0$
8	$\frac{dx_1}{dt} = 0.4x_1 + x_2; \frac{dx_2}{dt} = x_1 - x_2 + 2t$	[2; 3], $h = 0.1$	$x_1(1) = 1,$ $x_2(1) = 0$
	$\frac{dy}{dx} = 4.5y + x$	[0; 1], $h = 0.05$	$y(0) = 9$
9	$\frac{dp_1}{dx} = p_1 + p_2 - x; \frac{dp_2}{dx} = 0.3p_1 - p_2$	[1; 2], $h = 0.1$	$p_1(1) = 3,$ $p_2(1) = 1$
	$\frac{dy}{dx} = 2 \sin x - y$	[1; 2], $h = 0.05$	$y(1) = 5$
10	$\frac{dz}{dt} = 5z - tz$	[0; 2], $h = 0.2$	$z(0) = 1$
	$\frac{dy_1}{dx} = y_1 - y_2; \frac{dy_2}{dx} = y_1 + y_2 - x$	[1; 2], $h = 0.05$	$y_1(1) = 6,$ $y_2(1) = 0$
11	$\frac{dy}{dx} = -3xy + y$	[0; 1], $h = 0.05$	$y(0) = 1$
	$\frac{dz_1}{dx} = z_1 + 7z_2; \frac{dz_2}{dx} = -z_1 + z_2 - x$	[2; 3], $h = 0.1$	$z_1(2) = 3,$ $z_2(2) = 0$
12	$\frac{dc}{dt} = 0.8c - t$	[1; 2], $h = 0.05$	$c(1) = 1$
	$\frac{dy_1}{dx} = y_1 - y_2 + x; \frac{dy_2}{dx} = 5y_1 + y_2$	[0; 1], $h = 0.1$	$y_1(0) = 0,$ $y_2(0) = 2$

## Окончание таблицы

1	2	3	4
13	$\frac{dx_1}{dt} = 4x_1 + x_2; \quad \frac{dx_2}{dt} = x_1 - x_2 + 2t$	$[1; 3],$ $h = 0.1$	$x_1(1) = 5,$ $x_2(1) = 0$
	$\frac{dy}{dx} = 0.9y + x$	$[0; 1],$ $h = 0.05$	$y(0) = 4$
14	$\frac{dp_1}{dx} = p_1 + p_2 - x; \quad \frac{dp_2}{dx} = 0.2p_1 - p_2$	$[0; 2],$ $h = 0.1$	$p_1(0) = 3,$ $p_2(0) = 0$
	$\frac{dy}{dx} = 3 \cos x - 4.5y$	$[1; 2],$ $h = 0.05$	$y(1) = 6$
15	$\frac{dz}{dt} = 4z - tz$	$[0; 2],$ $h = 0.2$	$z(0) = 1$
	$\frac{dy_1}{dx} = y_1 - y_2; \quad \frac{dy_2}{dx} = 3y_1 + y_2 - x$	$[1; 2],$ $h = 0.05$	$y_1(1) = 5,$ $y_2(1) = 1$



## 24. ПРИЛОЖЕНИЕ MACROMEDIA FLASH

Приложение предназначено для создания анимированных проектов на основе векторной графики (рис. 24.1).

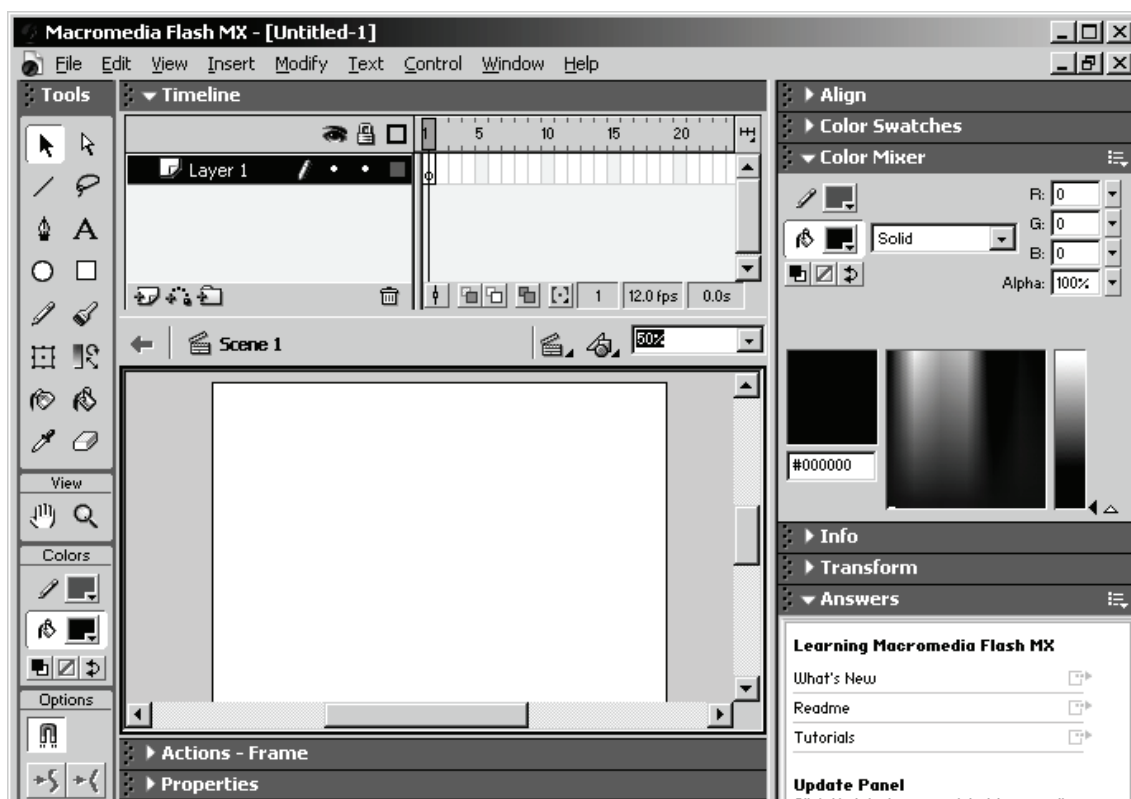


Рис. 24.1. Структура основного окна Macromedia Flash

Рабочая среда Flash состоит из рабочего поля, или сцены (**Stage**), – центральной области экрана, в которой происходит создание и воспроизведение клипа; панели инструментов (**Toolbar**), по умолчанию расположенной слева, которая используется для создания и редактирования изображений; монтажной линейки (**Timeline**), по умолчанию расположенной в верхней части экрана, в которой создается анимация графики во времени; окна библиотеки (**Library**), которое вызывается командой **Window/Library**. В библиотеке находятся элементы клипа, предназначенные для многократного использования, такие элементы называются **символами (Symbols)**.

Панели инструментов и дополнительные панели добавляются на экран соответствующими командами из пункта меню **Window**.

## 24.1. Создание объектов

*Рисование линий и фигур.* Для рисования используется панель инструментов, которая обычно располагается слева на экране. При выборе одного из инструментов в нижней части панели появляется раздел **Options** – дополнительные опции для каждого инструмента. Под рабочим полем расположена панель **Properties**, которую можно использовать для изменения свойств выбранных инструментов.

При создании объектов применяются следующие инструменты:

– **Line Tool** – вычерчивание прямой линии (для рисования горизонтальной или вертикальной линии нужно удерживать клавишу <Shift>). Для изменения свойств линии можно воспользоваться возможностями инспектора свойств **Properties**;

– **Pen Tool** – рисование линий по точкам, сегментов кривых. Если дважды щелкнуть на последней точке, то фигура останется незамкнутой. Если щелкнуть по начальной точке – фигура замкнется. Чтобы нарисовать сегмент кривой, надо отметить первую точку, нажать мышку для создания второй точки и, не отпуская мыши, переместить указатель в направлении, противоположном направлению изгиба кривой. Окончание работы – нажатие клавиши <Esc> или переключение на другой инструмент;

– **Oval Tool** – рисование овала (для изображения круга надо удерживать клавишу <Shift>);

– **Rectangle Tool** – рисование прямоугольника (для изображения квадрата нужно удерживать клавишу <Shift>). На панели **Options** можно скруглить углы;

– **Pencil Tool** – рисование от руки карандашом. На панели **Options** можно сгладить углы при помощи опции **Smooth**;

– **Brush Tool** – рисование кистью. На панели **Options** можно выбрать размер (**Brush Size**) и форму (**Brush Shape**). **Brush Mode** настраивает режим: можно рисовать впереди объектов с сохранением прозрачности (**Paint Fills**), позади объектов (**Paint Behind**), заполнять внутри контура с сохранением элементов рисунка (**Paint Inside**);

– **Paint Bucket Tool** – заливка цветом. На панели **Options** можно задать область;

– **Eyedropper Tool** – выбор цвета;

– **Eraser Tool** – стирка.

Для удобства рисования можно установить сетку командой **View/Grid/Show Grid**.

Каждый объект имеет цвет контура и цвет заполнения, которые определяются на панели **Colors (Stroke Color, Fill Color)**. Заполнение может быть сплошным (**Solid**), градиентным (**Linear** – линейным, **Radial** – радиальным), картинкой (**Bitmap**).

