

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ
ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Информатика

Направление подготовки (специальность) 35.03.06 «Агроинженерия»

Профиль образовательной программы Технический сервис в АПК

Форма обучения очная

СОДЕРЖАНИЕ

1. Конспект лекций	4
1.1 Лекция № 1 <i>Введение в предмет. Понятие информации.</i>	4
1.2 Лекция № 2 <i>Кодирование информации.</i>	6
1.3 Лекция № 3 <i>Измерение информации</i>	8
1.4 Лекция № 4 <i>Системы счисления.</i>	11
1.5 Лекция № 5 <i>Логические основы построения персонального компьютера</i>	13
1.6 Лекция № 6 <i>Технические средства реализации информационных процессов.</i>	15
1.7 Лекция № 7 <i>Персональный компьютер.</i>	18
1.8 Лекция № 8 <i>Понятие системного и служебного (сервисного) программного обеспечения: назначение, возможности, структура. Операционные системы. Файловая структура операционных систем. Операции с файлами...</i>	20
1.9 Лекция № 9 <i>Программные средства решения задач оформления управленческой документации</i>	23
1.10 Лекция № 10 <i>Графические редакторы</i>	25
1.11 Лекция № 11 <i>Электронные таблицы.</i>	34
1.12 Лекция № 12 <i>Обзор программ для решения аналитических задач.</i>	36
1.13 Лекция № 13 <i>Базы данных. Системы управления базами данных.</i>	38
1.14 Лекция № 14 <i>Модели решения функциональных и вычислительных задач</i>	42
1.15 Лекция № 15 <i>Алгоритмизация и программирование.</i>	46
1.16 Лекция № 16 <i>Локальные и глобальные сети ЭВМ.</i>	54
1.17 Лекция № 17 <i>Основы защиты информации и сведений.</i>	57
2. Методические указания по выполнению лабораторных работ	61
2.1 Лабораторная работа № ЛР-1 <i>Правила техники безопасности. Введение в предмет. Входной контроль</i>	61
2.2 Лабораторная работа № ЛР-2 <i>Способы сбора информации</i>	63
2.3 Лабораторная работа № ЛР-3 <i>Кодирование информации</i>	64
2.4 Лабораторная работа № ЛР-4 <i>Измерение информации</i>	65
2.5 Лабораторная работа № ЛР-5 <i>Информационно-логические основы построения персонального компьютера</i>	67
2.6 Лабораторная работа № ЛР-6 <i>Логические основы построения персонального компьютера</i>	71
2.7 Лабораторная работа № ЛР- 7 <i>Первоначальные сведения и правила работы в операционной системе Windows</i>	79

2.8 Лабораторная работа № ЛР-8 Работа с сервисными программами в операционной системе Windows	82
2.9 Лабораторная работа № ЛР-9 Возможности графического редактора Paint и текстового редактора WordPad.....	88
2.10 Лабораторная работа № ЛР-10-11 Текстовый редактор	90
2.11 Лабораторная работа № ЛР-12-13 Программные средства решения задач презентационного представления документации.....	98
2.12 Лабораторная работа № ЛР-14 Табличный процессор: работа с листами и графиками.....	101
2.13 Лабораторная работа № ЛР-15 Табличный процессор: операции с условием	113
2.14 Лабораторная работа № ЛР-16 Табличный процессор: работа с массивами	120
2.15 Лабораторная работа № ЛР-17 Специальные методы работы с программой Excel.....	125
2.16 Лабораторная работа № ЛР-18 Проектирование базы данных в СУБД MS Access	131
2.17 Лабораторная работа № ЛР-19 Создание таблиц и пользовательских форм для ввода данных в СУБД MS Access	139
2.18 Лабораторная работа № ЛР-20 Модификация таблиц и работа с данными с использованием запросов в СУБД MS Access	145
2.19 Лабораторная работа № ЛР-21 Работа с данными и создание отчетов в СУБД MS Access. Комплексная работа с объектами СУБД MS Access.....	150
2.20 Лабораторная работа № ЛР-22 Алгоритмизация математических задач	157
2.21 Лабораторная работа № ЛР-23 Элементы программирования на языке высокого уровня.....	159
2.22 Лабораторная работа № ЛР-24 Локальные и глобальные сети ЭВМ	163
2.23 Лабораторная работа № ЛР-25 Работа в глобальной сети Internet	167
2.24 Лабораторная работа № ЛР-26 Защита информации в компьютерах и сетях	176

1. КОНСПЕКТ ЛЕКЦИЙ

1. 1 Лекция №1 (2 часа).

Тема: «Введение в предмет. Понятие информации.»

1.1.1 Вопросы лекции:

1. Информатика как наука о технологии обработки информации.
2. Понятие и виды информации.
3. Свойства информации.

1.1.2 Краткое содержание вопросов:

1. Информатика как наука о технологии обработки информации

Информатика – самостоятельная научная дисциплина, предметом которой стали свойства информации, её поведение в техногенных, социальных и биологических системах, а также методы и технологии, ориентированные на сбор, обработку, хранение, передачу и распространение информации, или, кратко, информационные технологии.

Термин "информатика" (франц. informatique) происходит от французских слов information (информация) и automatique (автоматика) и дословно означает "информационная автоматика".

Информатика — комплексная научная дисциплина с широчайшим диапазоном применения. Её приоритетные направления:

- разработка вычислительных систем и программного обеспечения;
- теория информации, изучающая процессы, связанные с передачей, приёмом, преобразованием и хранением информации;
- математическое моделирование, методы вычислительной и прикладной математики и их применение к фундаментальным и прикладным исследованиям в различных областях знаний;
- методы искусственного интеллекта, моделирующие методы логического и аналитического мышления в интеллектуальной деятельности человека (логический вывод, обучение, понимание речи, визуальное восприятие, игры и др.);
- системный анализ, изучающий методологические средства, используемые для подготовки и обоснования решений по сложным проблемам различного характера;
- биоинформатика, изучающая информационные процессы в биологических системах;
- социальная информатика, изучающая процессы информатизации общества;
- методы машинной графики, анимации, средства мультимедиа;
- телекоммуникационные системы и сети, в том числе, глобальные компьютерные сети, объединяющие всё человечество в единое информационное сообщество;
- разнообразные приложения, охватывающие производство, науку, образование, медицину, торговлю, сельское хозяйство и все другие виды хозяйственной и общественной деятельности.

Роль информатики в развитии общества чрезвычайно велика. С ней связано начало революции в области накопления, передачи и обработки информации. Эта революция, следующая за революциями в овладении веществом и энергией, затрагивает и коренным образом преобразует не только сферу материального производства, но и интеллектуальную, духовную сферы жизни.

Информатика исследует следующие группы основных вопросов:

- технические, связанные с изучением методов и средств надежного сбора, хранения, передачи, обработки и выдачи информации;

- семантические, определяющие способы описания смысла информации, изучающие языки ее описания;
- прагматические, описывающие методы кодирования информации;
- синтаксические, связанные с решением задач по формализации и автоматизации некоторых видов научно-информационной деятельности, в частности индексирование, автоматическое реферирование, машинный перевод.

2. Понятие и виды информации

Понятие «информация» является ключевым понятием информатики, т.е. первичным и неопределяемым. Оно предполагает наличие материального носителя информации, источника информации, передатчика информации, приемника и канала связи между передатчиком и приемником. Термин «информация» происходит от латинского слова *informatio* - разъяснение, пояснение. Информация - это общенаучное понятие включающие, обмен сведениями между людьми. Человеком и автоматом, обмен сигналами в животном и растительном мире (передача признаков от клетки к клетке, от организма к организму). Более узкое определение дается в технике, где это понятие включает в себя все сведения, являющиеся объектом хранения, передачи и преобразования.

Экономическая информация - совокупность сведений, отражающих социально-экономические процессы и служащих для управления этими процессами и коллективами людей в производственной и непроизводственной сфере.

Все многообразие окружающей нас информации можно сгруппировать по различным признакам:

«Область возникновения»: элементарная или механическая - информация, отражающая процессы, явления неодушевленной природы, биологическая - процессы животного и растительного мира; социальная - информация, отражающая процессы человеческого общества

Способ передачи и восприятия. Информацию, передаваемую видимыми образами и символами, называют, визуальной, звуками - аудиальной, ощущениями - тактильной, запахами и вкусами - органолептической, информацию, выдаваемую и воспринимаемую средствами вычислительной техники, - машинной. Информацию, создаваемую и используемую человеком, по общественному назначению можно разбить на три вида: личная, массовая и специальная. По способу обработки электронными устройствами: числовая, текстовая, графическая, звуковую и видео (мультимедиа).

3. Свойства информации

1. Достоверность – свойство отражать реальные объекты с необходимой точностью (мера соответствия истине).

2. Полнота – характеризует качество информации и определяет достаточность данных для принятия решения (чем полнее данные, тем шире диапазон методов, которые можно использовать).

3. Актуальность – степень соответствия информации текущему моменту времени (прогноз погоды).

4. Объективность – мера отвлеченности от интересов определенного круга людей (не зависит от человеческого фактора).

5. Доступность – мера возможности получить ту или иную информацию.

Адекватность – уровень соответствия создаваемого с помощью информации образа реальному объекту, процессу, явлению (это свойство зависит от всех предыдущих).

Адекватность может выражаться в трех формах:

- синтаксическая (отражает формально-структурные характеристики и не затрагивает смысловое содержание информации). В этом случае информацию называют данными.

- семантическая форма (отражает смысловое содержание информации). На этом уровне анализируются сведения, предоставляемые информацией, рассматриваются ее смысловые связи.

- прагматический аспект (отражает потребительскую сторону информации, связан с ценностью, полезностью информации для достижения цели).).

1.2 Лекция №2 (2 часа).

Тема: «Кодирование информации.»

1.2.1 Вопросы лекции:

1. Понятие кодирования информации.
2. Кодирование чисел.
3. Кодирование текстовой, графической и звуковой информации.

1.2.2 Краткое содержание вопросов:

1 Понятие кодирования информации.

Для автоматизации работы с данными, относящимися к различным типам очень важно унифицировать их форму представления – для этого обычно используется приём **кодирования, т.е. выражение данных одного типа через данные другого типа.**

Для автоматизации работы с данными, относящимися к различным типам очень важно унифицировать их форму представления – для этого обычно используется приём **кодирования, т.е. выражение данных одного типа через данные другого типа.**

Код – это система условных знаков для предоставления информации.

Кодирование – это операция преобразования символов или группы символов одного кода в символы или группы символов другого кода.

Человек кодирует информацию с помощью языка.

Язык – это знаковая форма представления информации.

Виды языков: а) естественный (язык, на котором разговаривает та или иная народность, носит национальный характер); б) формальный (язык математической логики, язык программирования Паскаль или Бейсик).

информация	Естественный язык	Формальный язык
Нахождение гипотенузы прямоугольного треугольника	Гипотенуза равна корню квадратному из суммы квадратов катетов	$c = \sqrt{a^2 + b^2}$
Призыв о помощи	Помогите, спасите	SOS

Одну и ту же информацию можно кодировать разными способами, что зависит от источника, носителя, потребителя информации.

Обычно при кодировании информации используется средство кодирования – некоторая таблица, которая устанавливает соответствие между знаками разных кодов. Например, для кода Морзе есть специальная таблица, в которой указаны символы алфавита и их представление в виде набора точек и тире.

Закодировать слово файл несколькими способами.

В процессе обмена информацией происходит кодирование и декодирование информации:

Источник – кодирующее устройство – передача инф – декодирующее устройство – потребитель информации.

Огромное количество способов кодирования информации привело к попытке создать универсальный язык или азбуку кодирования. Система кодирования в вычислительной технике – двоичное кодирование.

Все виды информации кодируются в последовательности электрических импульсов: есть (1), нет (0), т. е. в последовательности 0 и 1. В двоичной форме представления информации знаки 1 и 0 называют БИТАМИ. Такое кодирование называется двоичным, а логические последовательности 0 и 1 – машинным языком. Благодаря введению понятия единицы информации появилась возможность определения размера любой информации числом битов.

Одним битом могут быть выражены два понятия: 0 или 1 (да или нет, чёрное или белое, истина или ложь и т.п.). Если количество битов увеличить до двух, то уже можно выразить четыре различных понятия. Тремя битами можно закодировать восемь различных значений. Общая формула: $N = 2^m$, где m – количество бит, N – количество значений информации.

2. Кодирование числовой

Кодирование числовой информации. Кодирование целых и действительных чисел

Целые числа кодируются двоичным кодом достаточно просто - необходимо взять целое число и делить его пополам до тех пор, пока частное не будет равно единице. Совокупность остатков от каждого деления, записанная справа налево вместе с последним частным, и образует двоичный аналог десятичного числа.

Для кодирования целых чисел от 0 до 255 достаточно иметь 8 разрядов двоичного кода (8 бит). 16 бит позволяют закодировать целые числа от 0 до 65535, а 24 – уже более 16,5 миллионов различных значений.

Естественная и экспоненциальная запись числа

Для кодирования действительных чисел используют 80-разрядное кодирование.

При этом число предварительно преобразовывают в нормализованную (экспоненциальную) форму:

$0,001 = 0,1 * 10^{-3}$ – запись в 10-ичной СС.

ест. экспон.

$A_q = m * q^n$ – число в любой СС в экспоненциальной форме.

m – мантисса числа (дробь, им после запятой цифру $\neq 0$)

q – основание СС

n – порядок числа.

Первая часть числа называется мантиссой, а вторая – характеристикой. Большую часть из 80 бит отводят для хранения мантиссы (вместе со знаком) и некоторое фиксированное количество разрядов отводят для хранения характеристики.

3. Кодирование текстовой, графической и звуковой информации

Кодирование текстовой информации. Компьютерный алфавит состоит из 256 символов: все символы английского и русского языков, как строчные, так и прописные, а также знаки препинания, символы основных арифметических действий и некоторые общепринятые специальные символы.

Если рассматривать символы как возможные события, то по формуле (1) $256 = 2^8$, т.е. для кодирования 1 символа требуется 8 бит или 1 байт информации. Кодирование заключается в том, что каждому символу компьютерного алфавита ставится в соответствие уникальный десятичный код от 0 до 255 и соответствующий ему двоичный код от 00000000_2 до 11111111_2 .

Для английского языка, захватившего де-факто нишу международного средства общения, противоречия уже сняты. Институт стандартизации США ввёл в действие систему кодирования ASCII (American Standard Code for Information Interchange – стандартный код информационного обмена США). В системе ASCII закреплены две таблицы кодирования базовая и расширенная. Базовая таблица закрепляет значения кодов от 0 до 127, а расширенная относится к символам с номерами от 128 до 255.

Кодировка символов русского языка, известная как кодировка Windows-1251, была введена “извне” - компанией Microsoft, но, учитывая широкое распространение операционных систем и других продуктов этой компании в России, она глубоко закрепились и нашла широкое распространение. Другая распространённая кодировка носит название КОИ-8 (код обмена информацией, восьмизначный) – её происхождение относится к временам действия Совета Экономической Взаимопомощи государств Восточной Европы. Сегодня кодировка КОИ – 8 имеет широкое распространение в компьютерных сетях на территории России и в российском секторе Интернета.

Международный стандарт, в котором предусмотрена кодировка символов русского языка, носит названия ISO (International Standard Organization – Международный институт стандартизации). На практике данная кодировка используется редко.

Система, основанная на 16-разрядном кодировании символов, получила название универсальной – UNICODE. Шестнадцать разрядов позволяют обеспечить уникальные коды для 65 536 различных символов – этого поля вполне достаточно для размещения в одной таблице символов большинства языков планеты.

Кодирование графической информации. Графическая информация на экране монитора представляется в виде растрового изображения, которое формируется из определенного количества строк, которые, в свою очередь, содержат определенное количество точек (пикселей).

Поскольку линейные координаты и индивидуальные свойства каждой точки (яркость) можно выразить с помощью целых чисел, то можно сказать, что растровое кодирование позволяет использовать двоичный код для представления графических данных. Общепринятым на сегодняшний день считается представление чёрно-белых иллюстраций в виде комбинации точек с 256 градациями серого цвета, и, таким образом, для кодирования яркости любой точки обычно достаточно восьмиразрядного двоичного числа.

Для кодирования цветных графических изображений применяется принцип декомпозиции произвольного цвета на основные составляющие. В качестве таких составляющих используют три основных цвета: красный (Red), зелёный (Green) и синий (Blue). На практике считается, что любой цвет, видимый человеческим глазом, можно получить механическим смешением этих трёх цветов. Такая система кодирования получила название RGB по первым буквам основных цветов.