В окне **Color Mixer** (рис. 24.2) можно изменять цвета градиентного заполнения.

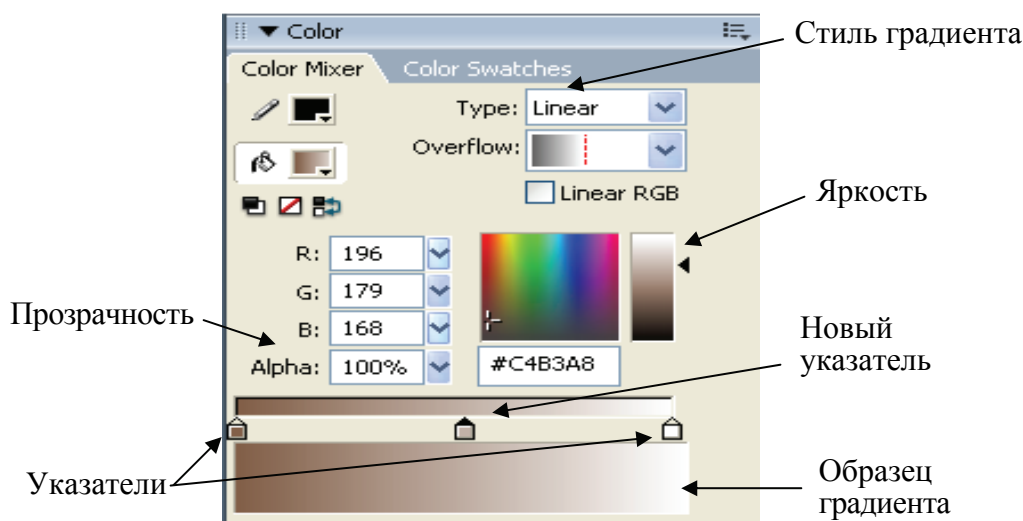


Рис. 24.2. Окно **Color Mixer**

Для изменения цвета нужно выделить соответствующий указатель и выбрать в палитре новый цвет. При этом можно изменять яркость цвета и процент прозрачности **Alpha**. Обычно используется два цвета, но можно добавить и больше, для чего надо сделать щелчок на линии цветового перехода, появится новый указатель, для которого следует установить нужный цвет. Тип градиента выбирается в раскрывающемся списке **Type**. С помощью инструмента **Fill Transform** можно преобразовать градиентную заливку (выбрать инструмент, щелкнуть по объекту с градиентной заливкой и переместить нужные маркеры).

**Выделение объектов, их изменение и группировка.** Чтобы выполнить над объектом различные манипуляции, надо его выделить.

Инструмент **Arrow Tool** позволяет выделить область прямоугольной формы двойным щелчком по объекту. Можно выделить отдельно контур и заливку одинарными щелчками. Чтобы выделить ломаную, надо щелкать по каждому отрезку с нажатой клавишей **<Shift>**. С помощью **Arrow** можно изменить форму объекта. Для этого надо выбрать инструмент и приблизить его к контуру объекта. Если рядом с курсором появился уголок, то можно переместить угловую точку объекта, если появилась дуга, можно переместить линию контура, зацепив ее мышкой.

Чтобы выделить объект инструментом **Subselection Tool**, надо заключить его в прямоугольную область, при этом по периметру объекта появятся маркеры, с помощью которых можно изменить форму объекта.

Инструмент **Lasso** применяется для выделения произвольной части изображения. Изменить размеры и форму объекта можно также с помощью инструмента **Free Transform Tool**.

Используя **Ink Bottle Tool**, можно сделать линию толще.

Чтобы сгруппировать фигуры, надо их выделить и выполнить команду **Modify/Group**.

**Работа с текстом.** Для создания текста служит инструмент **Text**. Различают три типа текстовых полей: статические (**Static**), динамические (**Dynamic**) и поля ввода (**Input**). Тип поля выбирается в инспекторе свойств **Properties** в раскрывающемся списке **Text Type**. Цвет и стиль шрифта, его размер устанавливаются также в инспекторе свойств.

Если нужно работать с текстом как с графикой (изменять контур, применять градиентную заливку), то надо преобразовать текст, предварительно выделив его и выбрав команду **Modify/Break Apart**. Если выполнить команду дважды, то каждая буква станет графическим символом, над которым можно производить различные манипуляции.

**Слои.** Слои (**Layers**) можно представить как прозрачные пленки с нанесенными изображениями, сложенные в стопку друг на друга. Порядок, в котором располагаются слои на монтажной линейке (**Timeline**), определяет, как объекты перекрывают друг друга на рабочем поле. Все названия слоев представлены на временной шкале. Здесь же находятся пиктограммы, с помощью которых можно добавлять и удалять слои, делать их невидимыми и т. д. Для добавления слоя нужно щелкнуть по существующему слою и из контекстного меню выбрать **Insert Layer**, для удаления – **Delete Layer**.

Слой можно сделать невидимым, щелкнув по значку «глаз»; заблокировать действия с ним, щелкнув по значку «замок». Значок с квадратиком управляет показом контуров слоев.

Различают слои обычные, слои траекторий и слои масок. Их использование будет рассмотрено ниже.

## 24.2. Анимация

Для работы с анимацией предназначена временная шкала **Timeline**, которая отображает слои фильма (**Layer1, Layer2, ...**) и кадры каждого слоя. Обычно отдельные объекты рисуются в разных слоях.

При анимации используются три типа кадров:

- ключевой кадр (**Keyframe**) вставляется нажатием <F6>, повторяет содержимое предыдущего кадра;
- пустой ключевой кадр (**Blank Keyframe**) вставляется нажатием <F7>, разбивает фильм на части, очищает содержимое предыдущего кадра;
- неключевой обычный кадр (**Frame**) вставляется нажатием <F5>, продлевает длительность фильма.

Инспектор свойств **Properties** отображает свойства выделенного объекта или всего фильма (**Background** – фон, **Size** – размер картинки, **Frame rate** – скорость кадров и др.).

Существуют два типа анимации: *покадровая* (весь фильм рисуется самостоятельно по кадрам и состоит только из ключевых кадров), *автоматическая* (создается первый и конечный кадр, и далее Flash сам создает промежуточные кадры).

Для создания покадровой анимации надо в первом кадре нарисовать рисунок, выделить второй кадр и нажать клавишу <F6> для вставки ключевого кадра, повторяющего содержание предыдущего. Затем следует в рисунок внести изменения и снова вставить следующий ключевой кадр. Эти действия повторяются до конца.

Автоматическая анимация делится на автоматическую анимацию формы (**shape tweening**) и автоматическую анимацию движения (**motion tweening**).

**Анимация формы.** При анимации формы в первый кадр помещают графический объект, а в последнем этот объект изменяют или заменяют на другой. Flash производит изменение формы в промежуточных кадрах. Чтобы применить анимацию формы к группам или растровым изображениям, необходимо сначала разбить объекты на составляющие командой **Modify/Break Apart**. К тексту эту команду надо применять дважды, чтобы текст разбился на буквы, а они, в свою очередь, на составляющие.

Для анимации формы нужно проделать следующие действия:

1. Создать исходный объект в первом кадре.
2. Выделить первый кадр на **Timeline** и на панели свойств **Properties** в поле **Tween** выбрать **Shape**.
3. В окошке **Ease** можно ввести число, которое задает скорость изменения промежуточных кадров.
4. Вставить пустой ключевой кадр (нажав клавишу <F7>) через необходимое количество кадров от первого ключевого. Создать новый объект, в который должен преобразоваться исходный.

5. Посмотреть результат, нажав одновременно клавиши <Ctrl> + <Enter> или выполнив команду **Control/Play**.

Для создания анимации с несколькими преобразованиями п. 4 следует повторить нужное количество раз.

**Анимация движения.** Можно анимировать расположение объекта, его размер, угол поворота, цвет, прозрачность. Для создания такой анимации надо выполнить следующие действия:

1. Создать исходный объект. Выделить его инструментом **Arrow Tool** и выполнить команду **Modify/Group**.

2. Выделить первый кадр на **Timeline** и в инспекторе свойств **Properties** в поле **Tween** выбрать **Motion** (или выполнить команду **Insert/Motion Tween**).

3. Вставить следующий ключевой кадр через некоторое количество кадров от первого ключевого (нажав клавишу <F6>). При этом временная шкала **Timeline** должна окраситься в сиреневый цвет. Следует переместить объект в нужное место. (П. 3 повторяется необходимое число раз.)

4. Чтобы посмотреть результаты, надо нажать одновременно клавиши <Ctrl> + <Enter> или выполнить команду **Control/Play**.

На панели свойств можно отрегулировать скорость кадров (пункт **Frame rate**).

Такая анимация предполагает движение фигуры из начального положения в конечное по кратчайшей траектории.

**Анимация движения по заданной траектории.** В приложении Macromedia Flash существует понятие ведущего слоя (**Guide Layer**). Ведущий слой имеет два назначения: на него можно поместить какие-либо комментарии, которые видны только в режиме редактирования фильма, и с его помощью можно управлять траекторией перемещения объекта.

Для создания анимации движения по заданной траектории надо проделать следующие действия:

1. Создать анимацию прямолинейного движения объекта.

2. Щелкнув на имени слоя в инспекторе свойств, можно включить флажок **Orient to Path** (ось анимируемого объекта будет ориентирована по направлению пути) и флажок **Snap** (объект будет «прилипать» к пути).

3. Щелкнуть правой кнопкой мыши на имени слоя, содержащего анимированный объект, и выбрать в контекстном меню команду **Add Motion Guide** (добавить слой, управляющий движением). В списке слоев появится ведущий слой, помеченный специальным значком, а имя слоя с анимацией сдвинется вправо – это признак того, что он является ведомым слоем.

4. Щелкнуть на имени ведущего слоя и с помощью любого инструмента рисования (**Pen**, **Pencil**, **Oval**, **Brush**, **Rectangle**) изобразить траекторию движения объекта.

5. Выделить первый кадр анимации и совместить центр анимируемого объекта с началом траектории.

6. Выделить последний кадр анимации и совместить центр анимируемого объекта с концом линии в последнем кадре.

7. После того, как траектория будет создана, можно сделать ее невидимой. Для этого требуется сделать невидимым ведущий слой (щелкнуть по точке под значком «глаз»).

8. Посмотреть результаты.

### 24.3. Маскирование слоев

Слой-маска позволяет создавать эффект отверстия, через которое «просвечивает» содержание нижележащих слоев. Маска может быть создана на основе следующих типов объектов слоя: заливки, текстового поля, экземпляра графического символа.

Маску можно заставить перемещаться, используя любой тип анимации. С помощью ведущего слоя для маски можно задать произвольную траекторию движения.

Слой-маска закрывает (маскирует) по умолчанию только тот слой, который расположен непосредственно под ним.

Чтобы маскировать слой, необходимо:

1. Щелкнуть правой кнопкой мыши на имени слоя, который нужно маскировать (закрыть маской), и в контекстном меню выбрать команду **Insert Layer** (этот новый слой впоследствии будет служить маской).

2. Поместить на слой-маску заливку или заливки, которые будут использоваться в качестве «смотровых окон»; если маска будет анимирована, то в слое должна присутствовать только одна заливка.

3. Щелкнуть правой кнопкой мыши на имени слоя-маски и в контекстном меню выбрать пункт **Mask (Маска)**. С этого момента новый слой становится маской (слева от имени слоя-маски появится соответствующий значок, имя маскируемого слоя сдвинется вправо, и возле него также появится новый значок; оба слоя будут автоматически заблокированы, т. е. появятся «замки» справа от имени слоя).

В результате выполнения указанных действий включается режим маскирования и через маску будут видны только незакрытые участки маскированного слоя.

Можно изменить расположение, форму и количество «смотровых окон» маски. Чтобы сделать маску доступной для редактирования, достаточно снять с нее блокировку (щелкнуть по значку «замок»). При этом автоматически снимается и режим маскирования. Переход между режимами редактирования и просмотра маски может также выполняться с помощью команд из контекстного меню слоя (включить режим редактирования – **Show All**, включить режим просмотра маски – **Show Masking**).

#### 24.4. Символы. Создание кнопок

**Символ** – это объект (элемент фильма), который помещается в библиотеку фильма и может быть неоднократно использован в этом или в другом фильме. Преимущество символов по сравнению с обычными объектами состоит в том, что их применение ускоряет процесс создания фильма и уменьшает его размер. Символы бывают трех типов: **Graphic** (**Графический символ**), **Movie Clip** (**Видеоклип**), **Button** (**Кнопка**). Графические символы представляют собой рисунки. Видеоклипы – это анимированные графические объекты, которые можно вставлять в фильм сколько угодно раз. Кнопки являются объектами интерактивного интерфейса и содержат графическую часть и программную.

Для создания нового символа необходимо выполнить команду **Insert/New Symbol**. В диалоговом окне вводится имя символа и указывается его тип. Символ можно создать из уже имеющегося объекта, для этого его необходимо выделить и выполнить команду **Modify/Convert to Symbol**.

С целью изменения цвета, яркости и прозрачности образца символа нужно его выделить инструментом **Arrow** и использовать соответствующее раскрывающееся меню инспектора свойств.

**Кнопки** позволяют внести в фильм элементы интерактивности, т. е. зритель может управлять порядком воспроизведения фильма. Кнопки могут реагировать на различные действия мышки: наезд, нажатие и др. Различные состояния кнопки описываются графически, а действия, на которые реагирует кнопка, назначаются с помощью встроенного языка программирования ActionScript. Временная шкала кнопки состоит из четырех кадров:

– 1-й кадр, или **Up**-состояние, показывает обычное состояние кнопки;



– 2-й кадр, или **Over**-состояние, показывает состояние кнопки, когда указатель мыши расположен над кнопкой;

– 3-й кадр, или **Down**-состояние, показывает состояние кнопки при ее нажатии;

– 4-й кадр, или **Hit**-состояние, определяет область, в которой кнопка реагирует на щелчок мышкой. Эту область в клипе не видно. Если **Hit**-кадр не нарисован, рабочая область будет соответствовать **Up**-кадру.

Чтобы создать кнопку, необходимо выполнить следующие действия:

1. Выполнить команду **Insert/New Symbol**.

2. В диалоговом окне ввести имя кнопки и выбрать тип символа **Button (Кнопка)**.

3. В режиме редактирования символа монтажная линейка будет состоять из последовательности четырех кадров, которые помечены как **Up**, **Over**, **Down** и **Hit**. С помощью инструментов рисования создать в первом ключевом кадре (**Up**) изображение кнопки.

4. Выделить второй кадр, помеченный как **Over**, и выбрать команду **Insert/Keyframe**. Flash создаст ключевой кадр, полностью повторяющий содержание **Up**-кадра. Внести необходимые изменения в **Over**-состояние кнопки (например, увеличить изображение на 10% и изменить цвет).

5. Повторить последние два шага для **Down**- и **Hit**-кадра кнопки.

6. Завершив создание кнопки, перетащить символ-кнопку из панели библиотеки на сцену, тем самым создав экземпляр кнопки для клипа.

При создании изображения кнопки можно использовать графические экземпляры символов, а для создания анимированной кнопки в кадр помещают видеоклипы. Увидеть работу кнопок в режиме редактирования можно, выполнив команду **Control/Enable Simple Buttons**.

Чтобы назначить кнопке команду, необходимо щелкнуть правой клавишей мыши по экземпляру кнопки и выполнить в контекстном меню команду **Actions**. На открывшейся панели **Actions** в списке команд двойным щелчком надо выбрать нужную команду языка программирования и при необходимости ввести требуемые параметры.

**Сохранение файлов и публикация Flash-роликов в Интернете.** Flash-файлы сохраняются в формате **FLA**. Файл формата **FLA** содержит все необходимые ресурсы, включая изображения, видеоматериал, звук и код, и может быть отредактирован.

Для показа фильма с помощью Flash-проигрывателя или в сети Интернет необходимо его опубликовать в формате **HTML**-документа

посредством команды **File/Publish**. При этом автоматически будет создан файл фильма (\*.SWF) и описывающий его открытие **HTML**-документ.

Чтобы протестировать созданную публикацию, надо запустить браузер, например **Internet Explorer**, и открыть в нем **HTML**-документ командой **File/Open**.

## 24.5. Задание для выполнения на компьютере

1. Создать анимацию формы следующим образом:

- в 1-м кадре нарисовать круг;
- выделить 1-й кадр на **Timeline** и на панели свойств **Properties** в поле **Tween** выбрать **Shape**;
- выделить 20-й кадр и вставить пустой ключевой кадр, нажав клавишу <F7>. Нарисовать треугольник;
- выделить 40-й кадр и вставить пустой ключевой кадр, нажав клавишу <F7>. Нарисовать квадрат;
- просмотреть ролик.

2. Чтобы создать анимацию движения кораблика, необходимо:

- в 1-м кадре нарисовать кораблик (рис. 24.3). Для выделения всего рисунка щелкнуть по 1-му кадру. Выполнить команду **Insert/Create Motion Tween** (вокруг кораблика появится синяя рамка). Переместить кораблик в правую часть сцены;
- выделить 20-й кадр и вставить ключевой кадр (нажав клавишу <F6>). Переместить кораблик в левую часть сцены;
- выделить 21-й кадр и вставить ключевой кадр, нажав клавишу <F6>;
- выполнить команду **Modify/Transform/Flip Horizontal** для переворота кораблика;
- выделить 40-й кадр и вставить ключевой кадр (нажав клавишу <F6>). Переместить кораблик в правую часть сцены;
- просмотреть ролик нажатием одновременно кнопок <Ctrl> + <Enter>.

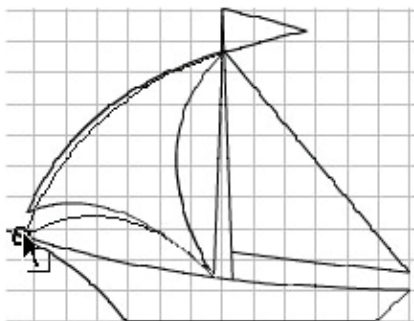


Рис. 24.3. Кораблик

3. Выполнить анимацию движения по заданной траектории следующим образом:

- скопировать кораблик из предыдущего задания;
- создать для него анимацию движения;
- вставить слой, задающий траекторию движения (**Insert/Motion Guide**).