Кодирование звуковой информации. Метод FM (Frequency Modulation) основан на том, что теоретически любой сложный звук можно разложить на последовательность простейших гармонических сигналов разных частот, каждый из которых представляет собой правильную синусоиду, а, следовательно, может быть описан числовыми параметрами, т.е. кодом. В природе звуковые сигналы имеют непрерывный спектр, т.е. являются аналоговыми. Их разложение в гармонические ряды и представление в виде дискретных цифровых сигналов выполняют специальные устройства – аналогово-цифровые преобразователи (АЦП). Обратное преобразование для воспроизведения звука, закодированного числовым кодом, выполняют цифро-аналоговые преобразователи (ЦАП).

Метод таблично волнового (Wave-Table) синтеза лучше соответствует современному уровню развития техники. В заранее подготовленных таблицах хранятся образцы звуков для множества различных музыкальных инструментов. В технике такие образцы называют сэмплами. Числовые коды выражают тип инструмента, номер его модели, высоту тона, продолжительность и интенсивность звука, динамику его изменения, некоторые параметры среды, в которой происходит звучание, а также прочие параметры, характеризующие особенности звучания. Поскольку в качестве образцов исполняются реальные звуки, то его качество получается очень высоким и приближается к качеству звучания реальных музыкальных инструментов.

1.3 Лекция №3 (2 часа).

Тема: «Измерение информации.»

1.3.1 Вопросы лекции:

1. Меры информации. Классификация мер.
2. Количество информации.
3. Качество информации.

1.3.2 Краткое содержание вопросов:

1. Меры информации. Классификация мер

Для измерения информации вводятся два параметра: количество информации I и объем данных V_d .

Каждой форме адекватности соответствует своя мера количества информации и объема данных.

Синтаксическая мера информации

Эта мера количества информации оперирует с обезличенной информацией, не выражающей смыслового отношения к объекту.

Объем данных V_d в сообщении измеряется количеством символов (разрядов) в этом сообщении. В различных системах счисления один разряд имеет различный вес и соответственно меняется единица измерения данных:

в двоичной системе счисления единица измерения - бит (*bit — binary digit*— двоичный разряд);

в десятичной системе счисления единица измерения - дит (десятичный разряд).

Семантическая мера информации. Для измерения смыслового содержания информации, т.е. ее количества на семантическом уровне, наибольшее признание получила тезаурусная мера, которая связывает семантические свойства информации со способностью пользователя принимать поступившее сообщение. Для этого используется понятие *тезаурус пользователя*.

Тезаурус - это совокупность сведений, которыми располагает пользователь или система.

В зависимости от соотношений между смысловым содержанием информации S и тезаурусом пользователя S_p изменяется количество семантической информации I_C , воспринимаемой пользователем и включаемой им в дальнейшем в свой тезаурус.

Максимальное количество семантической информации I_C потребитель приобретает при согласовании ее смыслового содержания S со своим тезаурусом S_p ($S_p = S_{p\text{opt}}$), когда поступающая информация понятна пользователю и несет ему ранее не известные (отсутствующие в его тезаурусе) сведения.

Прагматическая мера информации. Эта мера определяет полезность информации (ценность) для достижения пользователем поставленной цели. Эта мера также величина относительная, обусловленная особенностями использования этой информации в той или иной системе. Ценность информации целесообразно измерять в тех же самых единицах (или близких к ним), в которых измеряется целевая функция.

2.Количество информации.

Количество информации I на синтаксическом уровне невозможно определить без рассмотрения понятия неопределенности состояния системы (энтропии системы).

Количеством информации называют числовую характеристику сигнала, отражающую ту степень неопределенности (неполноту знаний), которая исчезает после получения сообщения в виде данного сигнала. Эту меру неопределенности в теории информации называют энтропией. Если в результате получения сообщения достигается полная ясность в каком-то вопросе, говорят, что была получена полная или исчерпывающая информация и необходимости в получении дополнительной информации

нет. И, наоборот, если после получения сообщения неопределенность осталась прежней, значит, информации получено не было (нулевая информация).

Бит - минимальная единица количества информации, ибо получить информацию меньшую, чем 1 бит, невозможно.

Связь между количеством информации и числом состояний системы устанавливается формулой Хартли:

$$I = \log_2 N - \text{формула Хартли} \quad (1)$$

где I - количество информации в битах, величина измеряющая неопределённость, ЭНТРОПИЯ;

N - число возможных состояний.

Ту же формулу можно представить иначе:

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

Группа из 8 битов информации называется байтом. Если бит - минимальная единица информации, то байт ее основная единица. Существуют производные единицы информации: килобайт (КБайт, Кб), мегабайт (Мбайт, Мб) и гигабайт (Гбайт, Гб).

1 Кб = 1024 байта = 2^{10} (1024) байтов.

1 Мб = 1024 Кбайта = 2^{20} байт.

1 Гб = 1024 Мбайта = 2^{30} байт.

1 Терабайт (Тбайт) = 1024 Гбайт = 2^{40} байт,

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

3. Качество информации

Возможность и эффективность использования информации обуславливаются такими основными ее потребительскими показателями качества, как репрезентативность, содержательность, достаточность, доступность, актуальность, своевременность, точность, достоверность, устойчивость.

Репрезентативность информации связана с правильностью ее отбора и формирования в целях адекватного отражения свойств объекта.

Содержательность информации отражает семантическую емкость, равную отношению количества семантической информации в сообщении к объему обрабатываемых данных.

Достаточность (полнота) информации означает, что она содержит минимальный, но достаточный для принятия правильного решения состав (набор показателей).

Доступность информации восприятию пользователя обеспечивается выполнением соответствующих процедур ее получения и преобразования.

Актуальность информации определяется степенью сохранения ценности информации для управления в момент ее использования и зависит от динамики изменения ее характеристик и от интервала времени, прошедшего с момента возникновения данной информации.

Своевременность информации означает ее поступление не позже заранее назначенного момента времени, согласованного со временем решения поставленной задачи.

Точность информации определяется степенью близости получаемой информации к реальному состоянию объекта, процесса, явления и т.п..

Достоверность информации определяется ее свойством отражать реально существующие объекты с необходимой точностью.

Устойчивость информации отражает ее способность реагировать на изменения исходных данных без нарушения необходимой точности.

1.4 Лекция №4 (2 часа).

Тема: «Системы счисления»

1.4.1 Вопросы лекции:

1. Понятие позиционной и непозиционной систем счисления.
2. Перевод целых чисел из одной системы счисления в другую
3. Арифметические операции в позиционных системах счисления

1.4.2 Краткое содержание вопросов:

1. Понятие позиционной и непозиционной систем счисления.

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

Все системы счисления делятся на две большие группы: позиционные и непозиционные. В непозиционных системах счисления количественный эквивалент каждой цифры не зависит от ее положения (места, позиции) в записи числа. В позиционных системах счисления количественный эквивалент (значение) цифры зависит от ее места (позиции) в записи числа.

Развернутая форма числа. В позиционной системе счисления любое вещественное число в развернутой форме может быть представлено в следующем виде:

$$A_q = \pm (a_{n-1}q^{n-1} + a_{n-2}q^{n-2} + \dots + a_0q^0 + a_{-1}q^{-1} + a_{-2}q^{-2} + \dots + a_{-m}q^{-m}) \quad (3)$$

Или

$$A_q = \pm \sum_{i=-m}^{n-1} a_i q^i \quad (3.1)$$

где A — само число,

q — основание системы счисления,

a_i — цифры, принадлежащие алфавиту данной системы счисления,

n — число целых разрядов числа,

m — число дробных разрядов числа.

Свернутой формой записи числа называется запись в виде:

$$A = a_{n-1}a_{n-2} \dots a_1a_0, a_{-1} \dots a_{-m} \quad (3.2)$$

Двоичная система счисления.

Основание $q=2$. В этом случае формула (3) принимает вид:

$$A_2 = \pm (a_{n-1}2^{n-1} + a_{n-2}2^{n-2} + \dots + a_02^0 + a_{-1}2^{-1} + a_{-2}2^{-2} + \dots + a_{-m}2^{-m}) \quad (3.3)$$

где a_i — возможные цифры (0, 1).

Восьмеричная система счисления. Основание: $q=8$. Алфавит: 0, 1, 2, 3, 4, 5, 6, 7.

Шестнадцатеричная система счисления

Основание: $q=16$. Алфавит: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

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

2. Перевод целых чисел из одной системы счисления в другую

Можно сформулировать алгоритм перевода целых чисел из системы с основанием p в систему с основанием q :

1. Основание новой системы счисления выразить цифрами исходной системы счисления и все последующие действия производить в исходной системе счисления.

2. Последовательно выполнять деление данного числа и получаемых целых частных на основание новой системы счисления до тех пор, пока не получим частное, меньшее делителя.

3. Полученные остатки, являющиеся цифрами числа в новой системе счисления, привести в соответствие с алфавитом новой системы счисления.

4. Составить число в новой системе счисления, записывая его, начиная с последнего остатка.

Перевод дробных чисел из одной системы счисления в другую

Алгоритм перевода правильной дроби с основанием p в дробь с основанием q :

1. Основание новой системы счисления выразить цифрами исходной системы счисления и все последующие действия производить в исходной системе счисления.

2. Последовательно умножать данное число и получаемые дробные части произведений на основание новой системы до тех пор, пока дробная часть произведения не станет равной нулю или будет достигнута требуемая точность представления числа.

3. Полученные целые части произведений, являющиеся цифрами числа в новой системе счисления, привести в соответствие с алфавитом новой системы счисления.

4. Составить дробную часть числа в новой системе счисления, начиная с целой части первого произведения.

Перевод произвольных чисел

Перевод произвольных чисел, т.е. чисел, содержащих целую и дробную части, осуществляется в два этапа. Отдельно переводится целая часть, отдельно — дробная. В итоговой записи полученного числа целая часть отделяется от дробной запятой (точкой).

Перевод чисел из системы счисления с основанием 2 в систему счисления с основанием 2^n и обратно. Если основание q -ичной системы счисления является степенью числа 2, то перевод чисел из q -ичной системы счисления в 2-ичную и обратно можно проводить по более простым правилам. Для того, чтобы целое двоичное число записать в системе счисления с основанием $q=2^n$, нужно:

1. Двоичное число разбить справа налево на группы по n цифр в каждой.

2. Если в последней левой группе окажется меньше n разрядов, то ее надо дополнить слева нулями до нужного числа разрядов.

3. Рассмотреть каждую группу как n -разрядное двоичное число и записать ее соответствующей цифрой в системе счисления с основанием $q=2^n$.

Перевод дробных чисел. Для того, чтобы дробное двоичное число записать в системе счисления с основанием $q=2^n$, нужно:

1. Двоичное число разбить слева направо на группы по n цифр в каждой.

2. Если в последней правой группе окажется меньше n разрядов, то ее надо дополнить справа нулями до нужного числа разрядов.

Перевод произвольных чисел. Для того, чтобы произвольное двоичное число записать в системе счисления с основанием $q=2^n$, нужно:

1. Целую часть данного двоичного числа разбить справа налево, а дробную — слева направо на группы по n цифр в каждой.

2. Если в последних левой и/или правой группах окажется меньше n разрядов, то их надо дополнить слева и/или справа нулями до нужного числа разрядов;

3. Рассмотреть каждую группу как n -разрядное двоичное число и записать ее соответствующей цифрой в системе счисления с основанием $q=2^n$.

3. Арифметические операции в позиционных системах счисления

При сложении цифры суммируются по разрядам, и если при этом возникает избыток, то он переносится влево.

Выполняя умножение многозначных чисел в различных позиционных системах счисления, можно использовать обычный алгоритм перемножения чисел в столбик, но при этом результаты перемножения и сложения однозначных чисел необходимо заимствовать из соответствующих рассматриваемой системе таблиц умножения и сложения.

1.5 Лекция №5 (2 часа).

Тема: «Логические основы построения персонального компьютера»

1.5.1 Вопросы лекции:

1. Основные понятия логики
2. Логические операции. Таблицы истинности
3. Связь между алгеброй логики и двоичным кодированием
4. Данные и команды в памяти компьютера и в регистрах процессора

1.5.2 Краткое содержание вопросов:

1. Основные понятия логики

Логика изучает внутреннюю структуру процесса мышления, который реализуется в таких естественно сложившихся формах как понятие, суждение, умозаключение и доказательство.

Понятие — это форма мышления, отражающая наиболее существенные свойства предмета, отличающие его от других предметов. Высказывание (суждение) — это форма мышления, выраженная с помощью понятий, посредством которой что-либо утверждают или отрицают о предметах, их свойствах и отношениях между ними. Умозаключение — это форма мышления, посредством которой из одного или нескольких суждений, называемых посылками, по определенным правилам логического вывода получается новое знание о предметах реального мира (вывод). Доказательство есть мыслительный процесс, направленный на подтверждение или опровержение какого-либо положения посредством других несомненных, ранее обоснованных доводов.

2. Логические операции. Таблицы истинности

Логическая операция КОНЪЮНКЦИЯ (логическое умножение).

Конъюнкция (&) - это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся истинным тогда и только тогда, когда оба исходных высказывания истинны.

Логическая операция ДИЗЪЮНКЦИЯ (логическое сложение). Дизъюнкция (\vee) - это логическая операция, которая каждому двум простым высказываниям ставит в соответствие составное высказывание, являющееся ложным тогда и только тогда, когда оба исходных высказывания ложны и истинным, когда хотя бы одно из двух образующих его высказываний истинно.

Логическая операция ИНВЕРСИЯ (отрицание). Отрицание (\bar{A}) - это логическая операция, которая каждому простому высказыванию ставит в соответствие составное высказывание, заключающееся в том, что исходное высказывание отрицается. В алгебре множеств логическому отрицанию соответствует операция *дополнения до универсального множества*, т.е. множеству получившемуся в результате отрицания множества A соответствует множество \bar{A} , дополняющее его до универсального множества.

Логическая операция ИМПЛИКАЦИЯ (логическое следование).

Импликация (\Rightarrow) - это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся ложным тогда и только тогда, когда условие (первое высказывание) истинно, а следствие (второе высказывание) ложно.

Логическая операция ЭКВИВАЛЕНЦИЯ (равнозначность). Эквиваленция (\Leftrightarrow) — это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся истинным тогда и только тогда, когда оба исходных высказывания одновременно истинны или одновременно ложны.

Таблицу, показывающую, какие значения принимает составное высказывание при всех сочетаниях (наборах) значений входящих в него простых высказываний, называют

таблицей истинности составного высказывания. Составные высказывания в алгебре логики записываются с помощью логических выражений. Для любого логического выражения достаточно просто построить таблицу истинности

Алгоритм построения таблицы истинности:

- 1) подсчитать количество переменных n в логическом выражении;
- 2) определить число строк в таблице, которое равно $m = 2^n$;
- 3) подсчитать количество логических операций в логическом выражении и определить количество столбцов в таблице, которое равно количеству переменных плюс количество операций;
- 4) ввести названия столбцов таблицы в соответствии с последовательностью выполнения логических операций с учетом скобок и приоритетов;
- 5) заполнить столбцы входных переменных наборами значений;
- 6) провести заполнение таблицы истинности по столбцам, выполняя логические операции в соответствии с установленной в п.4 последовательностью.

3. Связь между алгеброй логики и двоичным кодированием

Математический аппарат алгебры логики очень удобен для описания того, как функционируют аппаратные средства компьютера, поскольку основной системой счисления в компьютере является двоичная, в которой используются цифры 1 и 0, а значений логических переменных тоже два: “1” и “0”.

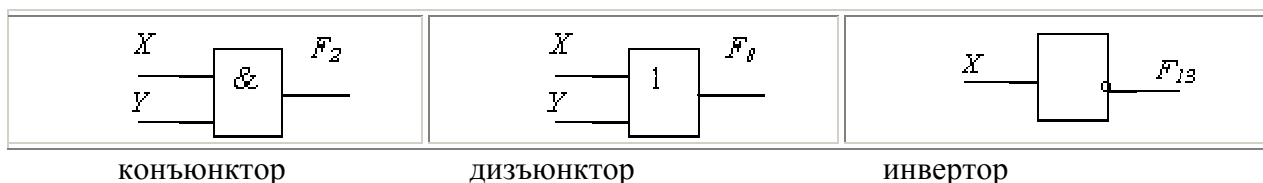
Из этого следует два вывода: 1) одни и те же устройства компьютера могут применяться для обработки и хранения как числовой информации, представленной в двоичной системе счисления, так и логических переменных; 2) на этапе конструирования аппаратных средств алгебра логики позволяет значительно упростить логические функции, описывающие функционирование схем компьютера, и, следовательно, уменьшить число элементарных логических элементов, из десятков тысяч которых состоят основные узлы компьютера.

4. Данные и команды в памяти компьютера и в регистрах процессора

Данные и команды представляются в виде двоичных последовательностей различной структуры и длины. Логический элемент компьютера — это часть электронной логической схемы, которая реализует элементарную логическую функцию. Логическими элементами компьютеров являются электронные схемы И, ИЛИ, НЕ, И—НЕ, ИЛИ—НЕ и другие (называемые также вентилями), а также триггер.

С помощью этих схем можно реализовать любую логическую функцию, описывающую работу устройств компьютера. Работу логических элементов описывают с помощью таблиц истинности.

Ниже приведены условные обозначения (схемы) базовых логических элементов (См. Рис.4.1), реализующих логическое умножение (конъюнктор), логическое сложение (дизъюнктор) и отрицание (инвертор).



Р и с у н о к **Ошибка! Текст указанного стиля в документе отсутствует.** 1 -- Базовые логические элементы

Устройства компьютера (сумматоры в процессоре, ячейки памяти в оперативной памяти и др.) строятся на основе базовых логических элементов.

1.6 Лекция №6 (2 часа).

Тема: «Технические средства реализации информационных процессов»

1.6.1 Вопросы лекции:

1. Принципы фон Неймана.
2. Классификация ЭВМ.
3. Поколения ЭВМ и перспективы их развития.
4. Архитектура ЭВМ.

1.6.2 Краткое содержание вопросов:

1. Принципы фон Неймана.

В 1946 году Нейман на основе критического анализа конструкции ENIAC предложил ряд новых идей организации ЭВМ, в том числе концепцию хранимой программы, он предложил записывать и хранить в памяти алгоритм вычислений вместе с данными. Принципы Дж.фон Неймана показались вначале простыми и очевидными и лишь в дальнейшем они приобрели статус фундаментальных положений, надолго определивших направление развития вычислительной техники. В результате реализации идей фон Неймана была создана архитектура ЭВМ, во многих чертах сохранившаяся до настоящего времени.

В отчете "Предварительное обсуждение логического конструирования электронного вычислительного устройства" Дж. фон Нейман опубликовал основные принципы, которые заключались в следующем:

1. Компьютеры на электронных элементах должны работать не в десятичной, а в двоичной системе счисления.
2. Компьютер управляется программой, составленной из отдельных шагов - команд. Программа должна размещаться в одном из блоков компьютера - в запоминающем устройстве, обладающем достаточной емкостью и скоростью выборки команд.
3. Команды, так же как и числа, с которыми оперирует компьютер, записываются в двоичном коде. Это обстоятельство приводит к следующим важным последствиям:
 - а) промежуточные результаты вычислений, константы и другие числа могут размещаться в том же запоминающем устройстве, что и программа;
 - б) числовая форма записи программы позволяет производить операции над величинами, которыми закодированы команды программы;
 - в) появляется возможность перехода в процессе вычислений на тот или иной участок программы в зависимости от результатов вычислений, условных переходов.
4. Трудности физической реализации запоминающего устройства, быстроедействие которого соответствует скорости работы логических схем требует иерархической организации памяти.
5. Арифметическое устройство конструируется на основе схем, выполняющих операцию сложения - создание специальных устройств для выполнения других операций нецелесообразно.
6. Необходимо использовать параллельный принцип организации вычислительного процесса (операции над словами производятся одновременно во всех разрядах слова)

Принцип использования двоичной системы счисления расширил набор физических приборов и явлений, которые можно использовать для представления информации в операционных и запоминающих устройствах компьютера. Две цифры для отображения "1" и "0" могут отображаться состоянием любой двухстабильной системы. Например,

открытое и закрытое состояние электронного ключа (ламповой схемы), два состояния триггера, намагниченным или не намагниченным состоянием ферромагнитной поверхности. Ну, а в настоящее время набор электронных приборов и физических явлений, позволяющих получить два состояния для записи и обработки информации стал намного шире, но об этом поговорим позже. В двоичной системе счисления возможно построение логических схем и реализация функций алгебры логики или Булевой алгебры.

Принцип хранимой в памяти программы, представленной в двоичном коде, позволяет производить не только вычисления, направляя команду в устройство управления, а данные в арифметическое устройство, но и преобразовывать сами команды, например в зависимости от результатов вычислений, используя для преобразования коды команд и оперируя с ними, как с данными.

Принцип реализации условных переходов позволяет осуществлять программы с циклическими вычислениями с автоматическим выходом из цикла. Благодаря принципу условного перехода сокращается число команд, в программе, так как не требуется повторять одинаковые участки программы.

Принцип иерархической организации памяти был сформулирован в связи с тем, что с самого первого компьютера с сохраняемой программой существовало несоответствие между быстродействием арифметического устройства и оперативной памяти. Противоречия бы не существовало, если выполнить память на тех же элементах, что и арифметическое устройство, но такая память получалась слишком дорогой, кроме того, непомерно увеличивалось количество радиоламп, что заметно снижало надежность компьютера. Иерархическое построение оперативного запоминающего устройства позволяет иметь быстродействующую память небольшого объема только для данных и команд, подготовленных к выполнению. Все остальное хранится в запоминающем устройстве более низкого уровня, для этого стали использоваться появившиеся вскоре магнитные носители информации.

Параллельный принцип организации вычислений позволяет значительно увеличить скорость вычислений, хотя это и приводит к более значительным затратам оборудования.

2. Классификация ЭВМ.

Опр 10.3.1 ЭВМ – это программируемое электронное устройство обработки и накопления информации.

По своим размерам, быстродействию (т.е. количеству выполняемых операций в секунду), объему памяти современные ЭВМ делятся на следующие классы:

- СуперЭВМ (типа Cray и Эльбрус);
- Большие ЭВМ (типа ЕС-1066);
- Средние ЭВМ (типа ЕС-1022, IBM-360);
- Мини-ЭВМ (типа Eclipse, PDP (СМ-ЭВМ));
- МикроЭВМ или персональные компьютеры (типа “Электронника-60”, Celeron 650, Pentium III).

Микро ЭВМ или персональные компьютеры используются для автоматизации отдельных рабочих мест, для обработки деловой информации, в быту и т.д.

Мини ЭВМ имеют большое количество дополнительных устройств – это средства для автоматизации различных технологических процессов, научных исследований, проектно-конструкторских работ и т.д.

Средние и большие ЭВМ используются для решения задач управления производством, имеют большой объем памяти.

СуперЭВМ применяются в решении сложнейших народнохозяйственных задач и проведении научных экспериментов (космические проекты, геофизические исследования и т.д.).

3. Поколения ЭВМ.

С середины 50-х годов начался бурный рост вычислительной техники. Каждый этап развития ЭВМ определялся тем материалом, из которого они изготовлялись, т.е. элементной базой. Поэтому принято говорить о поколениях ЭВМ. Можно выделить 4 поколения ЭВМ:

- 1) 1946-середина 50 х годов. Элементная база – электронно-вакуумные лампы, быстродействие 10-20 тысяч операций (сложений и сдвигов) в секунду. Примеры: ЭНИАК, ЭДСАК, БЭСМ, МЭСМ, УРАЛ (1954 г.), М-20 (1957 г.), “Стрела”, “Минск-1”.
- 2) Середина 50 х до середины 60 х. Элементная база – полупроводниковые элементы (транзисторы, диоды), быстродействие – 100000-1000000 операций в секунду. Примеры: RCA-501 (США 1959), Минск-22, Минск-32, М-220, Мир, Наири и др.
- 3) Середина 60 х – середина 70 х. Элементная база – интегральные схемы среднего уровня интеграции (ИС), быстродействие – сотни тысяч операций в секунду. Примеры: IBM-360 (США 1964), Урал-12, ЕС ЭВМ (машины единой серии), СМ ЭВМ (средние машины).
- 4) Середина 70 х до наших дней. Элементная база – большие интегральные схемы. Используются микропроцессоры (т.е. процессоры, размещенные на одном кристалле). Примеры: Эльбрус-2, М-10 (это большие ЭВМ), IBM PC 286, 386,486, Pentium (это персональные ЭВМ) и др. Сейчас разрабатываются ЭВМ на сверх больших интегральных схемах.

4. Архитектура ЭВМ

Однако какой бы из современных компьютеров мы не рассматривали – от самых маленьких до самых мощных, почти все они имеют общую принципиальную схему (архитектуру) (Рис. 43).

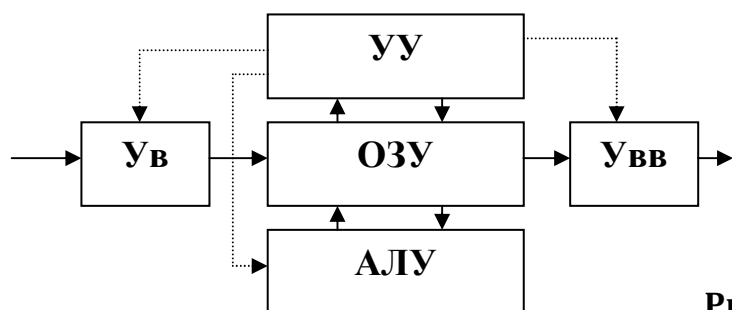


Рис. 43

Схема была предложена в 1946 году Джоном фон Нейманом (США). Т.о., в состав ЭВМ входят 5 основных устройств:

- Устройство ввода служит для ввода исходной информации в ЭВМ
- Устройство вывода служит для вывода результатов вычислений из памяти ЭВМ.
- АЛУ служит для выполнения арифметических и логических операций над числами;
- УУ служит для управления работой всего компьютера в процессе вычислений
- ОЗУ служит для приема, хранения и выдачи чисел

Основной принцип работы ЭВМ был предложен также Джоном фон Нейманом и получил название принципа программного управления.

Опр 10.3.2 Принцип действия ЭВМ, согласно которому переработка машиной исходных данных в конечный результат производится в соответствии с заранее составленной и введенной в машину программой называется *принципом программного управления*.

1.7 Лекция №7 (2 часа).

Тема: «Персональный компьютер»

1.7.1 Вопросы лекции:

1. Магистрально-модульный принцип построения компьютера
2. Процессор. Характеристики процессора.
3. Память компьютера, виды памяти.
4. Форматирование дисков.

1.7.2 Краткое содержание вопросов:

1. Магистрально-модульный принцип построения компьютера

Персональный компьютер – это комплекс взаимосвязанных устройств, каждое из которых выполняет определенную функцию. В основу архитектуры современных персональных компьютеров положен магистрально-модульный принцип. Модульный принцип позволяет потребителю самому комплектовать нужную ему конфигурацию компьютера и производить при необходимости ее модернизацию. Модульная организация компьютера опирается на магистральный (шинный) принцип обмена информацией между модулями (См. Рис.5.1).



Р и с у н о к **Ошибка! Текст указанного стиля в документе отсутствует.**2 -- Магистральный (шинный) принцип обмена информацией между модулями

- Для связи основных устройств компьютера между собой используется специальная информационная магистраль, обычно называемая инженерами шиной. Шина состоит из трех частей:

- шина адреса, на которой устанавливается адрес требуемой ячейки памяти или устройства, с которым будет происходить обмен информацией;
- шина данных, по которой собственно и будет передана необходимая информация; и, наконец,
- шина управления, регулирующей этот процесс (например, один из сигналов на этой шине позволяет компьютеру различать между собой адреса памяти и устройств ввода/вывода).

Интерфейс — это средство сопряжения двух устройств, в котором все физические и логические параметры согласуются между собой.

2. Процессор. Характеристики процессора

Процессор определяет производительность, эффективность всей вычислительной системы, он регулирует, управляет и контролирует рабочий процесс.

В современных персональных компьютерах центральный процессор конструктивно выполнен как микропроцессор на базе сверхбольшой интегральной схемы (СБИС). Это полупроводниковый кристалл или комплект кристаллов, на которых реализуются компоненты процессора. Логически центральный процессор представляет собой совокупность арифметико-логического устройства (АЛУ) и центрального устройства управления (УУ).

Характеристики процессора:

1. Тактовая частота – определяет его быстродействие. Тактовая частота – число вырабатываемых за 1 сек импульсов, синхронизирующих работу узлов компьютера. От нее, в частности зависит количество операций, которые производит процессор в одну секунду т.е скорость работы компьютера. На выполнение процессором каждой операции отводится определенное количество тактов. Тактовая частота измеряется в мегагерцах – МГц. Частота в 1 МГц соответствует 1 миллиону тактов в секунду.

2. Разрядность – максимальная длина двоичного кода, который может обрабатываться или передаваться процессором целиком. Разрядность связана с размером специальных ячеек памяти, которые находятся в самом процессоре. Они называются *регистрами*. Каждый процессор способен выполнять вполне определенный набор универсальных инструкций, называемых чаще всего машинными командами. Работа ЭВМ состоит в выполнении последовательности таких команд, подготовленных в виде программы.

3. Память компьютера. Виды памяти

Память компьютера предназначена для хранения информации. В компьютере имеются два вида памяти: внутренняя и внешняя.

а) Внутренняя память:

- специальная память (постоянное запоминающее устройство (ПЗУ));
- оперативное запоминающее устройство (ОЗУ);
- кэш-память.

В постоянной памяти или ПЗУ (по-английски Read-Only Memory- ROM что означает «память только для чтения») хранится программа BIOS (Basic Input/Output System), что переводится на русский язык, как Базовая система ввода-вывода. Эта программа обеспечивает при включении компьютера тестирование его основных узлов и загрузку операционной системы.

Оперативная память – это массив кристаллических ячеек, способных хранить данные. Каждая ячейка памяти имеет свой адрес, который выражается числом. Одна адресуемая ячейка содержит восемь двоичных ячеек, в которых можно хранить 8 бит (1 байт) данных. Любая программа, с которой мы собираемся работать, записывается или как говорят «загружается» в оперативную память, и в памяти хранятся все данные и результаты вычислений, которые производятся процессором во время выполнения программы. Занесение информации в память, а также извлечение ее из памяти производится по адресам. Это свойство памяти называется адресуемостью

Машинное слово. Наибольшую последовательность бит, которую процессор может обрабатывать как единое целое, называют машинным словом.

Кэш-память- это самая быстродействующая память, используемая микропроцессором для выполнения промежуточных действий и хранения данных, которые могут потребоваться для дальнейших расчетов. Находится кэш-память в трех местах: внутри микропроцессора, между микропроцессором и оперативной памятью, внутри жесткого диска.

б) Внешняя память. Реализуется в виде довольно разнообразных устройств хранения информации и обычно конструктивно оформляется в виде самостоятельных блоков. Внешняя память (ВЗУ) предназначена для длительного хранения программ и данных, и целостность её содержимого не зависит от того, включен или выключен компьютер.

4.Форматирование дисков

Форматирование - создание логической и физической структуры диска. Формирование физической структуры диска состоит в создании на диске концентрических дорожек (треков), которые в свою очередь делятся на секторы. Логическое строение диска представляет собой совокупность секторов, каждый из которых имеет свой порядковый номер. Сектора нумеруются в линейной последовательности от первого сектора нулевой дорожки до последнего сектора последней дорожки.

Накопители на жестких магнитных дисках

Если гибкие диски — это средство переноса данных между компьютерами, то жесткий диск — информационный склад компьютера.

Накопитель на жестких магнитных дисках (англ. HDD — Hard Disk Drive)— это наиболее массовое запоминающее устройство большой ёмкости, в котором носителями информации являются круглые алюминиевые пластины — платтеры, обе поверхности которых покрыты слоем магнитного материала. Используется для постоянного хранения информации — программ и данных.

Цилиндр — это общее количество дорожек, с которых можно считать информацию, не перемещая головок. Поскольку гибкий диск имеет только две стороны, а дисковод для гибких дисков — только две головки, в гибком диске на один цилиндр приходится две дорожки. В жестком диске может быть много дисковых пластин, каждая из которых имеет две (или больше) головки, поэтому одному цилиндру соответствует множество дорожек.

Кластер (или ячейка размещения данных) — наименьшая область диска, которую операционная система использует при записи файла. Обычно кластер — один или несколько секторов.

1.8 Лекция №8 (2 часа).

Тема: «Понятие системного и служебного (сервисного) программного обеспечения: назначение, возможности, структура. Операционные системы. Файловая структура операционных систем. Операции с файлами.»