В этом слое с помощью инструментов рисования создать желаемую траекторию;

– совместить центр анимируемого объекта с началом линии в первом кадре и с концом линии в последнем кадре;

– протестировать ролик и сохранить файл.

4. Выполнить маскирование слоя. Для этого проделать следующие действия:

– создать новый документ и нарисовать, например, домик, солнышко, травку. Сделать цветной фон, выбрав любой инструмент выделения, и на панели свойств определить цвет с помощью **Background**;

– в 100-й кадр вставить ключевой кадр, нажав клавишу <F6>;

– вставить над этим слоем новый. Создать в новом слое анимацию движения или формы на 100-м кадре (перемещение по экрану круга или расширение маленького круга из центра на весь экран и обратно). Сделать этот слой маской (из контекстного меню слоя выбрать команду **Mask**);

– протестировать ролик.

5. Использовать графические символы:

– нарисовать елочку, выделить контур и, используя в инспекторе свойств раскрывающийся список **Stroke style**, сделать контур прерывистым;

– поместить графический объект в библиотеку (выделить и выполнить **Insert/Convert to Symbol**). Переносом объекта из библиотеки на сцену создать из елок «лес»;

– создать новый слой, нарисовать колобок и организовать его движение по лесу. Просмотреть результат.

6. Осуществить управление просмотром ролика с помощью кнопок:

– открыть любой файл с анимацией;

– используя команду **Insert/New Symbol**, создать на сцене символ «Кнопка» по образцу (рис. 24.4).

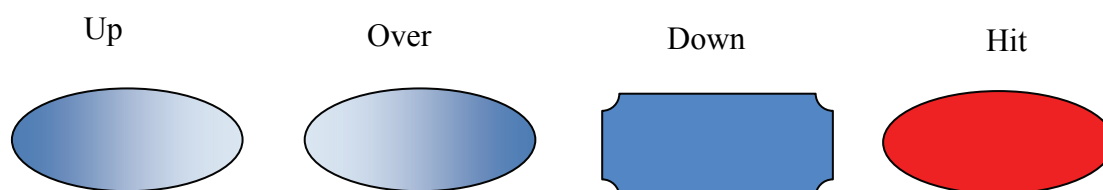


Рис. 24.4. Образцы изображения кадров кнопки

Для закрашивания кнопки нужно воспользоваться градиентной заливкой, а в **Down**-кадре изменить контуры объекта. Щелкнув на **Timeline** пиктограмму **Scene**, вернуться к исходному слою с анимацией;

– создать новый слой и разместить в нем экземпляр кнопки, взятой из библиотеки. Выполнить в контекстном меню команду **Actions** и назначить кнопке программу:

```
on (press) {  
  stop();  
}
```

– запустить фильм и, нажимая на кнопку, протестировать его работу;

– взять из библиотеки и разместить на этом же слое еще один экземпляр кнопки, назначить ей программу:

```
on (press) {  
  play();  
}
```

– опробовать работу кнопок.

## 25. КОМПЬЮТЕРНЫЕ СЕТИ

Если компьютеры объединены в сеть, то у пользователя появляются следующие дополнительные возможности:

- получать информацию с различных компьютеров сети и при необходимости предоставлять свою информацию;
- осуществлять по сети электронные сделки;
- общаться в сети посредством использования различных сервисов.

Различают сети локальные и глобальные. Локальная вычислительная сеть объединяет компьютеры в единую распределенную систему обработки и хранения информации, при этом расстояние между компьютерами обычно не превышает нескольких километров. Компьютеры глобальных сетей могут быть расположены в различных частях земного шара.

### 25.1. Сеть Интернет

В 1969 г. четыре компьютера были объединены в сеть ARPAnet, разработка которой финансировалась Министерством обороны США. Информация в сети передавалась в соответствии с протоколом. Протокол – это набор правил передачи информации, ее защиты, контроля и коррекции ошибок.

В 1983 г. протоколы **TCP/IP** стали основой компьютерной сети, объединяющей американские университеты и другие научно-исследовательские организации. Этот год считается годом рождения сети Интернет.

В 1989 г. был разработан язык гипертекстовой разметки (**HTML**), что привело к созданию в Интернет нового сервиса – сети World Wide Web («всемирная паутина» или WWW).

В настоящее время основная часть информации в Интернет представлена в виде гипертекстовых **HTML**-страниц. Для просмотра гипертекстовых страниц используются специальные программы – web-браузеры. На сегодняшний день существуют различные браузеры. Одним из них является Internet Explorer, который входит в состав ОС Windows.

Популярной услугой сети Интернет является электронная почта (e-mail). Электронная почта позволяет не только обмениваться

письмами, но и прикрепить к письму любой графический файл, документ Word, Excel и т. д.

## 25.2. Язык гипертекстовой разметки HTML

Все страницы в системе World Wide Web созданы с помощью специального языка, который называется языком гипертекстовой разметки документов – **Hyper Text Markup Language (HTML)**.

Web-страницы могут содержать текстовую информацию, рисунки, гиперссылки на другие страницы, звуки и анимацию, фреймы (для деления экрана на несколько областей), формы (для анкетирования, проведения опросов, поиска и т. п.), встроенное видео, встроенные программы на языках **VBScript**, **JavaScript** и т. п.

**Создание документа.** HTML-документ – это текстовый файл с расширением **\*.html** или **\*.htm**. Документ может быть подготовлен в любом текстовом редакторе, например в приложении **Блокнот**.

*Пример 25.1.*

HTML-документ	Web-документ
<pre>&lt;html&gt; &lt;head&gt; &lt;title&gt;Пример 1&lt;/title&gt; &lt;/head&gt; &lt;body&gt; &lt;H1&gt;Минск&lt;/H1&gt; &lt;P&gt;Минск – столица Беларуси.&lt;/P&gt; &lt;/body&gt; &lt;/html&gt;</pre>	Минск

Как видно из примера, информация о форматировании документа сосредоточена в его фрагментах, заключенных между знаками  $\langle \rangle$ . Такой фрагмент (например, **<head>**) называется тегом.

Большинство HTML-тегов – парные, т. е. на каждый открывающий тег вида **<tag>** есть закрывающий тег вида **</tag>** с тем же именем, но с добавлением слеш (/).

Теги можно вводить как прописными, так и строчными буквами. Многие теги, помимо имени, могут содержать атрибуты – элементы, дающие дополнительную информацию о том, как браузер должен обработать текущую метку. Рассмотрим основные теги языка.

**<html> ... </html>** – тег **<html>** должен открывать HTML-документ, а тег **</html>** – завершать.

**<head> ... </head>** – эта пара тегов указывает на начало и конец заголовка документа. Кроме наименования документа, сюда может включаться служебная информация.

**<title> ... </title>** – все, что находится между этими тегами, толкуется браузером как название документа. Название текущего документа отображается обычно в заголовке окна и печатается в левом верхнем углу каждой страницы при выводе на принтер.

**<body> ... </body>** – эта пара тегов указывает на начало и конец тела HTML-документа.

**<h1> ... </h1>** – **<h6> ... </h6>** – теги вида **<h*i*>** (где *i* – цифра от 1 до 6) описывают заголовки шести различных уровней. Заголовок первого уровня – самый крупный, шестого – самый мелкий.

**<p> ... </p>** – такая пара тегов описывает абзац.

Теги **<h*i*>** и **<p>** могут содержать дополнительный атрибут **align**, например:

**<h1 align=center>** Выравнивание по центру **</h1>** или

**<p align=right>** Выравнивание по правому краю **</p>**

**Сохранение HTML-документа.** В редакторе **Блокнот** после набора документа надо сохранить его, выполнив в пункте меню **Файл** команду **Сохранить как...** В появившемся окне следует набрать имя файла с расширением **\*.html** или **\*.htm**, указать место сохранения и нажать кнопку **Сохранить**.

**Просмотр созданных web-страниц.** Чтобы просмотреть результат работы, нужно перейти в браузер Internet Explorer или в другой браузер, выполнить команду **Открыть** из пункта меню **Файл**, нажать кнопку **Обзор** и выбрать созданный файл.

**Теги оформления страниц.** Рассмотрим некоторые теги языка HTML, применяемые при оформлении страниц.

**<br>** используется, если необходимо перейти на новую строку, не прерывая абзаца.

**<hr>** описывает горизонтальную линию. Тег может дополнительно включать атрибуты **size** (определяет толщину линии в пикселах) и/или **width** (определяет размах линии в процентах от ширины экрана).

Между тегами **<!--** и **-->** размещаются комментарии.

Под *физическим стилем* понимается прямое указание браузеру на модификацию текущего шрифта. Например, все, что находится между тегами **<b>** и **</b>**, будет записано жирным шрифтом. Текст между тегами **<i>** и **</i>** будет написан наклонным шрифтом. Текст, размещенный между тегами **<tt>** и **</tt>**, будет написан шрифтом, имеющим фиксированную ширину символа.