1.8.1 Вопросы лекции:

1. Программы. Виды программного обеспечения.
2. Системное программное обеспечение.
3. Служебные приложения windows (утилиты).
4. Прикладные программы.
5. Операционная система. Состав операционной системы
6. Загрузка операционной системы

1.8.2 Краткое содержание вопросов:

1 Программы. Виды программного обеспечения

Процесс функционирования любой вычислительной системы представляет собой пошаговое выполнение процессором определённой последовательности простых команд, называемой *п р о г р а м м о й*. Программа составляется в соответствии с алгоритмом решения конкретной задачи на языке, пригодном для автоматического выполнения процессором. Программы - это упорядоченные последовательности команд. Совокупность программ, хранящихся в долговременной памяти ЭВМ и предназначенных для массового использования, составляет программное обеспечение компьютера (ПО ЭВМ). Все множество программ, составляющих ПО ЭВМ, можно разделить на три

группы: системные программы; прикладные программы; системы программирования. Системные программные средства обеспечивают взаимодействие прочих программ с программами базового уровня и с аппаратным обеспечением. Базовое ПО - минимальный набор программных средств, обеспечивающих работу компьютера. Базовые программные средства отвечают за взаимодействие с базовыми аппаратными средствами, входят в состав базового оборудования и хранятся в специальных микросхемах (ПЗУ). Как правило, поставляется вместе с компьютером. Сервисное (служебное ПО) - специальные программы, выполняющие некоторые дополнительные услуги системного характера (например, архивирование файлов, защита от вирусов, «лечение» и оптимизация дисков и т. д.)- Эти программы называются утилитами. Системы, программирования — инструмент для работы профессиональных программистов. Каждая такая система ориентирована на определенный язык программирования: Паскаль, Бейсик, Фортран, Си, Ассемблер и др. Системы программирования позволяют создавать тексты программ, отлаживать и исполнять программы. Все перечисленные выше виды ПО создаются программистами с помощью специально созданных систем программирования

2 Системное программное обеспечение

Назначение, системных программ — управление работой устройств, составляющих компьютер; организация связи с пользователем (пользовательский интерфейс); работа с файлами. Перечисленные функции выполняет операционная система (ОС) ЭВМ. ОС — это основа всего программного обеспечения компьютера. (ОС) - комплекс программных средств, управляющих работой ЭВМ. Она представляет собой несколько программ, которые позволяют обмениваться информацией между процессором, памятью и периферийными устройствами: клавиатурой, монитором, принтером, дисковыми и др. В настоящее время наиболее популярными ОС для персональных компьютеров типа IBM PC/AT являются MS DOS, MS Windows 3.11, MS Windows 95, 98, ME, 2000, MS Windows NT, OS/2, Linux и другие, различающиеся по своим возможностям и назначению. Существуют также десятки других ОС, разработанных для различных вычислительных систем, серверов, специализированных компьютеров и контроллеров (например, UNIX, Mac OS, Solaris, BSD, Next и др.).

Операционная система состоит из так называемого *я д р а*, обеспечивающего важнейшие функции ОС (распределение памяти, запуск задач и процессов и др.), набора *д р а й в е р о в* (модули, управляющие каким-то определенным периферийным устройством, например, драйвер принтера или драйвер сетевой карты) и некоторого комплекта прикладных программ (например, простейший текстовый и графический редактор).

3 Служебные приложения windows (утилиты).

Служебные приложения windows (утилиты) - особый класс программ, предназначенный для обслуживания компонентов компьютерной системы и оптимизации ее работы. Своевременное обслуживание повышает надежность компьютера, а оптимизация - его быстродействие. Доступ к служебным приложениям можно получить по следующей команде: Пуск/Все программы/Стандартные/Служебные.

Установка (инсталляция) ПО - это процесс формирования приложения из исходного дистрибутивного комплекта. При установке новых приложений устанавливаются только те программные ресурсы, которые нужны для работы данного приложения, но отсутствуют на данном компьютере в динамических библиотеках.

4 Прикладные программы

Прикладные программы дают возможность пользователю непосредственно решать свои информационные задачи. Прикладное программное обеспечение, в свою очередь, делится на три части.

1. Прикладными программами общего назначения и включают в себя:

- системы обработки текстов (текстовые процессоры);
- системы компьютерной графики (графические редакторы);

- системы табличных расчетов — электронные таблицы (табличные процессоры);
- системы управления базами данных, ориентированные на пользователя;
- средства разработки презентаций;
- программы, обеспечивающие работу с электронной почтой и работу с Интернетом.

2. Специализированные программы (профессионально - ориентированные). Дать их полный перечень практически невозможно. Математики, инженеры, научные работники многих специальностей нуждаются в программах, выполняющих математические расчеты; профессиональные издатели книг не могут довольствоваться текстовыми процессорами общего назначения и нуждаются в специальных программах — издательских системах; бухгалтерам и экономистам требуются свои программы. Дело идет к тому, что для любой профессии, связанной с обработкой информации, будет создано свое специализированное ПО.

5 Операционная система. Состав операционной системы

Операционная система (ОС) – это комплекс специальных программных средств, предназначенный для управления загрузкой, запуском и выполнением других (пользовательских) программ, а также для планирования и управления вычислительными ресурсами ЭВМ. Операционная система обеспечивает совместное функционирование всех устройств компьютера и предоставляет пользователю доступ к его ресурсам.

Различают ОС, использующие командную строку для ввода команд и запуска программ с использованием клавиатуры, и графические операционные системы, в которых основным устройством для запуска является мышь или другое устройство позиционирования.

Состав ОС

1. Программные модули, управляющие файловой системой, т.к. процесс работы компьютера в определенном смысле сводится к обмену файлами между устройствами.
2. Командный процессор, — программа, которая запрашивает у пользователя команды и выполняет их.
3. Драйверы устройств - специальные программы, которые обеспечивают управление работой устройств и согласование информационного обмена с другими устройствами, а также позволяют производить настройку некоторых параметров устройств. Каждому устройству соответствует свой драйвер.
4. Программные модули, создающие графический пользовательский интерфейс. В операционных системах с графическим интерфейсом пользователь может вводить команды с помощью мыши, тогда как в режиме командной строки необходимо вводить команды с помощью клавиатуры.
5. Справочная система - позволяет оперативно получить необходимую информацию как о функционировании операционной системы в целом, так и о работе ее отдельных модулей.

6 Загрузка операционной системы

После включения компьютера производится загрузка операционной системы с системного диска в оперативную память.

В состав компьютера входит энергонезависимое постоянное запоминающее устройство (ПЗУ), содержащее программы тестирования компьютера и первого этапа загрузки операционной системы — это BIOS (базовая система ввода/вывода). После включения питания компьютера процессор начинает выполнение программы самотестирования компьютера. Производится тестирование работоспособности процессора, памяти и других аппаратных средств компьютера.

После проведения самотестирования специальная программа, содержащаяся в BIOS, начинает поиск загрузчика операционной системы. Происходит поочередное

обращение к имеющимся в компьютере дискам (гибким, жестким, CD - ROM) и поиск на определенном месте (в первом, так называемом загрузочном секторе диска) наличия специальной программы Master Boot (программы-загрузчика операционной системы).

Если диск системный и программа-загрузчик оказывается на месте, то она загружается в оперативную память и ей передается управление работой компьютера

1.9 Лекция №9 (2 часа).

Тема: «Программные средства решения задач оформления управленческой документации»

1.9.1 Вопросы лекции:

1. Функциональные возможности текстового процессора.
2. Вычисления в таблицах.
3. Создание гипертекстового документа.
4. Запись макроса.

1.9.2 Краткое содержание вопросов:

1 Функциональные возможности текстового процессора

Возможности текстового процессора:

- обеспечение ввода текста с клавиатуры или из существующего файла;
- редактирование текста (добавление, изменение, удаление или копирование фрагментов текста - символов, слов и т.д.);
- оформление текста (выбор шрифтов, способа выравнивания, установление межстрочного и межабзацного интервалов и т.п.);
- размещение текста на странице (установка размера страницы, полей, отступов; разбиение на колонки; расстановка номеров страниц, колонтитулов и пр.)
- сохранение текста в файле на внешнем носителе или получение твердой копии (печать текста);
- проверка орфографии, подбор синонимов, контекстный поиск и замена;
- система подсказок и т.д.
- настраиваемое пользователем меню;
- использование контекстного меню;
- сопровождение текста таблицами и проведение в них простейших расчетов;
- вставка графических объектов (рисунков, диаграмм, заголовков и пр.) или создание рисунков с помощью встроенных инструментов;
- вставка формул, графиков, диаграмм;
- оформление текста списками, буквицами;
- использование инструмента автокоррекции текста и его автореферирования;
- фоновая проверка орфографии, синтаксиса и многое другое.

Word поддерживает разработанный Microsoft механизм связывания и внедрения объектов OLE, что обеспечивает возможность разделения данных и функций с любыми программами, для которых реализована поддержка OLE. Являясь частью Microsoft Office, Word может использовать данные совместно с Microsoft Excel, Microsoft PowerPoint, Microsoft Access и Microsoft Outlook, а также с Microsoft Internet Explorer.

Каждый документ Word может содержать текст наряду с графикой, звуками, полями, гиперссылками или ссылками на другие документы и даже с видеоклипами. Документы могут быть сохранены в формате Web-страниц с добавлением HTML-сценариев.

2Вычисления в таблицах

Одной из возможностей текстового процессора является вычисления в таблицах. Таблица может содержать максимум 31 столбец и произвольное число строк. Ячейки

таблицы имеют адреса, образованные именем столбца (А, В, С...) и номером строки (1, 2, 3...), например, А1, С4 и т.д.

Word позволяет выполнять вычисления, записывая в отдельные ячейки таблицы формулы с помощью команды *Таблица\Формула...* Формула задается как выражение, в котором использованы:

- *абсолютные* ссылки на ячейки таблицы в виде списка (разделяемые знаком«;» А1; В5; Е10 и т.д.) или блока (начало и конец блока ячеек — А1:А10);
- *ключевые* слова для ссылки на блок ячеек:
- LEFT — ячейки, расположенные в строке левее ячейки с формулой; RIGHT — ячейки, расположенные в строке правее ячейки с формулой; ABOVE — ячейки, расположенные в столбце выше ячейки с формулой BELOW — ячейки, расположенные в столбце ниже ячейки с формулой
- константы — числа, текст в двойных кавычках;
- закладки, которым соответствует определенный текст документа (например числа), созданный с помощью команды *Правка\Закладка*;
- встроенные функции Word, например, SUM(), AVERAGE().
- знаки операции (+, —, *, /, %, ^, =, <, >, <=, >=).

3 Создание гипертекстового документа

Гипертекст — это способ организации текстовой информации, внутри которой установлены смысловые связи между ее различными фрагментами. Такие связи называются гиперсвязями. Гиперссылка - цветной подчеркнутый текст при указании на который осуществляется переход на связанный объект в этом же или другом документе.

Гипертекстовую информацию можно читать не только в обычном порядке, «листая страницы» на экране, но и перемещаясь по смысловым связям в произвольном порядке. Наиболее важные сферы применения гипертекста — это справочные системы, создание гипертекстовых страничек в Internet с помощью языка HTML и электронные учебники. Содержание Web-страниц и публикаций Word можно обогатить с помощью гиперссылок на другие страницы и документы.

Гиперссылки позволяют перейти к другому разделу текущего документа или Web-страницы, к другому документу Word или к другой Web-странице или к файлу, созданному в другой программе.

Для создания ссылок на элементы текущего документа рекомендуется использовать либо стили заголовков, либо закладки Microsoft Word. Закладка - это специальная пометка в документе, которая позволяет ссылаться на определенное место в документе или фрагмент текста по имени закладки. Такие ссылки могут использоваться для быстрого перемещения по тексту, в формулах, для перекрестных ссылок

4 Запись макроса

Если какая-либо задача часто выполняется в Microsoft Word, ее выполнение можно автоматизировать с помощью макроса. Макрос — это набор команд и инструкций, выполняемых как одна команда. Макросы часто используются для следующих целей:

- для ускорения часто выполняемых операций редактирования или форматирования
- для объединения нескольких команд, например, для вставки таблицы с указанными размерами и границами и определенным числом строк и столбцов
- для упрощения доступа к параметрам в диалоговых окнах
- для автоматизации обработки сложных последовательных действий в задачах

Средство записи макросов очень полезно, особенно для начинающих. Запись макроса можно временно приостановить, а затем продолжить.

Для выбора команд и параметров допускается применение мыши. Однако действия, сделанные в окне документа с помощью мыши, не записываются. Для записи таких

действий, как перемещение курсора, а также выделение, копирование и перемещение текста, необходимо использовать клавиатуру. Необходимо использовать пункты меню в текстовом редакторе (пиктограммы не работают!).

Алгоритм создания макроса:

1. В меню Сервис выделите пункт Макрос, а затем выберите команду Начать запись.
2. В поле Имя макроса введите имя нового макроса.
3. Нажмите кнопку ОК, чтобы начать запись макроса.
4. Выполните действия, которые следует включить в макрос.
5. После выполнения нужных действий Сервис – Макрос – Остановить запись
6. Отследить работы макроса: Сервис – Макрос - Макросы – ввести имя макроса - Выполнить

Советы по записи макросов:

1. Перед записью макроса запланируйте заранее шаги и команды, которые он должен выполнить.
2. Если при записи будет допущена ошибка, то в макросе также запишутся все исправления. Ненужные инструкции можно будет убрать позже путем редактирования макроса.
3. Постарайтесь предусмотреть все сообщения Microsoft Word, которые могут приостановить действие макроса или направить его по ложному пути.
4. Перед использованием макроса, записанного в другом документе, убедитесь, что он не зависит от содержимого этого документа.

1.10 Лекция №10 (2 часа).

Тема: «Графические редакторы»

1.10.1 Вопросы лекции:

1. Основы компьютерной графики.
2. Профессиональные графические редакторы.
3. Форматы графических файлов.

1.10.2 Краткое содержание вопросов:

1 Основы компьютерной графики.

Создать объекты иллюстративной графики, включая динамические ролики, можно средствами программирования, а также графическими редакторами и системами. Рассмотрим два из них.

Редактор Paint

Графический редактор Paint, входящий в комплект стандартных программ MS Windows 95, позволяет, используя манипулятор «мышь», выполнять черно-белые и цветные рисунки, обрамлять их текстом, выводить на печать. В Paint можно работать с фрагментами графических изображений: копировать, перемещать, поворачивать, изменять размеры, записывать на диск и считывать с диска. С помощью Paint можно обрабатывать графические изображения, а также считывать и записывать в файл полностью или частично изображение с дисплея, если монитор работает в графическом режиме.

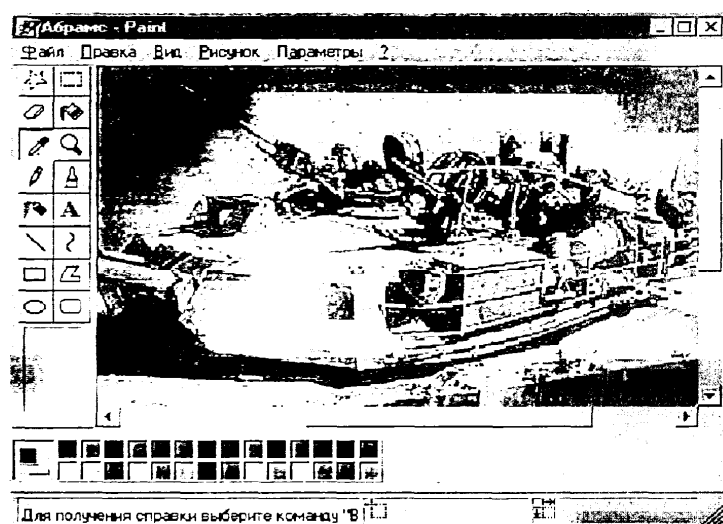


Рис. 2.18. Рабочее окно программы Paint

После загрузки пакета появляется рабочий экран редактора (рис. 2.18). Большую часть экрана занимает рабочее поле, окрашенное в фоновый цвет. Над рабочим полем - меню, позволяющее выполнять команды редактирования. Слева от рабочего поля расположена панель инструментов, на которой высвечен инструмент, в данный момент являющийся рабочим. Под рабочим полем находится палитра. С левого края палитры показаны два вложенных квадрата, внутренний из которых окрашен в рабочий цвет, а внешний - в фоновый. В левом нижнем углу экрана выводится калибровочная шкала, которая позволяет устанавливать ширину рабочего инструмента (кисти, резинки и т.д.). Установленная в данный момент ширина инструмента отмечена стрелкой. Вдоль нижнего правого и правого края рабочего поля находятся линейки прокрутки для перемещения рабочего поля по картинке, если размеры картинки больше размеров рабочего поля.

Общие правила работы с редактором таковы. Для выбора (установки) параметров работы и выполнения команд в Paintbrush необходимо поместить указатель мыши на пункт меню и щелкнуть левой кнопкой. Выход из программы: *File/Quit*.

Программа *Freeze* предназначена для сохранения выводимого на экран изображения в графическом РСХ-файле с одновременным сохранением оформления экрана, возможностью последующего редактирования данного изображения и вставки его в качестве иллюстрации в текстовые редакторы и настольные издательские системы. Установка программы происходит автоматически при запуске Paintbrush.