При использовании *логических стилей* теги `<em> ... </em>` обозначают акцент (обычно курсив), `<strong> ... </strong>` – сильный акцент (обычно жирный шрифт). `<code> ... </code>` рекомендуется использовать для фрагментов исходных текстов. `<samp> ... </samp>` служат для демонстрации образцов сообщений, выводимых на экран программами. `<kbd> ... </kbd>` обычно применяются для указания того, что нужно ввести с клавиатуры. `<var> ... </var>` необходимы для написания имен переменных.

Текст, расположенный между тегами `<ul>` и `</ul>`, воспринимается как нумерованный список. Каждый новый элемент списка следует начинать с тега `<li>`, у которого нет парного закрывающего тега. Например, чтобы создать список:

- Минск;
- Витебск;

необходим такой текст программы:

```
<ul><li>Минск; </li><li>Витебск; </ul>
```

Нумерованные списки устроены точно так же, как нумерованные, только вместо символов, выделяющих новый элемент, используются цифры. Если модифицировать предыдущий текст:

```
<ol><li>Минск; </li><li>Витебск; </ol>
```

получится список:

1. Минск;
2. Витебск.

В списках определений вместо тега `<li>` используются теги `<dt>` и `<dd>`, не имеющие парных закрывающих меток.

*Пример 25.2.*

Фрагмент HTML-текста	Вид на экране
<pre>&lt;dl&gt; &lt;dt&gt;html &lt;dd&gt;Язык HTML разработал сотрудник&lt;br&gt;Европейской лаборатории физики&lt;br&gt;элементарных частиц Тим Бернерс-Ли. &lt;dt&gt;HTML-документ &lt;dd&gt;Текстовый файл с расширением *.html&lt;/dl&gt;</pre>	<pre>html Язык HTML разработал сотрудник Европейской лаборатории физики элементарных частиц Тим Бернерс-Ли. HTML-документ Текстовый файл с расширением *.html</pre>



Текст, заключенный между тегами `<pre>` и `</pre>`, выводится браузером на экран так, как он есть – со всеми пробелами, символами табуляции и конца строки.

Пример 25.3.

HTML-документ	Web-документ
<pre> &lt;html&gt;&lt;head&gt;&lt;title&gt;Пример 2&lt;/title&gt; &lt;/head&gt; &lt;body&gt; &lt;h2&gt;Минск&lt;/h2&gt; &lt;p&gt;Минск – &lt;b&gt;красивый &lt;/b&gt; &lt;i&gt;чистый &lt;/i&gt;&lt;tt&gt;город&lt;/tt&gt;&lt;/p&gt; &lt;p&gt;&lt;em&gt;В городе имеются магази- ны,&lt;/em&gt; &lt;strong&gt;театры, институты и пр. &lt;/strong&gt; &lt;br&gt;&lt;h3&gt;Список институтов&lt;/h3&gt; &lt;dl&gt; &lt;dt&gt;&lt;dd&gt;&lt;ul&gt;&lt;li&gt;БГУ&lt;/li&gt;&lt;li&gt;БГТУ&lt;/li&gt;&lt;/ul&gt; &lt;dt&gt;Список магазинов &lt;dd&gt;&lt;ol&gt;&lt;li&gt;Беларусь&lt;/li&gt;&lt;li&gt;ГУМ&lt;/li&gt; &lt;/ol&gt; &lt;/dd&gt; &lt;/dl&gt; </pre>	<p><b>Минск</b></p> <p>Минск – <b>красивый</b> <i>чистый</i> город</p> <p><i>В городе имеются магазины, театры, институты и пр.</i></p> <p><b>Список институтов</b></p> <ul style="list-style-type: none"> <li>• БГУ</li> <li>• БГТУ</li> </ul> <p>Список магазинов</p> <ol style="list-style-type: none"> <li>1. Беларусь</li> <li>2. ГУМ</li> </ol>

### 25.3. Задание для выполнения на компьютере

1. В текстовом редакторе **Блокнот** создать **HTML**-документ с произвольным содержимым, включающий текст, списки. Определить различный шрифт текста.

2. Просмотреть созданный документ в браузере.

## 26. РАЗРАБОТКА WEB-СТРАНИЦ

### 26.1. Гиперссылки и рисунки в web-документах

**Гиперссылки.** В HTML-документах переход от одного фрагмента текста к другому задается с помощью тега вида:

**<a href="[адрес перехода]">выделенный фрагмент текста</a>**

В качестве параметра [адрес перехода] может использоваться несколько типов аргументов:

1. Чтобы организовать ссылку на другой документ, надо задать имя этого HTML-документа, к которому нужно перейти. Например:

**<a href="primer1.htm">столица</a>**

Такой текст приведет к появлению в документе гиперссылки, т. е. подчеркнутого текста столица, при нажатии мышкой на который в текущее окно будет загружен документ с именем **primer1.htm**.

Если в адресе перехода не указано место размещения этого документа, то компьютер будет искать его в текущей папке.

2. Если необходимо дать ссылку на документ, находящийся на другом компьютере, нужно использовать тег, например, вида:

**<a href="http://www.ok.open.by/about.htm"> информация</a>**

3. При необходимости можно задать переход не просто к некоторому документу, но и к определенному месту внутри этого документа. Для этого следует создать в документе, к которому будет задан переход, некоторую опорную точку или закладку.

Пусть, например, необходимо осуществить переход из файла **1.htm** к словам «Раздел 5» в файле **2.htm** (файлы находятся в одной папке). Прежде всего, следует создать в файле **2.htm** следующий тег:

**<a name="aaa">Раздел 5</a>**

Слова «Раздел 5» при этом никак не будут выделены в тексте документа. Затем в файле **1.htm** можно определить переход на этот тег:

**<a href="2.htm#aaa">Переход к разделу 5</a>**

4. Если пользователь совершит переход по ссылке:

**<a href="mailto:user@mail.box">Послать письмо</a>**

то у него на экране откроется окно ввода исходящего сообщения его почтовой программы. В строке **To:** (Куда) окна почтовой программы будет указано **user@mail.box**.

**Изображения в HTML-документе.** В HTML-документах используются графические файлы в формате **\*.gif** или **\*.jpg**. Для вставки изображения с именем **picture.gif**, находящегося в одной папке с HTML-документом, надо использовать тег вида:

```

```

Такой тег может также включать атрибут **alt="[текст]"**, например:

```

```

Встретив такой тег, браузер покажет на экране текст **Картинка** и начнет загружать на его место картинку из файла **picture.gif**. Атрибут **alt** может оказаться необходимым на случай, если у браузера отключена автоматическая загрузка изображений.

Чтобы поместить Flash-ролик на web-страницу, надо в Macromedia Flash выполнить команду **File/Publish Preview/HTML**. Откроется Flash-ролик в одном из браузеров. Нужно выбрать команду **Вид/Исходный код страницы** (или **Вид/Источник**), в появившемся окне скопировать текст между тегами **<object>** и **</object>**, включая эти теги, и вставить в тело HTML-документа. Для просмотра ролика надо открыть HTML-документ в одном из браузеров. В браузере Internet Explorer возможна ситуация, когда приложение показывает панель информации с предупреждением о наличии элемента **ActiveX**. Надо щелкнуть на верхней панели и выбрать пункт **Разрешить заблокированное содержимое**.

**Цветовая гамма.** Цвета всего HTML-документа задаются атрибутами, размещенными внутри тега **<body>**: **bgcolor** определяет цвет фона документа; **text** – цвет текста документа; **link** – цвет гиперссылки; **vlink** – цвет ссылки на документ, который уже был просмотрен ранее; **alink** – цвет ссылки, когда на нее указывает курсор мыши и нажата ее правая кнопка.

Цвет кодируется последовательностью из трех пар символов. Каждая пара представляет собой шестнадцатеричное значение насыщенности заданного цвета одним из трех основных цветов (красным, зеленым и синим) в диапазоне от нуля (00) до 255 (FF).

Например: **bgcolor=#FFFFFF** – цвет фона (здесь – белый), **text=#000000** – цвет текста (здесь – черный), **link=#FF0000** – цвет гипертекстовой ссылки (здесь – красный).

Тег **<body>** может включать атрибут **background="[имя файла]"**, который задает изображение, служащее фоном для текста и других изображений. Фон должен быть представлен в формате **\*.gif** или **\*.jpg**. Браузеры заполняют копиями изображения-фона все окно.

Цвет фона и изображение-фон не отображаются при выводе **HTML**-документа на печать, поэтому лучше не использовать текст белого цвета.

*Пример.*

HTML-документ	Web-документ
<pre> &lt;html&gt;&lt;head&gt; &lt;title&gt;Пример 3&lt;/title&gt; &lt;/head&gt; &lt;body&gt; &lt;h3&gt; Минск – столица &lt;a href="primer1.htm"&gt; Беларуси &lt;/a&gt;&lt;/h3&gt; &lt;p&gt;&lt;img src="minsk.jpg"&gt;&lt;/p&gt; &lt;p&gt;Щелкните по нижнему рисунку&lt;/p&gt; &lt;p&gt;&lt;a href="pr.html"&gt; &lt;img src="chuch.jpg"&gt;&lt;/a&gt;&lt;/p&gt; &lt;/body&gt; &lt;/html&gt; </pre>	<p>Минск – столица <b>Беларуси</b></p>  <p>Щелкните по нижнему рисунку</p> 

В этом примере используются файлы с рисунками **minsk.jpg** и **chuch.jpg**, которые хранятся в той же папке, что и исходный пример, а также устанавливается гиперсвязь с файлом **primer1.htm**, также хранящимся в этой же папке.