Редактор CorelDraw

Начиная работать с графическим редактором CorelDraw, мы прежде всего видим его рабочее окно, рис. 2.19.

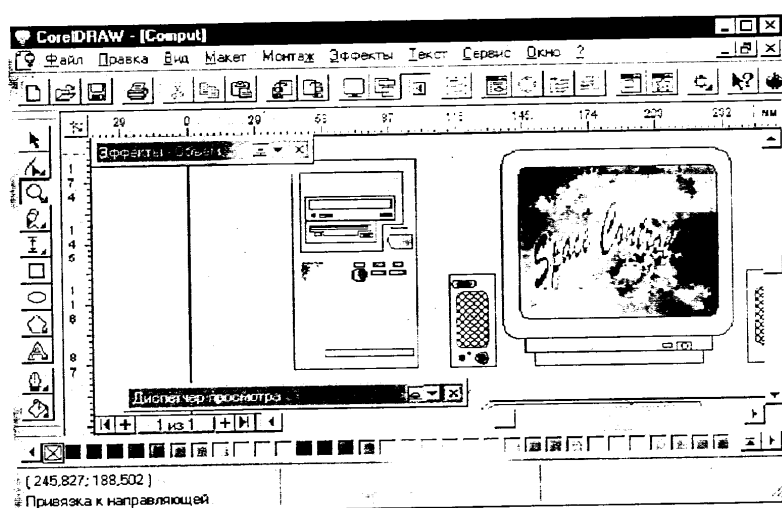


Рис. 2.19. Рабочее окно программы CorelDraw

Чтобы активизировать **меню выбора** (установки) **параметров** работы или выполнения команд, установите указатель мыши на пункт меню и щелкните левой кнопкой или нажмите клавишу Alt и клавишу, соответствующую выделенной букве. В некоторых случаях для облегчения выполнения наиболее часто употребляемых команд ввод их с клавиатуры осуществляется нажатием комбинации клавиш или определенной клавиши. Выполнение команд происходит после подтверждения правильности установки всех параметров или выбора значений активизацией экранной кнопки ОК в диалоговой панели или нажатием клавиши Enter.

Выход из меню - переместить указатель мыши за пределы меню и щелкнуть кнопкой или нажать клавишу Esc.

Выход из программы: *File/Quit* или Alt + F4.

Для вызова **диалоговой панели** выбора (установки) параметров работы или выполнения команд необходимо установить указатель мыши на пункт меню или команду и щелкнуть кнопкой или нажать клавишу Alt и клавишу, соответствующую выделенной букве. Выполнение команд происходит после подтверждения правильности установки всех параметров (выбора значений) активизацией экранной кнопки ОК или нажатием клавиши Enter.

Для **редактирования рисунка** следует активизировать пиктограмму с помощью мыши или нажатием клавиши *Пробел*, переместить указатель на любую точку контура рисунка и щелкнуть кнопкой. Выбранный рисунок будет окружен восемью квадратами черного цвета. Можно выделить одновременно несколько объектов, последовательно выбирая их с помощью мыши при нажатой клавише Shift или отмечая на экране прямоугольную область, в которой они расположены. Для одновременного выбора всех рисунков на экране необходимо активизировать пиктограмму, переместить указатель мыши в один из углов выбираемого прямоугольного контура, нажать кнопку и, не отпуская ее, переместить указатель в противоположный угол и отпустить кнопку. Контур будет изображен штриховой линией. Для отмены выбора - переместить указатель мыши за контур и щелкнуть кнопкой.

Для изменения масштаба выводимого на экран рисунка необходимо активизировать соответствующую пиктограмму.

Работа с текстом начинается с активизации пиктограммы текста. Для перемещения по тексту в диалоговой панели *Text* необходимо переместить указатель мыши и щелкнуть левой кнопкой. Однако в программе для перемещения предусмотрены также специальные клавиши.

Для выделения фрагмента текста необходимо активизировать пиктограмму с помощью мыши или клавиши *Пробел*. Переместите указатель мыши в один из углов выбираемого прямоугольного контура, нажмите кнопку K, не отпуская ее, переместите указатель в противоположный угол, отпустите кнопку. Весь контур будет окружен штриховой линией. Для выделения отдельного символа 'необходимо активизировать пиктограмму, переместить указатель мыши в один из углов выбираемого прямоугольного контура, нажать кнопку и, не отпуская ее, переместить указатель в противоположный угол. Выбранный объект не будет окружен рамкой из квадратов, однако на нем будут выделены все узловые точки. Выбор нескольких символов осуществляется аналогично при нажатой клавише Shift.

Прежде чем начать оформление ранее введенного текста, его необходимо выделить одним из выше описанных способов.

2 Профессиональные графические редакторы.

ДЕЛОВАЯ ГРАФИКА

Одним из первых приложений компьютерной графики стало отображение данных экономических расчетов.

Графические представления расчетных и статистических данных удобно представлять в виде схем, диаграмм, гистограмм и графиков. Различают следующие их виды:

гистограмма - группа столбцов, пропорциональных по высоте определенным числовым значениям;

круговая диаграмма - секторы круга, углы которых пропорциональны элементам данных;

линейный график - отображение исходных величин в виде точек, соединенных отрезками прямых линий;

временная диаграмма - последовательность операций или процессов определенной длительности (измерение динамических процессов);

структурная схема - представление сложных объектов в виде дерева или графа;

круговая гистограмма - представление относительных величин объектов, которым на изображении сопоставляются размеры и расположение кругов в прямоугольной системе координат.

Из числа средств прикладного программного обеспечения общего назначения графическое представление данных лучше всего развито в электронных таблицах и в СУБД.

Одним из первых практических применений машинной графики было автоматическое построение графиков функции в различных системах координат. Обычно графики функций строят в декартовых координатах (в прямоугольной системе, рис. 2.20).

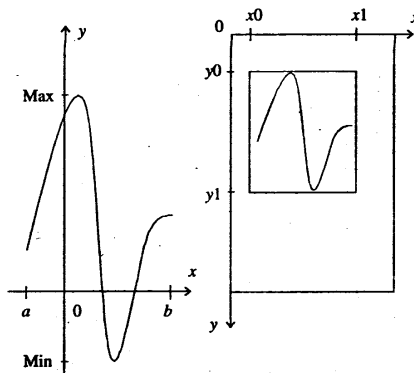


Рис. 2.20. Построение на экране графиков функций (в декартовой системе координат)

В общем виде алгоритм построения графика заданной функции $y = f(x)$ на отрезке $[a, b]$ заключается в следующем.

1. Определяем область значений функции, для чего найдем максимальное по модулю значение функции на заданном отрезке $[a, b]$, $m = \max(\text{abs}(f(x)))$ для всех x из $[a, b]$.

Примем для удобства, что минимальное значение функции на отрезке совпадает с максимальным, но с обратным знаком. Таким образом, область значений функции лежит в интервале $[-m, m]$.

Поиск максимума можно осуществить разными способами, например, табулируя функцию $f(x)$ на отрезке с разбиением на n частей и определяя максимальное значение в

массиве чисел $Y_i = f(x_i)$, где $x_i = a + i \cdot (b - a) / n$, для $i = 0, \dots, n$.

2. Задаем координаты окна $x0, y0, x1, y1$ на графическом дисплее, в котором будем строить график функции.

3. Формулы преобразования координат x, y точек прямоугольника $[a, b] \cdot [-m, m]$ обычной декартовой системы в соответствующие координаты u, v окна $[x0, x1] \cdot [y0, y1]$ графического экрана можно задать в следующем виде :

$$u = x0 + (x - a)(x1 - x0)/(b - a),$$

$$v = (y0 + y1) / 2 - f(x) (y1 - y0) / (2m).$$

Тогда автоматическое построение графика функции на экране дисплея осуществляется путем установки точек (u_i, v_i) , соответствующих точкам $(x_i, f(x_i))$, выбранным в декартовой системе. Часто бывает полезно соединять полученные точки отрезками или специальными линиями, что программы могут делать (или не делать) по желанию пользователя.

4. Далее можно оформить график, нарисовав оси координат, нанести масштабные сетки, вывести соответствующие обозначения и комментарии. Оси координат на графическом экране в заданном окне легко построить, вычислив экранные координаты начала выбранной декартовой системы $\{xv, yv\}$:

$$xv = x0 - a(x1 - x0)/(b - a),$$

$$yv = (y0 + y1) / 2.$$

ИНЖЕНЕРНАЯ ГРАФИКА

Компьютеризацию чертежных и конструкторских работ проводят давно и в настоящее время используют различные системы автоматизации проектных работ (САПР). Аббревиатуру САПР впервые использовал основоположник этого научного направления Айвен Сазерленд в своих лекциях, прочитанных в Массачусетском технологическом институте в начале 60-х годов. Фактически инженеры применяли компьютеры для решения сложных задач проектирования еще в эпоху ранних послевоенных моделей универсальных компьютеров, а первые образцы специализированного оборудования САПР были созданы уже в середине 50-х годов. Однако широкое распространение САПР обусловлено появлением микропроцессорной техники, предоставившей возможности создавать, модифицировать и обрабатывать сложные графические изображения на экране монитора.

В настоящее время САПР обозначает аппаратно-программный комплекс, поддерживающий процесс проектирования с использованием специальных средств машинной графики, поддерживаемых пакетами программного обеспечения, для решения задач, связанных с проектной деятельностью. В совокупности развитая САПР представляет собой специализированную информационную систему. Сфера применения САПР охватывает такие разные области приложения, как архитектура, гражданское строительство, картография, медицина, геофизика, разработка моделей одежды, издательское дело, реклама.

Полная система САПР состоит из компонентов аппаратного и программного обеспечения. Общими компонентами аппаратного обеспечения системы САПР являются ЦП (центральный процессор), несколько рабочих станций, разделяемая между рабочими станциями периферия.

Состав типичной системы САПР:

дисплей (графический и алфавитно-цифровой);

процессор;

клавиатура;
устройство управления курсором (мышь, дигитайзер);
электронный командный планшет;
принтер.

Одним из наиболее давних и популярных средств автоматизированного проектирования является система АВТОКАД (AutoCad). АВТОКАД не является проблемно-ориентированной системой, т.е. не содержит специализированных баз данных, экспертных систем и многого из того, что входит в состав специализированной интеллектуальной САПР. АВТОКАД - достаточно простая универсальная система. Ее возможности таковы:

- развитая система экранных меню;
- высокая точность графической информации;
- разбивка информации (расслоение);
- прочерчивание на дисплее координатной сетки;
- средство захвата графических объектов;
- мощное редактирование;
- отображение параметров графических характеристик;
- полуавтоматическая и автоматическая простановка размеров;
- штриховка;
- работа с блоками.

После запуска АВТОКАДа на текстовом экране появляется главное меню:

О - выход;
1 - создание нового чертежа;
2-редактирование существующего чертежа;
3 - вывод на плоттер (графопостроитель);
4 - вывод на принтер;
5-конфигурация;
6 - файловые утилиты;
7-шрифты;
8 -стыковка со старыми версиями.

Режимы экранного меню:

AUTOCAD	- выход в головное меню;
* * * *	- режим объектного захвата;
BLOCKS	-работа с блоками;
DISPLAY	- работа с изображением без его изменения;
SETTINGS	-настройка;
DIM	- обезразмеривание;
EDIT	-редактирование;
DRAW	-рисование;
LAYER	-работа со слоями;

INQUIRY	-справки о примитивах;
UTILTTS	- выход в ДОС, запись чертежей в разных форматах;
PLOT	-получение твердой копии и т.д.

В режиме DRAW (рисуи) имеется возможность строить графические примитивы и проводить с их помощью синтез изображений. Например, здесь существует восемь способов рисования дуг:

- по трем точкам на дуге /3 points/;
- по начальной точке, центру и длине хорды /S, C, L/;
- по начальной точке, центру и заключенному углу /S, C, A/;
- по начальной точке, конечной точке и радиусу /S, E, R/;
- по начальной точке, конечной точке и заключенному углу /S, E, A/;
- по начальной точке, конечной точке и исходному направлению /S, E, D/;
- по продолжению предыдущей линии или дуги /CONTIN:/.

Можно выделить следующие правила изображения дуги:

- 1) обычно дуга строится против часовой стрелки от точки к точке;
- 2) если есть в выбранной опции возможность задать угол, то отрицательный угол позволяет рисовать дугу по часовой стрелке;
- 3) по умолчанию дуга рисуется по начальной, промежуточной и конечной точкам.

При выборе в основном меню АВТОКАД режима *Edit* имеется возможность стирать созданные объекты, возвращать случайно стерты, перемещать и копировать, вычерчивать сопряжения между двумя существующими объектами, снимать фаски, разбивать объект на части, зеркально отображать, поворачивать, увеличивать и уменьшать, отсекал и т.п.

Практически все команды редактирования запрашивают выбор одного или нескольких объектов для обработки. Совокупность таких объектов называется набором выбора. Когда АВТОКАДу требуется не один объект, а набор выбора, появляется подсказка *Select objects*: (выбрать объекты) и на экране вы видите маленький прямоугольник - мишень для выбора объектов. Способы выбора объектов указываются в опциях соответствующей команды.

НАУЧНАЯ ГРАФИКА

Компьютерная графика представляет значительный интерес для научных исследований. В частности, она выступает как средство формирования научной документации с использованием специальной нотации - математических знаков, индексов, шрифтов и т.п. В последнее время ученые чаще стали обращаться к имитационному моделированию на компьютере.

В компьютерной графике большое значение имеют методы и способы **геометрического моделирования**. Модели, геометрические преобразования составляют в настоящее время основу теории компьютерной графики и геометрического моделирования.

Аналитические модели - это набор чисел, логических параметров, играющих роль коэффициентов в уравнениях, которые задают графический объект заданной формы. Например, аналитической моделью окружности на плоскости в параметрической форме являются уравнения

$$x=x_0+R\cdot\cos A,$$

$$y = y_0 + R \cdot \sin A,$$

где x_0, y_0 - координаты центра, R - радиус, A - угол. Параметрическое задание образов широко применяется в машинной графике и геометрии. Изображение окружности можно осуществить установкой последовательных точек (близко расположенных), изменяя генерирующий параметр A от 0 до 360°.

Координатные модели - это массивы координат точек, принадлежащих объектам. Например, поверхность задается массивом точек $Z = f(x, y)$ на координатной сетке $[x_i, y_j]$. Если точки в модели расположены в том же порядке, что и на линии образа, то модели называют упорядоченными. Помимо координат, в модели могут быть указаны дополнительные характеристики проекции касательных или нормальных векторов.

Приближенные модели содержат аппроксимации кривых методами вычислительной геометрии. Например, изображение гладких кривых можно осуществить ломаными линиями: линейными, параболическими или сплайнами. Используя вышеперечисленные геометрические модели, можно создавать различные демонстрационные картины. Например, модель Солнечной системы для наглядности удобно представить в динамической форме. Организуем движение точки (Земли) по окружности, в центре которой размещается круг (имитация Солнца). Установку точки на орбите осуществим по параметрическим формулам окружности:

$$X_0 = 320 + r_l \cdot \sin(A_l);$$

$$Y_0 = 240 + r_l \cdot \cos(A_l),$$

где r_l - радиус орбиты Земли, A_l - параметрический угол, меняющийся от 0 до 360°. Чтобы организовать движение, достаточно в цикле устанавливать точку с координатами (x_0, y_0) для всех углов A_l , принимающих значения от 0 до 360° с шагом h . Аналогичная процедура справедлива и для второй точки (Луны), которая изображается по подобным формулам, в которых центр орбиты (Земля) является подвижным:

$$x = x_0 + r \cdot \sin(A);$$

$$y = y_0 + r \cdot \cos(A),$$

где r - радиус орбиты Луны, A - угол вращения.

3 Форматы графических файлов.

Форматы файлов – основа работы с цифровыми фотографиями. FotoTips.ru расскажет вам о всех основных форматах графических файлов.

RAW.

Формат файлов содержащий необработанную информацию, поступающую напрямую с матрицы фотокамеры. Эти файлы не обрабатываются процессором камеры (в отличие от JPG) и содержат оригинальную информацию о съемке. RAW может быть сжат без потери качества.

Преимущества RAW очевидны – в отличие от JPG, который был обработан в камере и уже сохранен с сжатием данных – RAW дает широчайшие возможности по обработке фотографии и сохраняет максимальное качество.

Заметка. Разные производители фототехники используют разные алгоритмы для создания RAW в своих камерах. Каждый производитель придумывает собственное разрешение для своего RAW-файла – NEF – Nikon, CR2 – Canon...

JPEG (он же JPG).

Это самый распространенный формат графических файлов.