## 26.2. Использование таблиц

Страница может быть представлена в виде таблицы. Тогда в **HTML**-документе используются теги **<table>** и **</table>**. Тег может включать несколько атрибутов:

– **align** – устанавливает расположение таблицы по отношению к полям документа. Допустимые значения: **align=left** (выравнивание влево), **align=center** (выравнивание по центру), **align=right** (выравнивание вправо);

– **width** – ширина таблицы. Ее можно задать в пикселах (например, **width=400**) или в процентах от ширины страницы (например, **width=80%**);

– **border** – задает ширину внешней рамки таблицы и ячеек в пикселах (например, **border=4**). Если атрибут не установлен, таблица показывается без рамки;

– **cellspacing** – устанавливает расстояние между рамками ячеек таблицы в пикселах.

Таблица может иметь заголовок (**<caption> ... </caption>**), хотя он не является обязательным. Этот тег также может включать атрибут **align**. Допустимые значения: **<caption align=top>** (заголовок над таблицей) и **<caption align=bottom>** (заголовок под таблицей).

Каждая строка таблицы начинается с тега **<tr>** и заканчивается тегом **</tr>**. Он может включать атрибуты:

– **align** – устанавливает выравнивание текста в ячейках (влево, по центру, вправо);

– **valign** – задает вертикальное выравнивание текста в ячейках строки. Допустимые значения: **valign=top** (выравнивание по верхнему краю), **valign=middle** (выравнивание по центру), **valign=bottom** (выравнивание по нижнему краю).

Каждая ячейка таблицы начинается с метки **<td>** и заканчивается меткой **</td>**. Тег может включать следующие атрибуты:

– **nowrap** – означает, что содержимое ячейки должно быть показано в одну строку;

– **colspan** – устанавливает «размах» ячейки по горизонтали. Например, **colspan=3** означает, что ячейка простирается на три колонки;

– **rowspan** – задает «размах» ячейки по вертикали. Например, **rowspan=2** означает, что ячейка занимает две строки;

– **width** – устанавливает ширину ячейки в пикселах;

– **height** – задает высоту ячейки в пикселах (например, **height=40**).

Если ячейка таблицы пуста, вокруг нее не рисуется рамка. Если ячейка пуста, а рамка нужна, в ячейку можно ввести символьный объект **&nbsp;**. Ячейка таблицы может содержать в себе другую таблицу.

### 26.3. Создание web-страниц в приложении FrontPage

Для профессионального создания web-страниц существуют специальные программы, которые называются гипертекстовыми

редакторами. Одним из них является приложение FrontPage, которое входит в MS Office.

Окно редактора делится по вертикали на три части. Слева размещается панель переключателя режимов (**Views**). С помощью имеющихся в ней кнопок можно выбрать один из вариантов представления web-сайта. На панели **Folder List (Список папок)** показана иерархическая структура папок и файлов, входящих в состав издания. Основную часть рабочего окна занимает панель представления, в которой отображается выбранный элемент структуры web-сайта, обычно одна из страниц, которая входит в состав данного узла.

Web-сайт может быть представлен в следующих режимах:

- режим страницы (**Page**) позволяет редактировать любую выбранную страницу, изменять размещенный на ней текст и рисунки, вводить новые теги, задавать гиперссылки. Для переключения режимов просмотра страницы в нижней части панели представления имеются три вкладки: **Normal**, **HTML** и **Preview**;

- кнопка **Folders (Папки)** включает режим анализа структуры папок и файлов, входящих в web-сайт;

- в режиме **Reports (Отчеты)** в рабочем окне представлен отчет о текущем состоянии web-сайта;

- кнопка **Navigation (Навигация)** отображает в рабочем окне программы структуру сайта с гиперссылками, позволяющими перемещаться как внутри данного сайта, так и на страницы других сайтов;

- кнопка **Hyperlinks (Гиперссылки)** в графической форме представляет все гиперссылки, переводящие на текущую страницу и с текущей страницы на другие страницы узла;

- кнопка **Tasks (Задачи)** переводит рабочее окно в режим, напоминающий записную книжку. В нем в рабочем окне отображается список задач, намеченный пользователем для выполнения в будущем.

FrontPage включает в себя все необходимое для работы с web-сайтом: программу FrontPage Explorer для навигации по web-сайту, редактор web-страниц FrontPage Editor; средства для работы с графикой; средства для публикации документов и пр.

Панели **Стандартная** и **Форматирование** используются обычным образом. Кроме того, на панели **Стандартная** имеется кнопка **Web-публикация**, которая осуществляет публикацию разработанного web-сайта в сети по определенному адресу.

*Пример создания web-страницы в редакторе FrontPage.* Надо открыть приложение и выбрать режим создания новой страницы

(**File/New Page**). Затем следует набрать в окне некоторый текст. Установить шрифт (**Format/Font**), стиль, размер, цвет.

Далее можно открыть вкладку **HTML** и просмотреть **HTML**-код созданной страницы, на вкладке **Preview** оценить, как будет выглядеть страница в окне браузера.

Используя вкладку **Normal**, можно изменить размер, цвет, начертание введенного текста.

Для добавления таблицы на страницу служит команда **Table/Insert/Table**.

Одну из строк текста можно сделать бегущей. Для этого надо ее выделить и выполнить команду **Insert/Component/Marquee**.

Для вставки рисунка предназначена команда **Insert/Picture**, а с помощью команды **Insert/Data and Time** можно вставить дату и время.

Чтобы сделать текст гиперссылкой, надо его выделить и выбрать команду **Insert/Hyperlink**. В появившемся окне в строке **URL** следует ввести адрес сайта, на который будет происходить переход.

#### 26.4. Задание для выполнения на компьютере

1. В текстовом редакторе **Блокнот** создать **HTML**-документ с произвольным содержимым, включающий текст, рисунки, таблицы, списки, гиперссылки. Определить различные цвета текста.

2. Просмотреть созданный документ в браузере.

3. Тот же самый документ создать в гипертекстовом редакторе **FrontPage**.

## 27. ЗАЩИТА ИНФОРМАЦИИ И КОМПЬЮТЕРНЫЕ ВИРУСЫ

**Компьютерные вирусы.** Это небольшие программы, способные к саморазмножению и выполнению различных деструктивных действий. На сегодняшний день известны десятки тысяч компьютерных вирусов.

Компьютерные вирусы появились в середине 80-х гг. прошлого века. В 1999 г. вирус **Melissa** поразил десятки тысяч компьютеров во всем мире и нанес ущерб в 80 млн. долларов США. После этого инцидента в мире появился спрос на антивирусные программы.

Компьютерные вирусы становятся более изощренными и разрушительными. Изначально пользователь компьютера мог занести вирус лишь открыв инфицированный файл на дискете, компакт-диске или открыв вложенный файл, полученный с электронной почтой. Чуть позже вирусы стали распространяться непосредственно с Интернет-сайтов. Особой опасности подвергаются пользователи, скачивающие аудио- или видеофайлы с Интернета. Практически не осталось типов файлов, которые невозможно заразить вирусом.

Появились вирусы, заражающие сотовые телефоны и иные устройства, в которых используется программное обеспечение. Даже наличие антивирусного программного обеспечения не гарантирует стопроцентной безопасности компьютеров и сетей. Примерно 12–15% вирусов проникают в компьютеры, несмотря на установленную антивирусную защиту.

**Классификация вирусов.** Можно классифицировать вирусы следующим образом:

- загрузочные вирусы, или **boot**-вирусы, которые заражают **boot**-секторы дисков и могут привести к полной потере всей информации, хранящейся на диске;
- файловые вирусы, которые делятся на вирусы, заражающие программы (файлы с расширением **.exe** и **.com**), макровирусы, заражающие документы Word, Excel и др., вирусы-спутники, использующие имена других файлов, вирусы семейства **dir**, искажающие системную информацию о файловых структурах;
- загрузочно-файловые вирусы, способные поражать как код **boot**-секторов, так и код файлов;



- вирусы-невидимки, или **stealth**-вирусы, фальсифицирующие информацию, прочитанную с диска так, что программа, которой предназначалась эта информация, получает неверные данные;
- ретровирусы, которые заражают антивирусные программы;
- вирусы-черви, снабжающие сообщения электронной почты так называемым заголовком, который по своей сути есть web-адрес местонахождения самого вируса. При попытке прочитать такое сообщение вирус считывает через глобальную сеть Интернет свое «тело» и после загрузки начинает деструктивное действие. Очень опасные вирусы, так как обнаружить их тяжело в связи с тем, что зараженный файл фактически не содержит кода вируса.

*Пути проникновения вирусов:*

- ошибки в операционной системе и установленном программном обеспечении;
- безграмотность пользователей в области компьютерной безопасности;
- дискеты;
- пиратские CD- и DVD-диски;
- USB-флешки;
- электронная почта;
- системы обмена мгновенными сообщениями;
- Интернет и локальные сети.

*Основные ранние признаки заражения компьютера вирусом:*

- уменьшение объема свободной оперативной памяти;
- замедление загрузки и работы компьютера;
- непонятные изменения в файлах, а также изменения размеров и даты последней модификации файлов;
- ошибки при загрузке операционной системы;
- непонятные системные сообщения, музыкальные и визуальные эффекты и т. д.

*Признаки активной фазы вируса:*

- исчезновение файлов;
- произвольное форматирование жесткого диска;
- невозможность загрузки файлов или операционной системы.

Если не принимать меры для защиты от компьютерных вирусов, то последствия заражения могут быть очень серьезными.

**Меры профилактики.** К общим средствам, помогающим предотвратить заражение и его разрушительные последствия, относят:

- резервное копирование информации (создание копий файлов и системных областей жестких дисков);

– чаще всего вирусы распространяются вместе с компьютерными программами, поэтому не надо пользоваться случайными и неизвестными программами;

– перезагрузка компьютера перед началом работы, в частности, в случае, если за этим компьютером работали другие пользователи;

– ограничение доступа к информации, в частности физическая защита дискеты во время копирования файлов с нее.

**Антивирусные программы.** К программным средствам защиты относятся антивирусные программы, выявляющие и обезвреживающие компьютерные вирусы. Следует заметить, что вирусы в своем развитии опережают антивирусные программы, поэтому даже в случае регулярного пользования антивирусными программами нет стопроцентной гарантии безопасности. Антивирусные программы могут выявлять и уничтожать лишь известные вирусы, при появлении нового компьютерного вируса защиты от него не существует до тех пор, пока не будет разработан свой антивирус.

Современные антивирусные пакеты имеют в своем составе специальный программный модуль, называемый эвристическим анализатором, который способен исследовать содержимое файлов на наличие кода, характерного для компьютерных вирусов. Это дает возможность своевременно предупреждать об опасности заражения новым вирусом.

Различают следующие типы антивирусных программ:

– программы-детекторы, используемые для нахождения зараженных файлов одним из известных вирусов. Некоторые программы-детекторы могут также лечить файлы от вирусов или уничтожать зараженные файлы. Существуют специализированные программы-детекторы, предназначенные для борьбы с одним вирусом, и полифаги, которые могут бороться с многими вирусами;

– программы-лекари, которые необходимы для лечения зараженных дисков и программ. Лечение программы состоит в изъятии из зараженной программы «тела» вируса. Также могут быть как полифагами, так и специализированными;

– программы-ревизоры, служащие для выявления заражения вирусом файлов, а также нахождения поврежденных файлов. Эти программы запоминают данные о состоянии программы и системных областей дисков в нормальном состоянии (до заражения) и сравнивают эти данные в процессе работы компьютера. В случае несоответствия данных выводится сообщение о возможности заражения;

– лекари-ревизоры, которые выявляют изменения в файлах и системных областях дисков и, в случае изменений, возвращают их в начальное состояние;

– программы-фильтры, перехватывающие обращения вирусов к операционной системе и сообщающие об этом пользователю. Пользователь может разрешить или запретить выполнение соответствующей операции. Такие программы являются резидентными, т. е. они находятся в оперативной памяти компьютера;

– программы-вакцины, которые используются для обработки файлов и **boot**-секторов с целью предупреждения заражения известными вирусами (в последнее время этот метод применяется все чаще).

Рекомендуется использовать несколько разных антивирусных пакетов одновременно.

На сегодняшний день существует большое количество разнообразных антивирусных программ.

**Антивирус DrWeb.** Один из лучших антивирусов с мощным алгоритмом нахождения вирусов. Полифаг, способный проверять файлы в архивах, документы Word и рабочие книги Excel. Выявляет полиморфные вирусы, которые в последнее время получают все большее распространение. Эвристический анализатор DrWeb, исследуя программы на наличие фрагментов кода, характерных для вирусов, позволяет найти около 90% неизвестных вирусов. При загрузке программы в первую очередь DrWeb проверяет самого себя на целостность, после чего тестирует оперативную память. Программа может работать в диалоговом режиме, имеет удобный настраиваемый интерфейс пользователя.

**Антивирус AVP.** Антивирус AVP (AntiVirus Program) относится к полифагам, в процессе работы проверяет оперативную память, файлы, в том числе архивные, на гибких, локальных, сетевых и компакт-дисках, а также системные структуры данных, такие как загрузочный сектор, таблицу разделов и т. д. Программа имеет эвристический анализатор, который, по утверждениям разработчиков антивируса, способен находить почти 80% всех вирусов. Базы антивирусов к AVP обновляются часто и их можно получить из Интернет. Антивирус осуществляет контроль файловых операций в системе, выявляет вирус до момента реального заражения системы, а также определяет неизвестные вирусы с помощью эвристического модуля.

**Антивирусная система NOD32.** Данная система состоит из нескольких модулей, или системных компонентов. Центр управления NOD32 является центральной программой управления антивирусной

системой NOD32. Система состоит из следующих резидентных модулей и фильтров:

- AMON – резидентный (всегда выполняющийся в оперативной памяти) антивирусный монитор или сканер «на доступе». Эта программа является наиболее важным инструментом антивирусной защиты;

- NOD32 – сканер (также именуемый сканером «по требованию»), запускаемый пользователем вручную или планировщиком автоматически;

- IMON (Интернет-монитор) – этот сканер обеспечивает первую линию защиты, контролируя Интернет-трафик;

- DMON (монитор документов) – данный сканер гарантирует защиту от макровирусов в документах MS Office, работает с приложениями, использующими MS Antivirus API (например MS Office 2000 и выше, Internet Explorer 5.0 и выше);

- EMON (Email-монитор) – этот сканер обеспечивает защиту от вирусов, перенесенных электронной почтой.

Eset NOD32 представлен на рынке с 1997 г. и первым ввел эффективную эвристическую технологию обнаружения прежде неизвестных вирусов. Обладает хорошей эффективностью при обнаружении вирусов, высокой скоростью сканирования и в то же время характеризуется использованием минимального количества системных ресурсов.

## ЛИТЕРАТУРА

1. Гурин, Н. И. Работа в среде Windows с программами Excel и Word: учеб. пособие / Н. И. Гурин. – Минск: БГТУ, 1997. – 84 с.
2. Информатика и компьютерная графика / А. П. Лащенко [и др.]. – Минск: БГТУ, 2004. – 79 с.
3. Гурин, Н. И. Программирование на языке Visual Basic: учеб.-метод. пособие / Н. И. Гурин, Н. Н. Пустовалова. – Минск: БГТУ, 2006. – 151 с.
4. Дятко, А. А. Математический пакет Mathcad 6.0 Plus: учеб. пособие / А. А. Дятко, Т. В. Кишкурно. – Минск: БГТУ, 1999. – 97 с.
5. Пустовалова, Н. Н. Вычислительная математика: метод. указания к выполнению лабораторных работ для студентов всех специальностей / Н. Н. Пустовалова, Т. П. Брусенцова. – Минск: БГТУ, 2002. – 85 с.
6. Гурин, Н. И. Работа с базами данных в СУБД ACCESS: учеб.-метод. пособие для аспирантов и студентов всех специальностей / Н. И. Гурин. – Минск: БГТУ, 2002. – 60 с.
7. Пустовалова, Н. Н. Компьютерная графика / Н. Н. Пустовалова, И. Г. Сухорукова, Д. В. Занько. – Минск: БГТУ, 2005. – 72 с.
8. Лащенко, А. П. Компьютерные информационные технологии / А. П. Лащенко, Т. П. Брусенцова, Н. И. Потапенко. – Минск: БГТУ, 2004. – 59 с.
10. Урбанович, П. П. Создание презентаций в системе MS PowerPoint: учеб.-метод. пособие для аспирантов и студентов всех специальностей / П. П. Урбанович, Н. В. Пацей, И. Г. Сухорукова. – Минск: БГТУ, 2004. – 68 с.
11. Пацей, Н. В. Методы архивации данных: учеб.-метод. пособие для студентов и аспирантов / Н. В. Пацей, П. П. Урбанович. – Минск: БГТУ, 2003. – 76 с.
12. Гусак, А. А. Элементы методов вычислений / А. А. Гусак. – Минск: Выш. шк., 1984. – 257 с.
13. Урбанович, П. П. Основы web-дизайна: учеб. пособие для аспирантов и студентов всех специальностей / П. П. Урбанович, Н. В. Пацей. – Минск: БГТУ, 2004. – 67 с.
14. Информатика. Базовый курс / под ред. С. В. Симоновича. – СПб.: БХВ-Петербург, 2006. – 374 с.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1. АРХИТЕКТУРА И ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ КОМПЬЮТЕРА .....	5
1.1. Обобщенная схема компьютера .....	5
1.2. Операционные системы .....	6
1.3. Программы группы «Стандартные» .....	8
1.4. Группа программ Microsoft Office .....	10
1.5. Задание для выполнения на компьютере .....	11
2. ПРИЛОЖЕНИЕ MICROSOFT WORD .....	12
2.1. Основные операции в приложении Word.....	12
2.2. Таблицы и диаграммы.....	16
2.3. Задание для выполнения на компьютере .....	18
3. ОФОРМЛЕНИЕ ДОКУМЕНТОВ В ПРИЛОЖЕНИИ WORD.....	20
3.1. Текстовые эффекты и ввод формул .....	20
3.2. Оформление документа, его редактирование и печать.....	22
3.3. Задание для выполнения на компьютере .....	25
4. ПРИЛОЖЕНИЕ MICROSOFT EXCEL .....	28
4.1. Основные операции в приложении Excel.....	28
4.2. Вычисления по формулам .....	31
4.3. Построение диаграмм, обработка данных и печать .....	33
4.4. Задание для выполнения на компьютере .....	34
5. ЯЗЫК VBA. ЛИНЕЙНЫЕ И ЦИКЛИЧЕСКИЕ ПРОГРАММЫ.....	36
5.1. Общие сведения о принципах программирования в Excel.....	36
5.2. Основные конструкции VBA .....	38
5.3. Линейные программы. Расчет по формулам.....	41
5.4. Циклические программы .....	42
5.5. Задание для выполнения на компьютере .....	45
6. РАЗВЕТВЛЕНИЯ В ЦИКЛИЧЕСКИХ ПРОГРАММАХ .....	48
6.1. Условный оператор If.....	48
6.2. Вычисление сумм, произведений, экстремумов.....	51
6.3. Одномерные массивы.....	52
6.4. Задание для выполнения на компьютере .....	53