Свою популярность JPG заслужил гибкой возможностью сжатия данных. При необходимости изображение можно сохранить с максимальным качеством. Либо сжать его до минимального размера файла для передачи по сети.

В JPG применяется алгоритм сжатия с потерей качества. Что это нам дает? Явный минус такой системы – потеря качества изображения при каждом сохранении файла. С другой сжатие изображения в 10 раз упрощает передачу данных.

На практике, сохранение фотографии с минимальной степенью сжатия не дает видимого ухудшения качества изображения. Именно поэтому JPG – самый распространенный и популярный формат хранения графических файлов.

TIFF.

Формат TIFF очень популярен для хранения изображений. Он позволяет сохранять фотографии в различных цветовых пространствах (RGB, CMYK, YCbCr, CIE Lab и пр.) и с большой глубиной цвета (8, 16, 32 и 64 бит). TIFF широко поддерживается графическими приложениями и используется в полиграфии.

В отличие от JPG, изображение в TIFF не будет терять в качестве после каждого сохранения файла. Но, к сожалению, именно из-за этого TIFF файлы весят в разы больше JPG.

Право на формат TIFF в данный момент принадлежит компании Adobe. Photoshop может сохранять TIFF без объединения слоев.

PSD.

Формат PSD используется в программе Photoshop. PSD позволяет сохранять растровое изображение со многими слоями, любой глубиной цвета и в любом цветовом пространстве.

Чаще всего формат используется для сохранения промежуточных или итоговых результатов сложной обработки с возможностью изменения отдельных элементов.

Так же PSD поддерживает сжатие без потери качества. Но обилие информации, которое может содержать PSD файл, сильно увеличивает его вес.

BMP.

Формат BMP один из первых графических форматов. Его распознает любая программа работающая с графикой, поддержка формата интегрирована в операционные системы Windows и OS/2.

BMP хранит данные с глубиной цвета до 48 бит и максимальным размером 65535×65535 пикселей.

На данный момент формат BMP практически не используется ни в интернете (JPG весит в разы меньше), ни в полиграфии (TIFF справляется с этой задачей лучше).

GIF.

Формат GIF был создан на заре интернета для обмена изображениями. Он может хранить сжатые без потери данных изображения в формате до 256 цветом. Формат GIF идеально подходит для чертежей и графиков, а так же поддерживает прозрачность и анимацию.

Так же GIF поддерживает сжатие без потери качества.

PNG.

Формат PNG создан как для улучшения, так и для замены формата GIF графическим форматом, не требующим лицензии для использования. В отличие от GIF, у PNG есть поддержка альфа-канала и возможность хранить неограниченное количество цветов.

PNG сжимает данные без потерь, что делает его очень удобным для хранения промежуточных версий обработки изображений.

JPEG 2000 (или jp2).

Новый графический формат, созданный для замены JPEG. При одинаковом качестве размер файла в формате JPEG 2000 на 30% меньше, чем JPG.

При сильном сжатии JPEG 2000 не разбивает изображение на квадраты, характерные формату JPEG.

К сожалению, на данный момент этот формат мало распространен и поддерживается только браузерами Safari и Mozilla/Firefox (через Quicktime).

1.11 Лекция №11 (2 часа).

Тема: «Электронные таблицы»

1.11.1 Вопросы лекции:

1. Задачи структуризации и первичной обработки данных.
2. Средства структуризации и первичной обработки данных: сортировка; фильтрация; создание итоговых, сводных и консолидированных таблиц.
3. Визуализация данных, диаграммы.

1.11.2 Краткое содержание вопросов:

1 Задачи структуризации и первичной обработки данных

Электронная таблица – самая распространенная и мощная информационная технология для профессиональной работы с данными. Для управления работой с электронной таблицей созданы специальные программные продукты - табличные процессоры. Электронная таблица — это прикладная программа, предназначенная для обработки структурированных в виде таблицы данных т.е. это компьютерный эквивалент обычной таблицы, клетках (ячейках) которой записаны данные различных типов: текста, даты, формулы, числа.

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

Одним из наиболее распространенных средств работы с документами, имеющими табличную структуру, является программа Microsoft Excel.

Документ Excel называется рабочей книгой. Рабочая книга представляет собой набор рабочих листов, каждый из которых имеет табличную структуру и может содержать одну или несколько таблиц. Рабочий лист состоит из строк и столбцов. Столбцы озаглавлены прописными латинскими буквами и, далее, двухбуквенными комбинациями. Всего рабочий лист может содержать до 256 столбцов, пронумерованных от A до IV. Строки последовательно нумеруются цифрами, от 1 до 65 536 (максимально допустимый номер строки). Пересечении строк и столбцов образуют ячейки.

Ячейки являются минимальными элементами для хранения данных. Обозначение отдельной ячейки сочетает в себе номера столбца и строки, на пересечении которых она расположена, например: A1 или DE234. Обозначение ячейки выполняет функции ее адреса. Адреса ячеек используются при записи формул, определяющих взаимосвязь между значениями, расположенными в разных ячейках. На данные, расположенные в ячейках, можно ссылаться в формулах как на единое целое. Прямоугольную группу ячеек называют диапазоном или блоком. Диапазон ячеек обозначают, указывая через двоеточие номера ячеек, расположенных в противоположных углах прямоугольника, например A1:C15.

2 Средства структуризации и первичной обработки данных: сортировка; фильтрация; создание итоговых, сводных и консолидированных таблиц

Сортировки данных. Процесс упорядочения строк в таблице называется сортировкой. При сортировке изменяется исходный порядок следования строк таблицы или записей базы данных. Чтобы можно было вернуть исходный порядок записей лучше ввести столбец номеров записей.

Команда Данные - Сортировка устанавливает порядок следования строк в таблице в соответствии с возрастанием или убыванием значений в конкретном столбце. Если необходимо отсортировать записи по значениям только одного столбца, надо установить курсор в этот столбец и использовать кнопку сортировки на панели инструментов. В

случае многоуровневой сортировки, в окне команды Данные - Сортировка указываются последовательно заголовки столбцов таблицы. Если таблица размещена не в начале листа Excel, то перед вызовом команды сортировки надо выделить таблицу или заголовки ее столбцов. Если в таблице нет строки заголовков столбцов, то идентифицировать столбцы можно по обозначениям Excel (A1, B1 и т.д.). Для этого установите переключатель внизу окна команды в нужную позицию. Если сортировку надо произвести не по строкам таблицы, а по столбцам, то нажав кнопку Параметры внизу окна команды откроете возможность указать это в окне параметров.

Фильтрация (отбор) данных в списке. Для отбора части записей таблицы удобно использовать команду Данные ► Фильтр. В этой команде имеется несколько функций.

а) Функция Автофильтр используется тогда, когда надо отобразить записи по значениям одного из столбцов таблицы..

Если в качестве критерия фильтра выбрать не значение, а режим Условие, то откроется окно для ввода условия отбора. В этом окне можно выбором задать любые одно или несколько условий на значения отбираемых записей столбца. Чтобы отменить автофильтр для таблицы надо повторно открыть команду Данные - Фильтр - Автофильтр и щелчком убрать галочку в режиме.

Б) Функция команды фильтрации Данные - Фильтр - Расширенный Фильтр. Она позволяет связать условия отбора по значениям нескольких столбцов списка.

Сводная таблица - это инструмент Excel для переупорядочивания данных уже существующих таблиц или списков. Так как построение сводной таблицы многоступенчатый процесс, то помогает ее строить режим Мастер сводных таблиц. Он открывается из меню Данные. Сводную таблицу можно в любой момент изменить. Для этого надо на область таблицы поставить курсор и повторно вызвать Мастер сводных таблиц. При этом откроется сразу третье окно мастера и нажав кнопку Макет можно открыть окно структуры и внести в него изменения. Имеется дополнительная панель инструментов, которая позволяет формировать форму печатного отчета, показывать детальные и сводные результаты и многое другое.

Консолидацией называется сведение данных из нескольких таблиц в одну итоговую таблицу. Используется команда Данные–Консолидация. В Excel существует два способа консолидации. По расположению, в этом случае таблицы-источники должны иметь одинаковые заголовки строк и столбцов, т.е. должны быть расположены идентично. По категориям, в этом случае в таблицах-источниках должны иметься одинаковые заголовки строк и столбцов, но размещены они могут быть по-разному.

3 Визуализация данных. Диаграммы.

Диаграммы - это графическое представление числовых данных и их отношений посредством геометрических фигур. Графический способ облегчает рассмотрение данных, делает их наглядными, выразительными, обзримыми.

Данные электронной таблицы можно представить в виде диаграмм. Диаграмма состоит из элементов: линий, столбиков, секторов, точек и т.п. Каждому элементу диаграммы соответствует число в таблице. Числа в таблице и элементы диаграммы связаны между собой таким образом, что при изменении чисел автоматически изменяется изображение элементов диаграммы и наоборот.

Основные типы диаграмм любой электронной таблицы (табличного процессора). Гистограмма (столбцовая диаграмма) представляет собой набор вертикальных столбиков, высота которых определяется значениями данных. Гистограмма применяется для отображения одного или нескольких рядов данных и обычно используется для сопоставления числовых данных, а также для представления изменения данных во времени или по категориям. Линейчатые диаграммы – это гистограмма, столбики которой располагаются не вертикально, а горизонтально. График представляется в виде сглаженной или ломаной линии, соединяющей точки, соответствующие значениям данных. Для некоторых видов графиков значения данных изображаются в виде маркеров.

Круговые диаграммы используются для представления одного ряда данных, они демонстрируют соотношение между целым и его частями и отвечают на вопрос: какую часть целого составляет тот или иной его компонент. Каждое значение на круговой диаграмме представляется в виде сектора круга. Угол сектора пропорционален доле представляемого значения в общей сумме всех значений. Для каждого значения ведется автоматический пересчет в проценты, которые затем могут быть отмечены на диаграмме. Точечная диаграмма (XY– точечная) отображает взаимосвязь между числовыми значениями в нескольких рядах в координатах XY. При подготовке данных в первом ряду помещаются значения аргумента (ось X), во втором (и последующих рядах) – значения функции (функций), соответствующие значениям аргумента из первого ряда данных и откладываемых по оси Y.

Смешанные (комбинированные) типы диаграмм позволяют построить диаграмму с двумя осями и расположить на одной диаграмме данные различных типов. При создании диаграммы смешанного типа возникают две группы рядов. Одна группа – гистограммы, а другая может быть графиком или XY – точечной диаграммой.

1.12 Лекция №12 (2 часа).

Тема: «Обзор программ для решения аналитических задач»

1.12.1 Вопросы лекции:

1. Регрессионная модель. Получение регрессионной модели.
2. Корреляционные зависимости. Корреляционная модель.
3. Подбор параметра. Поиск решения.

1.12.2 Краткое содержание вопросов:

1 Регрессионная модель. Получение регрессионной модели

Зависимости между величинами могут быть представлены в виде математической модели, в табличной и графической формах. Зависимость, представленная в виде формулы, является математической моделью.

Регрессионная модель — это функция, описывающая зависимость между количественными характеристиками сложных систем. Вид регрессионной функции определяется путем подбора по экспериментальным данным.

Получение регрессионной модели происходит в два этапа:

1. подбор вида функции;
2. вычисление параметров функции.

График регрессионной модели называется трендом. Английское слово trend можно перевести как общее направление, или тенденция. На графиках присутствует еще одна величина, полученная в результате построения трендов. Она обозначена как R^2 . В статистике эта величина называется коэффициентом детерминированности. (В ЭТ величина достоверности аппроксимации. Аппроксимация – сглаживание кривой.) Именно она определяет, насколько удачной является полученная регрессионная модель. Характеристикой построенной модели является параметр R^2 — коэффициент детерминированности. Коэффициент детерминированности всегда заключен в диапазоне от 0 до 1. Если он равен 1, то функция точно проходит через табличные значения, если 0, то выбранный вид регрессионной модели предельно неудачен. Чем R^2 ближе к 1, тем удачнее регрессионная модель. Существует два способа прогнозов по регрессионной модели. Если прогноз производится в пределах экспериментальных значений независимой переменной, то это называется восстановлением значения. Прогнозирование за пределами экспериментальных данных называется экстраполяцией

2 Корреляционные зависимости. Корреляционная модель

Зависимости между величинами, каждая из которых подвергается не контролируемому полностью разбросу, называются корреляционными зависимостями. Раздел математической статистики, который исследует такие зависимости, называется корреляционным анализом. Корреляционный анализ изучает усредненный закон поведения каждой из величин в зависимости от значений другой величины, а также меру такой зависимости.

Оценку корреляции величин начинают с высказывания гипотезы о возможном характере зависимости между их значениями. Чаще всего допускают наличие линейной зависимости. В таком случае мерой корреляционной зависимости является величина, которая называется *коэффициентом корреляции*: —• коэффициент корреляции (обычно обозначаемый греческой буквой r) есть число, заключенное в диапазоне от -1 до +1;

- если это число по модулю близко к 1, то имеет место сильная корреляция, если к 0, то слабая;
- близость r к +1 означает, что возрастанию одного набора значений соответствует возрастание другого набора, близость к -1 означает обратное;
- значение r легко найти с помощью Excel без всяких формул (в Excel они встроены).

В Excel функция вычисления коэффициента корреляции называется КОРРЕЛ и входит в группу статистических функций.

3 Подбор параметра. Поиск решения

Подбор параметра – средство Excel, позволяющее решать так называемую обратную задачу, когда требуется, меняя значение одного из исходных данных (параметров), получить заданное значение результата. При этом результат решения задачи должен быть задан в целевой ячейке формулой, содержащей ссылку на изменяемую ячейку с параметром. При подборе параметра его значение изменяется так, чтобы результат в целевой ячейке стал равным заданному числу.

Для подбора параметра выполняется команда Сервис/Подбор параметра... Результат подбора параметра будет показан в окне Результат подбора параметра.

Если решение найдено, его можно сохранить, нажав кнопку ОК (подобранный значение параметра сохранится в изменяемой ячейке), или вернуться к исходному состоянию, нажав кнопку Отмена. Решение может быть не найдено, если результат зависит не от одного параметра или если изменяемая и целевая ячейки логически не связаны.

Создание таблицы подстановки. Для анализа зависимости результата от различных наборов исходных данных в Excel используется Таблица подстановки. Таблица подстановки – это средство Excel, позволяющее получить таблицу, содержащую результаты подстановки заданных значений одного или двух аргументов (параметров) в одну или несколько формул. Выбирая пункт меню Данные / Таблица подстановки..., можно построить либо таблицу с одним параметром для одной или нескольких функций, либо таблицу с двумя параметрами для одной функции.

Принцип использования Таблицы подстановки состоит в следующем:

Задаются исходные данные задачи, в том числе одно из значений аргумента - параметра. Задается диапазон значений параметра (в виде столбца или строки). Выделяется место для размещения значений функции от заданных значений аргумента (столбец справа или строка ниже). Задается формула, определяющая зависимость функции от аргумента – параметра.

Таблица подстановки обеспечивает последовательный выбор элементов из диапазона значений аргумента и подстановку каждого из них в ячейку – параметр. После чего выполняется перерасчет таблицы и значение функции, определяемое формулой, заносится в отведенное для него место рядом с соответствующим значением аргумента (для каждой ячейки столбца в соседнюю ячейку строки, т.е. по строкам, или для каждой ячейки строки в соседнюю ячейку столбца, т.е. по столбцам).

Поиск решения. Средство Excel Поиск решения используется в тех случаях, когда необходимо найти оптимальное (минимальное или максимальное) или заданное значение для отдельной ячейки (целевой ячейки) путем подбора значений других ячеек (изменяемых ячеек) с учетом требуемых ограничений. Чтобы применить это средство, нужно определить входные данные – переменные задачи, целевую функцию и ограничения, если они существуют, то есть осуществить постановку задачи оптимизации:

Поиск решения (последовательность итераций) запускается командой Сервис/Поиск решения...

В открывшемся после этого диалоговом окне задаются:

- целевая ячейка, содержащая формулу вычисления значения целевой функции;
- критерий поиска: минимальное, максимальное или заданное значение (в последнем случае оно указывается);
- изменяемые ячейки, содержащие значения переменных;
- ограничения, если они есть (ограничения можно добавлять, изменять, удалять, щелкая по соответствующей кнопке).

Если после завершения поиска решение найдено, то можно сохранить полученное решение или восстановить исходные данные.

1.13 Лекция №13 (2 часа).

Тема: «Базы данных. Системы управления базами данных»

1.13.1 Вопросы лекции:

1. Информационная система (ИС), назначение, классификация и свойства информационных систем.
2. База данных (БД). Основные понятия баз данных.
3. Системы управления базами данных (СУБД) MS Access . Базовые объекты СУБД Access.
4. Однотабличные и многотабличные базы данных. Межтабличные связи.
5. Создание запросов. Формирование отчетов.

1.13.2 Краткое содержание вопросов:

1 Информационная система (ИС), назначение, классификация и свойства информационных систем

Информационные системы обеспечивают сбор, хранение, обработку, поиск, выдачу информации, необходимой в процессе принятия решений задач из любой области. Они помогают анализировать проблемы и создавать новые продукты.

Информационная система (ИС) – взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели.

Современное понимание информационной системы предполагает использование в качестве основного технического средства переработки информации персонального компьютера. В крупных организациях наряду с персональным компьютером в состав технической базы информационной системы может входить мэйнфрейм или суперЭВМ. Кроме того, техническое воплощение информационной системы само по себе ничего не будет значить, если не учтена роль человека, для которого предназначена производимая информация и без которого невозможно ее получение и представление. Необходимо понимать разницу между компьютерами и информационными системами. Компьютеры, оснащенные специализированными программными средствами, являются технической базой и инструментом для информационных систем. Информационная система немыслима без персонала, взаимодействующего с компьютерами и телекоммуникациями.

Под персоналом понимается сообщество людей, объединенных общими целями и использующих общие материальные и финансовые средства для производства материальных и информационных продуктов и услуг. Назначение информационных систем и баз данных (хранение, поиск, внесение изменений, группировка и сортировка),

2 База данных (БД). Основные понятия баз данных

База данных (БД) – совокупность определенным образом организованной информации на какую-то тему (в рамках некоторой предметной области).

Классификация БД:

- по характеру хранимой информации: фактографические и документальные;
- по способу хранения данных: централизованные и распределенные;
- по структуре организации данных: реляционные* (табличные БД), иерархические и сетевые БД.

Основные понятия и соглашения для фактографической реляционной («Relation» = «отношение») БД: строка таблицы – запись, столбцы – поля. Всякая таблица содержит в себе информацию о некоторой реальной системе (процессе) и, следовательно, является ее информационной моделью. Запись – информация о конкретном объекте (событии) данной системы, значение поля – определенная характеристика (свойство, атрибут) объекта.

Табличная база данных содержит перечень объектов одного типа, т.е. объектов, имеющих одинаковый набор свойств. Такую базу данных удобно представлять в виде двумерной таблицы: в каждой ее строке последовательно размещаются значения свойств одного из объектов; каждое значение свойства — в своем столбце, озаглавленном именем свойства. Столбцы такой таблицы называют полями; каждое поле характеризуется своим именем (именем соответствующего свойства) и типом данных, представляющих значения данного свойства.

Поле базы данных — это столбец таблицы, содержащий значения определенного свойства.

Строки таблицы являются записями об объекте; эти записи разбиты на поля столбцами таблицы, поэтому каждая запись представляет набор значений, содержащихся в полях.

Запись базы данных — это строка таблицы, содержащая набор значений определенного свойства, размещенный в полях базы данных. Каждая таблица должна содержать, по крайней мере, одно ключевое поле, содержимое которого уникально для каждой записи в этой таблице. Ключевое поле позволяет однозначно идентифицировать каждую запись в таблице. Ключевое поле — это поле, значения которого однозначно определяют каждую запись в таблице.

3 Системы управления базами данных (СУБД) MS Access. Базовые объекты СУБД Access

Системы управления базами данных (СУБД) являются едва ли не самым распространенным видом программного обеспечения. СУБД имеют более чем тридцатилетнюю историю развития с сохранением преемственности и устойчивых традиций. Идеологическая ценность СУБД объясняется тем, что в основе программ такого рода лежит концепция *модели данных*, то есть некоторой абстракции представления данных. В большинстве случаев предполагается, что данные представлены в виде *файлов*, состоящих из *записей*. Структура всех записей в файлах одинакова, а количество записей в файле является переменным. Элементы данных, из которых состоит каждая запись, называются *полями*. Поскольку во всех записях имеются одни и те же поля (с разными значениями), полям удобно давать уникальные имена.

СУБД является адекватным средством во всех случаях, когда исходную информацию можно представить в виде таблицы постоянной структуры, но неопределенной длины или в виде картотеки, содержащей неопределенное количество карточек постоянной структуры.

Все СУБД поддерживают в той или иной форме четыре основных операции:

- добавить в базу данных одну или несколько записей;
- удалить из базы данных одну или несколько записей;
- найти в базе данных одну или несколько записей, удовлетворяющих заданному условию;
- обновить в базе данных значение некоторых полей.

Большинство СУБД поддерживают, кроме того, механизм *связей* между различными файлами, входящих в базу. Например, связь может установиться явным образом, когда значением некоторых полей является ссылка на другой файл, такие СУБД называются *сетевыми*, или же связь может установиться неявным образом, например, по совпадению значений полей в различных файлах. Такие СУБД называются *реляционными*.

СУБД MS Access является СУБД реляционного типа, в которой разумно сбалансированы все средства и возможности, типичных для современных СУБД. Реляционная база упрощает поиск, анализ, поддержку и защиту данных, поскольку они сохраняются в одном месте. Access в переводе с английского означает «доступ». MS Access — это функционально полная реляционная СУБД. Кроме того, MS Access одна из самых мощных, гибких и простых в использовании СУБД. В ней можно создавать большинство приложений, не написав ни единой строки программы, но если нужно создать нечто очень сложное, то на этот случай MS Access предоставляет мощный язык программирования — Visual Basic Application.

Популярность СУБД Microsoft Access обусловлена следующими причинами:

- доступность в изучении и понятность позволяют Access являться одной из лучших систем быстрого создания приложений управления базами данных;
- СУБД полностью русифицирована;
- возможность использования OLE технологии;
- интегрированность с пакетами Microsoft Office;
- поддержка WWW-идеологии (только в Access 97);
- визуальная технология позволяет постоянно видеть результаты своих действий и корректировать их; кроме того, работа с конструктором форм может существенно облегчить дальнейшее изучение таких систем программирования, как Visual Basic или Delphi;
- широко и наглядно представлена справочная система;
- наличие большого набора «мастеров» по разработке объектов. Базовые объекты СУБД Access.

Таблица – объект, который используется для хранения данных. Каждая таблица содержит информацию по объектам предметной области определённого типа. Поля таблицы служат для хранения различных характеристик этих объектов, а каждая запись – о конкретном объекте. Для каждой таблицы можно определить первичные ключи, один или несколько индексов, ускоряющих доступ к данным. Таблица – это базовый объект Microsoft Access, все остальные объекты являются производными и создаются только на базе ранее подготовленных таблиц.

Запрос – позволяет пользователю получить нужные данные из одной или нескольких таблиц. Для определения запроса можно использовать бланк запроса по образцу или написать инструкцию SQL. Можно создать запросы на выборку, обновление, удаление, добавление данных. Можно также создать новую таблицу, используя данные из одной или нескольких существующих таблиц.

Отчёт - объект, который предназначен для форматирования, вычисления итогов и печати выбранных данных..

Форма – объект, который предназначен для ввода данных, отображения их на экране или управления работой приложения.

Макрос – объект, который представляет собой структурированное описание действий, которые должна выполнить СУБД Access в ответ на определённое событие

(открыть форму, проверить значение некоторого поля, выполнить запрос, просмотреть/напечатать отчет, запустить другой макрос).

Модуль – объект, который содержит программы на языке Visual Basic для приложений, позволяющих разбить некоторый процесс на несколько небольших процедур.

Прежде чем приступить к созданию реляционной базы данных, необходимо продумать ее проект. Проект представляет собой модель будущей БД, состоящей из объектов и их связей, необходимых для выполнения поставленных задач. Процесс проектирования включает, прежде всего, определение перечня необходимых таблиц и задание их структуры, а также установки типа связей между этими таблицами.

4 Однотабличные и многотабличные базы данных. Межтабличные связи

Достаточно часто встречается ситуация, когда хранить всю базу данных в одной таблице неудобно и нерационально. Таблица может содержать слишком большое количество полей, что неудобно пользователю. Различные записи при этом во многих полях дублируют друг друга, что увеличивает информационный объем базы данных и замедляет процедуры ее обработки. БД, состоящая из нескольких таблиц, называется многотабличной.

После создания различных таблиц, содержащих данные, относящиеся к различным аспектам базы данных, необходимо обеспечить целостность базы данных. Для этого необходимо связать таблицы между собой.

Всего имеются три типа связей (отношений):

- один-к-одному,
- многие-ко-многим,
- один-ко-многим.

Если одной записи в первой таблице могут соответствовать несколько записей во второй таблице, и, наоборот, одной записи во второй таблице — несколько записей в первой таблице, то реализуется связь в отношении «многие-ко-многим». Две таблицы, находящиеся в отношении «многие-ко-многим» могут быть связаны только с помощью третьей (связующей) таблицы. Связь между таблицами устанавливает отношения между совпадающими значениями в полях с одинаковыми именами. С ключевым полем главной таблицы (первичный ключ), связывается одноименное поле подчиненной таблицы (внешний ключ).

При связи с отношением «один-ко-многим» каждой записи в одной (главной) таблице могут соответствовать несколько записей в другой (подчиненной) таблице, а запись в подчиненной таблице не может иметь более одной соответствующей ей записи в главной таблице. Межтабличная связь обеспечивает целостность данных. Связанные таблицы представляют собой единую базу данных, в которой можно создавать новые таблицы, а также запросы и отчеты, содержащие данные из связанных таблиц.

5 Создание запросов. Формирование отчетов

Запросы предназначены для поиска в базе данных информации, отвечающей определенным критериям. Найденные записи, называемые результатами запроса, можно просматривать, редактировать и анализировать различными способами. Кроме того, результаты запроса могут использоваться в качестве основы для создания других объектов Access. При выполнении запроса Access находит в указанных таблицах данные, соответствующие заданному условию, и отображает их в табличном виде.

Запрос — это команда на выполнение определенного вида манипулирования данными.

Манипулирование данными — это действия, выполняемые над информацией, хранящейся в базе данных. К ним относятся выборка данных по некоторым условиям, сортировка данных, обновление, удаление и добавление данных. Выполнение этих действий производится с помощью запросов.

Виды запросов - запросы на выборку, параметрические и перекрестные запросы.

Условие отбора — это логическое выражение, которое должно быть истинным для отбираемых записей БД.

Отчеты служат для форматированного вывода данных на печатающие устройства и, соответственно, при этом должны учитывать параметры принтера и параметры используемой бумаги.

Здесь существуют средства автоматического, автоматизированного и ручного проектирования. Средства автоматического проектирования реализованы автоотчетами (База данных > Создать > Новый отчет > Автоотчет в столбец). Кроме автоотчетов «в столбец» существуют «ленточные» автоотчеты.

Средством автоматизированного создания отчетов является Мастер отчетов. Он запускается двойным щелчком на значке Создание отчета с помощью мастера в окне База данных. Мастер отчетов работает в шесть этапов. При его работе выполняется выбор базовых таблиц или запросов, на которых отчет базируется, выбор полей, отображаемых в отчете, выбор полей группировки, выбор полей и методов сортировки, выбор формы печатного макета и стиля оформления.

Важной особенностью отчетов является наличие средства для вставки в область верхнего или нижнего колонтитула текущего номера страницы и полного количества страниц. Эту операцию выполняют с помощью диалогового окна Номера страниц (Вставка > Номера страниц)

1.14 Лекция №14 (2 часа).

Тема: «Модели решения функциональных и вычислительных задач»

1.14.1 Вопросы лекции:

1. Компьютерное моделирование как метод научного познания.
2. Классификация моделей.
3. Виды моделей.
4. Этапы компьютерного моделирования.

1.14.2 Краткое содержание вопросов:

1 Компьютерное моделирование как метод научного познания

Компьютерное моделирование, возникшее как одно из направлений математического моделирования с развитием информационных компьютерных технологий стало самостоятельной и важной областью применения компьютеров. В настоящее время компьютерное моделирование в научных и практических исследованиях является одним из основных методов познания. Без компьютерного моделирования сейчас невозможно решение крупных научных и экономических задач. Выработана технология исследования сложных проблем, основанная на построении и анализе с помощью вычислительной техники математической модели изучаемого объекта. Такой метод исследования называется вычислительным экспериментом. Вычислительный эксперимент применяется практически во всех отраслях науки - в физике, химии, астрономии, биологии, экологии, даже в таких сугубо гуманитарных науках как психология, лингвистика и филология, кроме научных областей вычислительные эксперименты широко применяются в экономике, в социологии, в промышленности, в управлении. Проведение вычислительного эксперимента имеет ряд преимуществ перед так называемым натурным экспериментом:

- для ВЭ не требуется сложного лабораторного оборудования;
- существенное сокращение временных затрат на эксперимент;
- возможность свободного управления параметрами, произвольного их изменения, вплоть до придания им нереальных, неправдоподобных значений;

- возможность проведения вычислительного эксперимента там, где натурный эксперимент невозможен из-за удаленности исследуемого явления в пространстве (астрономия) либо из-за его значительной растянутости во времени (биология), либо из-за возможности внесения необратимых изменений в изучаемый процесс.

Моделирование — метод научного исследования явлений, процессов, объектов, устройств или систем (обобщенно – объектов исследований), основанный на построении и изучении моделей с целью получения новых знаний, совершенствования характеристик объектов исследований или управления ими.

Модель — материальный объект или образ (мысленный или условный: гипотеза, идея, абстракция, изображение, описание, схема, формула, чертеж, план, карта, блок-схема алгоритма, ноты и т.п.), которые упрощенно отображают самые существенные свойства объекта исследования.

Любая модель всегда проще реального объекта и отображает лишь часть его самых существенных черт, основных элементов и связей. По этой причине для одного объекта исследования существует множество различных моделей. Вид модели зависит от выбранной цели моделирования.

Потребность в создании и использовании моделей связана с тем, что исследовать многие реальные явления и объекты сложно или дорого, а порой вовсе невозможно. Например, безумно экспериментально изучать, к чему приведет мировая термоядерная война. Опасны эксперименты с реальными реакторами на атомных электростанциях. Неразумны опыты с радиоаппаратурой при предельных значениях напряжения питания и окружающей температуры.

В основе термина «модель» лежит латинское слово *modulus* — мера, образец. Модель – это заместитель реального объекта исследования. Модель всегда проще исследуемого объекта. При изучении сложных явлений, процессов, объектов не удается учесть полную совокупность всех элементов и связей, определяющих их свойства.

Но все элементы и связи в создаваемой модели и не следует учитывать. Нужно лишь выделить наиболее характерные, доминирующие составляющие, которые в подавляющей степени определяют основные свойства объекта исследования. В результате объект исследования заменяется некоторым упрощенным подобием, но обладающим характерными, главными свойствами, аналогичными свойствам объекта исследования. Появившийся вследствие проведенной подмены новый объект (или абстракция) принято называть моделью объекта исследования.

2 Классификация моделей

В зависимости от средств построения различают следующие классы моделей:

- словесные или описательные модели их также в некоторой литературе называют вербальными или текстовыми моделями (например, милицкий протокол с места происшествия, стихотворение Лермонтова "Тиха украинская ночь");
- натурные модели (макет Солнечной системы, игрушечный кораблик);
- абстрактные или знаковые модели. Интересующие нас математические модели явлений и компьютерные модели относятся как раз к этому классу.

Можно классифицировать модели по предметной области:

- физические модели,
- биологические,
- социологические,
- экономические и т.д.
- Классификация модели по применяемому математическому аппарату:
- модели, основанные на применении обыкновенных дифференциальных уравнений;
- модели, основанные на применении уравнений в частных производных;
- вероятностные модели и т.д.

Также можно классифицировать модели по цели моделирования. В зависимости от целей моделирования различают:

- дескриптивные модели (описательные) описывают моделируемые объекты и явления и как бы фиксируют сведения человека о них. Примером может служить модель Солнечной системы, или модель движения кометы, в которой мы моделируем траекторию ее полета, расстояние, на котором она пройдет от Земли. У нас нет никаких возможностей повлиять на движение кометы или движение планет Солнечной системы;
- оптимизационные модели служат для поиска наилучших решений при соблюдении определенных условий и ограничений. В этом случае в модель входит один или несколько параметров, доступных нашему влиянию, например, известная задача коммивояжера, оптимизируя его маршрут, мы снижаем стоимость перевозок;
- игровые модели (компьютерные игры);
- обучающие модели (всевозможные тренажеры);
- имитационные модели (модели, в которых сделана попытка более или менее полного и достоверного воспроизведения некоторого реального процесса, например, моделирование движения молекул в газе, поведение колонии микробов и т.д.).

Существует также классификация моделей в зависимости от их изменения во времени. Различают:

- статические модели - неизменные во времени;
- динамические модели - состояние которых меняется со временем.