7. ФУНКЦИИ И ПРОЦЕДУРЫ ПОЛЬЗОВАТЕЛЯ.....	58
7.1. Функции пользователя.....	58
7.2. Создание пользовательской функции листа .....	60
7.3. Процедуры пользователя типа Sub .....	61
7.4. Задание для выполнения на компьютере .....	63
8. ФОРМЫ ПОЛЬЗОВАТЕЛЯ.....	65
8.1. Свойства и методы .....	65
8.2. Примеры использования форм.....	67
8.3. Работа со списками.....	70
8.4. Использование комбинированного списка .....	70
8.5. Работа с простым списком.....	72
8.6. Задание для выполнения на компьютере .....	73
9. ПРИМЕР РАЗРАБОТКИ ИНЖЕНЕРНОГО ПРОЕКТА.....	74
9.1. Постановка и решение задачи .....	74
9.2. Формирование новой таблицы программным путем.....	77
9.3. Использование Мастера слияния .....	78
9.4. Задание для выполнения на компьютере .....	79
10. БАЗЫ ДАННЫХ.....	80
10.1. Access – реляционная база данных .....	80
10.2. Создание таблиц и установление связей .....	82
10.3. Задание для выполнения на компьютере .....	86
11. ФОРМИРОВАНИЕ ЗАПРОСОВ В ACCESS .....	87
11.1. Условный запрос .....	87
11.2. Запросы с вычисляемыми полями .....	89
11.3. Параметрический запрос .....	90
11.4. Итоговые запросы.....	91
11.5. Запросы с сортировкой и запросы на обновление.....	92
11.6. Задание для выполнения на компьютере .....	93
12. СОЗДАНИЕ ФОРМ И ОТЧЕТОВ .....	94
12.1. Формы.....	94
12.2. Разработка отчетов .....	97
12.3. Задание для выполнения на компьютере .....	100
13. ПРИЛОЖЕНИЕ POWERPOINT .....	101
13.1. Создание презентации.....	101
13.2. Создание слайдов и объектов .....	103
13.3. Презентация .....	105

13.4. Особенности работы с приложением PowerPoint 2007 .....	106
13.5. Задание для выполнения на компьютере .....	107
14. КОМПЬЮТЕРНАЯ ГРАФИКА. ПРИЛОЖЕНИЕ MS VISIO .....	108
14.1. Виды компьютерной графики .....	108
14.2. Разрешение изображения и цветовые модели .....	108
14.3. Приложение Microsoft Visio 2003 .....	110
14.4. Операции над фигурами в MS Visio .....	113
14.5. Задание для выполнения на компьютере .....	114
15. СОЗДАНИЕ И ОФОРМЛЕНИЕ ДОКУМЕНТОВ MS VISIO .....	115
15.1. Настройка параметров листа и стиль .....	115
15.2. Соединение фигур .....	116
15.3. Группировка фигур. Слои и печать .....	117
15.4. Пользовательская среда разработки .....	119
15.5. Задание для выполнения на компьютере .....	121
16. ГРАФИЧЕСКИЙ РЕДАКТОР ADOBE PHOTOSHOP.....	122
16.1. Основные инструменты и окна .....	123
16.2. Создание изображений.....	124
16.3. Слои в графическом редакторе Adobe Photoshop.....	126
16.4. Выделение областей .....	127
16.5. Сохранение файлов с изображениями.....	128
16.6. Задание для выполнения на компьютере .....	129
17. ОБРАБОТКА ИЗОБРАЖЕНИЙ В ADOBE PHOTOSHOP .....	130
17.1. Просмотр рисунка, изменение размеров и поворот .....	130
17.2. Работа с текстом .....	131
17.3. Корректировка изображений.....	131
17.4. Использование фильтров .....	133
17.5. Создание коллажа.....	135
17.6. Задание для выполнения на компьютере .....	136
18. ПРИЛОЖЕНИЕ MATHCAD .....	137
18.1. Ввод и форматирование информации .....	137
18.2. Вычисления в Mathcad .....	139
18.3. Построение графиков.....	140
18.4. Задание для выполнения на компьютере .....	141
19. ВЫЧИСЛЕНИЕ ИНТЕГРАЛОВ И РЕШЕНИЕ УРАВНЕНИЙ .....	142
19.1. Методы вычисления определенных интегралов .....	142



19.2. Приближенное вычисление интеграла в приложениях Mathcad и Excel .....	144
19.3. Численные методы решения уравнений.....	145
19.4. Решение уравнений в приложениях Mathcad и Excel .....	146
19.5. Задание для выполнения на компьютере .....	147
20. РЕШЕНИЕ СИСТЕМ УРАВНЕНИЙ.....	149
20.1. Системы линейных уравнений.....	149
20.2. Решение систем линейных уравнений в приложениях Mathcad и Excel .....	150
20.3. Решение систем нелинейных уравнений в приложении Mathcad .....	150
20.4. Задание для выполнения на компьютере .....	151
21. МАТЕМАТИЧЕСКИЕ МОДЕЛИ И ОПТИМИЗАЦИЯ ПРОЦЕССОВ .....	152
21.1. Модель одномерного объекта .....	152
21.2. Получение математических моделей в приложениях Mathcad и Excel .....	153
21.3. Модель многомерного объекта .....	154
21.4. Задание для выполнения на компьютере .....	157
22. МЕТОДЫ ОПТИМИЗАЦИИ.....	159
22.1. Метод сканирования .....	159
22.2. Метод случайного поиска.....	160
22.3. Решение задач оптимизации в приложениях Mathcad и Excel .....	161
22.4. Задание для выполнения на компьютере .....	151
23. РЕШЕНИЕ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ.....	163
23.1. Решение уравнений первого порядка .....	163
23.2. Системы дифференциальных уравнений и уравнения высших порядков .....	164
23.3. Решение дифференциальных уравнений в приложении Mathcad .....	165
23.4. Задание для выполнения на компьютере .....	166
24. ПРИЛОЖЕНИЕ MACROMEDIA FLASH .....	169
24.1. Создание объектов.....	170
24.2. Анимация .....	172

24.3. Маскирование слоев .....	175
24.4. Символы. Создание кнопок .....	176
24.5. Задание для выполнения на компьютере .....	178
25. КОМПЬЮТЕРНЫЕ СЕТИ .....	181
25.1. Сеть Интернет .....	181
25.2. Язык гипертекстовой разметки HTML .....	182
25.3. Задание для выполнения на компьютере .....	185
26. РАЗРАБОТКА WEB-СТРАНИЦ .....	186
26.1. Гиперссылки и рисунки в web-документах .....	186
26.2. Использование таблиц .....	188
26.3. Создание web-страниц в приложении FrontPage .....	189
26.4. Задание для выполнения на компьютере .....	191
27. ЗАЩИТА ИНФОРМАЦИИ И КОМПЬЮТЕРНЫЕ ВИРУСЫ .....	192
ЛИТЕРАТУРА .....	197

Учебное издание

**Пустовалова** Наталья Николаевна  
**Бракович** Андрей Игоревич

**ИНФОРМАТИКА  
И КОМПЬЮТЕРНАЯ ГРАФИКА**

Курс лекций

Редактор *Е. С. Ватеичкина*  
Компьютерная верстка *О. В. Трусевич*

Подписано в печать 26.01.2010. Формат 60×84<sup>1</sup>/<sub>16</sub>.  
Бумага офсетная. Гарнитура Таймс. Печать офсетная.  
Усл. печ. л. 11,9. Уч.-изд. л. 12,2.  
Тираж 600 экз. Заказ 20.

Отпечатано в Центре издательско-полиграфических  
и информационных технологий учреждения образования  
«Белорусский государственный технологический университет».  
220006. Минск, Свердлова, 13а.  
ЛИ № 02330/0549423 от 08.04.2009.  
ЛП № 02330/0150477 от 16.01.2009.