3 Виды моделей

Процесс моделирования начинается с создания концептуальной модели.

Концептуальная модель (содержательная) — это абстрактная модель, определяющая структуру системы (элементы и связи).

В концептуальной модели обычно в словесной (вербальной) форме приводятся самые главные сведения об объекте исследования, основных элементах и важнейших связях между элементами. Процесс создания концептуальной модели в настоящее время не формализован: не существует точных правил ее создания.

Основная проблема при создании концептуальной модели заключается в нахождении компромисса между компактностью модели и ее точностью (адекватностью).

Термин «адекватна» (происходит от лат. *adaequatus* — приравненный, равный) означает верное воспроизведение в модели связей и отношений объективного мира. Этим термином характеризуют качество созданной модели.

Процесс создания концептуальной модели, вероятно, никогда не сможет быть полностью формализован. Трудно придумать набор простых правил, выполняя которые, можно создать хорошую концептуальную модель. Именно в связи с этим иногда говорят, что моделирование является не только наукой, но и искусством.

Концептуальную модель, содержащую основные сведения об объекте исследований, порой называют информационной моделью.

В научной литературе широко используется термин математическая модель (ММ). ММ — описание объекта исследования, выполненное с помощью математической символики.

Математическая модель представляет собой совокупность формул, уравнений, неравенств, логических условий и т.д. Используемые в ММ математические соотношения определяют процесс изменения состояния объекта исследования в зависимости от его параметров, входных сигналов, начальных условий и времени. По существу, вся математика создана для формирования математических моделей.

Математическое моделирование — метод изучения объекта исследования, основанный на создании его математической модели и использовании её для получения новых знаний, совершенствования объекта исследования или управления объектом.

Математическое моделирование можно подразделить на аналитическое и компьютерное (машинное) моделирование.

При компьютерном моделировании математическая модель создается и анализируется с помощью вычислительной техники. В этом случае нередко используются приближенные (численные) методы расчета. При компьютерном моделировании используются наиболее прогрессивные информационные технологии, например, виртуальная реальность. При этом моделирование медицинской операции вызывает иллюзию реально происходящего события. Моделирование игровых ситуаций сопровождается мультимедийными эффектами (звуками, видеоэффектами). Компьютерная модель – модель, реализованная на одном из языков программирования (программа для ЭВМ).

Рассмотрим еще два понятия: полная математическая модель и макромодель. Полная математическая модель — это модель, отражающая состояния как моделируемой системы, так и всех ее внутренних элементов. Полная ММ электронного усилителя позволяет определить потенциалы всех узлов схемы и токи через все радиоэлементы (т. е. можно определить фазовые переменные для всех элементов модели).

Макромодель проще полной математической модели.

Макромодель адекватна в отношении внешних свойств объекта исследования. Однако, в отличие от полной математической модели, макромодель не описывает внутреннее состояние отдельных элементов. Например, макромодель радиоэлектронного усилителя определяет, как изменяются сигналы на входах (X и Z) и выходе (Y) устройства, но не дает сведения о том, как сигналы изменяются на каждом радиоэлементе (резисторах, транзисторах и т. д.), находящемся внутри усилителя. Другими словами, полная математическая модель описывает и систему, и элементы, входящие в систему. Макромодель же описывает только систему моделирования. Макромодель представляет объект исследования в виде «черного ящика», содержимое которого неизвестно.

Модель называется статической, если среди входных воздействий X и Z нет параметров, зависящих от времени. Статическая модель в каждый момент времени дает лишь застывшую «фотографию» объекта исследования, ее срез. С помощью статических моделей удобно изучать, например, работу логических элементов.

Модель называется динамической, если входные воздействия изменяются во времени, или нужно определить, как изменяется состояние объекта исследования с изменением времени. С помощью динамических моделей исследуют, в частности, переходные процессы в электрических цепях.

Модель называется детерминированной, если каждому набору входных параметров всегда соответствует единственный набор выходных параметров. В противном случае модель называется недетерминированной (стохастической, вероятностной). В стохастических моделях используются генераторы случайных чисел с различными законами распределения.

При моделировании часто оперируют следующими категориями: элемент и система. Рассмотрим эти понятия.

Элемент — составная часть сложного объекта исследования.

Система — целое, составленное из частей. Другими словами, система — это множество элементов, находящихся в отношениях и связях друг с другом, образующих определенную целостность, единство.

В зависимости от уровня моделирования, понятия «элемент» и «система» получают различное смысловое наполнение.

Например, в глобальной вычислительной сети элементами являются компьютеры и каналы связи. В другом случае, при моделировании ЭВМ как сложной системы, в качестве элементов можно выделить процессор, устройства ввода-вывода, память. Если системой считать процессор, то элементами станут АЛУ, регистры, мультиплексоры, дешифраторы, триггеры и т. п. В случае исследования триггера элементами будут резисторы и транзисторы.

4 Этапы компьютерного моделирования

1 этап - построение содержательной модели - предполагает следующие действия:

- определить: что является целью моделирования и что является искомым результатом

- выбрать свойства объекта, влияющие на искомый результат
- определить зависимости результата от выбранных свойств

2 этап - построение информационной модели (формализация)

- ввести условные обозначения (формальный язык)
- описать зависимости результата от выбранных свойств на формальном языке (с помощью формул, графиков, таблиц и т.д.)

Таким образом, можно сказать, что формализация - это описание модели объекта с помощью какого-либо *формального языка*

3 этап - компьютерный эксперимент

- объявление переменных, используемых в компьютерной программе
- описание зависимости выходных (или промежуточных) переменных от входных переменных и констант
- разработка блок-схемы алгоритма
- ввод программы в компьютер, отладка и контрольный запуск (задаются такие значения входных переменных, для которых заранее известны значения выходных переменных)
- запуск программы и исследование полученного результата

Если цель эксперимента не достигнута, то производится уточнение модели и повторно выполняются второй и третий этапы моделирования

1.15 Лекция №15 (2 часа).

Тема: «Алгоритмизация и программирование»

1.15.1 Вопросы лекции:

1. Понятие алгоритма. Исполнитель алгоритма
2. Основные свойства алгоритмов.
3. Формы записи алгоритмов
4. Графические обозначения (обозначения на блок-схемах)
5. Технологии программирования.
6. Структурное программирование.
7. Объектно-ориентированное программирование и визуальное программирование.
8. Понятие о языке программирования. Уровни языков программирования
9. Языки программирования высокого уровня.
10. Трансляция и отладка программы.
11. Эксплуатация программы.

1.15.2 Краткое содержание вопросов:

1 Понятие алгоритма. Исполнитель алгоритма

Алгоритм — заранее заданное понятное и точное предписание возможному исполнителю совершить определенную последовательность действий для получения решения задачи за конечное число шагов. Это — не определение в математическом смысле слова, а, скорее, описание интуитивного понятия алгоритма, раскрывающее его сущность.

Исполнитель алгоритма — это некоторая абстрактная или реальная (техническая, биологическая или биотехническая) система, способная выполнить действия, предписываемые алгоритмом.

Среда (или обстановка) — это "место обитания" исполнителя. Каждый исполнитель может выполнять команды только из некоторого строго заданного списка — системы команд исполнителя. Для каждой команды должны быть заданы условия применимости (в каких состояниях среды может быть выполнена команда) и описаны результаты выполнения команды. Отказы исполнителя возникают, если команда вызывается при недопустимом для нее состоянии среды. В информатике универсальным исполнителем алгоритмов является компьютер.

2 Основные свойства алгоритмов

Понятность для исполнителя — исполнитель алгоритма должен понимать, как его выполнять. Иными словами, имея алгоритм и произвольный вариант исходных данных, исполнитель должен знать, как надо действовать для выполнения этого алгоритма.

Дискретность (прерывность, раздельность) — алгоритм должен представлять процесс решения задачи как последовательное выполнение простых (или ранее определенных) шагов (этапов).

Определенность — каждое правило алгоритма должно быть четким, однозначным и не оставлять места для произвола. Благодаря этому свойству выполнение алгоритма носит механический характер и не требует никаких дополнительных указаний или сведений о решаемой задаче.

Результативность (или конечность) состоит в том, что за конечное число шагов алгоритм либо должен приводить к решению задачи, либо после конечного числа шагов останавливаться из-за невозможности получить решение с выдачей соответствующего сообщения, либо неограниченно продолжаться в течение времени, отведенного для исполнения алгоритма, с выдачей промежуточных результатов.

Массовость означает, что алгоритм решения задачи разрабатывается в общем виде, т.е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными. При этом исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.

3 Формы записи алгоритмов

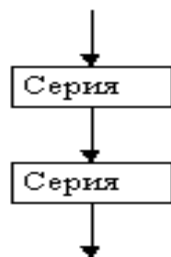
На практике наиболее распространены следующие формы представления алгоритмов:

- словесная (запись на естественном языке);
- графическая (изображения из графических символов);
- псевдокоды (полуформализованные описания алгоритмов на условном алгоритмическом языке, включающие в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические обозначения и др.);
- программная (тексты на языках программирования).

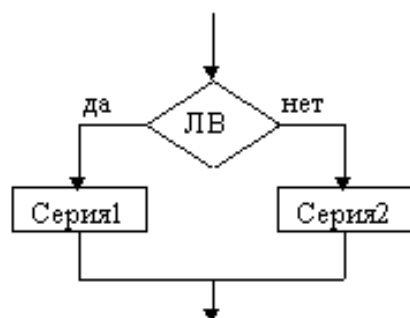
Словесный способ записи алгоритмов представляет собой описание последовательных этапов обработки данных. Алгоритм задается в произвольном изложении на естественном языке.

4 Графические обозначения (обозначения на блок-схемах)

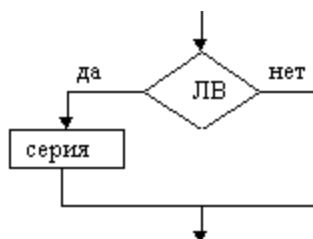
При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий ((См. Рис. 13.1 – 13.6)



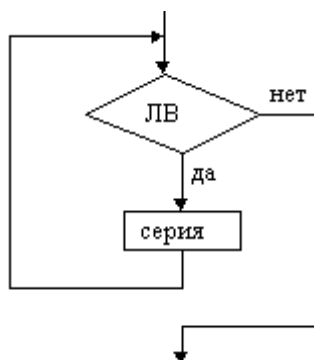
Р и с у н о к Ошибка! Текст указанного стиля в документе отсутствует..1. --
Структура “следование”



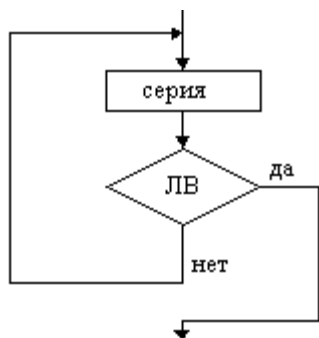
Р и с у н о к Ошибка! Текст указанного стиля в документе отсутствует..2 --
Полная развилка



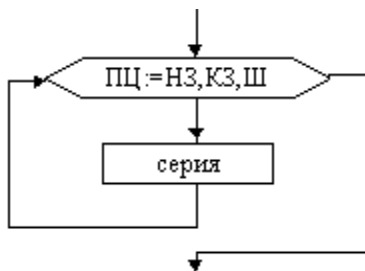
Р и с у н о к Ошибка! Текст указанного стиля в документе отсутствует..3-
-Неполная развилка



Р и с у н о к Ошибка! Текст указанного стиля в документе отсутствует..4. --
Цикл с предусловие (цикл ПОКА)



Р и с у н о к **Ошибка! Текст указанного стиля в документе отсутствует..5** --
Цикл с постусловием (цикл ДО)



Р и с у н о к **Ошибка! Текст указанного стиля в документе отсутствует..6** --
Цикл с параметром

Такое графическое представление называется схемой алгоритма или блок-схемой. В блок-схеме каждому типу действий (вводу исходных данных, вычислению значений выражений, проверке условий, управлению повторением действий, окончанию обработки и т.п.) соответствует геометрическая фигура, представленная в виде блочного символа.

Начало и конец алгоритма на блок-схемах обозначают овалом, вводимые и выводимые переменные записываются в параллелограмме.

В примерах мы будем использовать запись алгоритмов с помощью блок-схем и словесное описание.

Для проверки работоспособности алгоритма необходимо задать значения входных переменных, вычислить конечный результат по алгоритму и сравнить с результатом ручного счета.

При тестировании алгоритмов с развилкой необходимо подбирать такие исходные данные, чтобы можно было проверить все ветви.

Простейшие задачи имеют линейный алгоритм решения. Это означает, что он не содержит проверок условий и повторений.

Достаточно часто то или иное действие должно быть выполнено в зависимости от значения логического выражения, выступающего в качестве условия. В таких случаях используется развилка.

Для проверки работоспособности алгоритма необходимо задать значения входных переменных, вычислить конечный результат по алгоритму и сравнить с результатом ручного счета. При тестировании алгоритмов с развилкой необходимо подбирать такие исходные данные, чтобы можно было проверить все ветви.

Если какие-либо операторы необходимо выполнить несколько раз, то их не переписывают каждый раз заново, а организуют цикл. Командой повторения или циклом называется такая форма организации действий, при которой одна и та же последовательность действий повторяется до тех пор, пока сохраняется значение некоторого логического выражения. При изменении значения логического выражения на противоположное повторения прекращаются (цикл завершается).

Для организации цикла необходимо выполнить следующие действия:

- перед началом цикла задать начальное значение параметра;
- внутри цикла изменять параметр цикла с помощью оператора присваивания;
- проверять условие повторения или окончания цикла;
- управлять циклом, т.е. переходить к его началу, если он не закончен, или выходить из цикла в противном случае.

Различают циклы с известным числом повторений (цикл с параметром) и итерационные (с пред- и постусловием).

В цикле с известным числом повторений параметр изменяется в заданном диапазоне. Если в цикле изменяется простая переменная, то она является параметром цикла; если в цикле изменяется переменная с индексом, то индекс этой переменной является параметром цикла.

Если в операторе цикла с параметром начальное или конечное значение параметра заданы переменными или выражениями, то значения этих переменных должны быть определены в программе до оператора цикла. Не следует внутри цикла изменять параметр цикла, его начальное и конечное значения с помощью операторов присваивания или ввода.

5 Технологии программирования

При создании первых компьютеров возникла проблема выбора: где хранить алгоритм и где хранить данные. У этого вопроса имеется два различных ответа:

- 1) алгоритм и данные хранятся в одном устройстве;
- 2) алгоритм и данные хранятся в разных устройствах.

Принцип работы компьютера по первому способу называется архитектурой фон Неймана и применяется практически во всех современных компьютерах: алгоритм и данные при выполнении программы хранятся в оперативной памяти.

Проектирование сверху вниз

Основной метод создания алгоритмов — проектирование, или программирование, сверху вниз, или пошаговая детализация. Он заключается в разбиении исходной задачи на последовательность нескольких меньших подзадач. Эти подзадачи, в свою очередь, тоже распадаются на подзадачи и т. д. до тех пор, пока не останутся только элементарные алгоритмы.

Например, чтобы написать число 512, сначала пишут цифру 5, затем 1 и, наконец, 2. При этом цифры рисуют, последовательно прорисовывая линии, из которых они состоят. Принтер напечатает это число точками.

Модульность алгоритмов, головная программа

При программировании сверху вниз алгоритмы и данные делятся на относительно независимые части, называемые модулями. Некоторые из модулей являются стандартными и поставляются в составе языков программирования, например, вычисление элементарных математических функций квадратный корень, логарифм, синус и т. д. Но главные модули все равно приходится проектировать программистам.

Таким образом, алгоритм является деревом модулей: одни модули вызывают другие модули, начиная с самого верхнего первого модуля, называемого корневым модулем, или головной программой.

Принцип черного ящика. При проектировании сколько-нибудь больших алгоритмов невозможно держать в памяти одновременно детали всех модулей алгоритма. Если модуль составлен правильно, то с ним можно обращаться как с черным ящиком.

Принцип черного ящика означает, что не имеет значения, как модуль выполняет свою функцию, какие алгоритмы скрыты у него внутри. Это не важно для остальных модулей алгоритма. Для модулей, которые обращаются к этому модулю, имеет значение только следующее:

- 1) какова функция модуля, т. е. что он делает;
- 2) описание входных и выходных данных модуля.

Правильное проектирование алгоритмов позволяет абстрагироваться от внутренней структуры модулей и рассматривать при сборке полного алгоритма только функции модулей.

6 Структурное программирование

Структурное программирование является технологией простого и прозрачного создания алгоритмов. Это единственный способ строить алгоритмы быстро и в последующем легко вносить в них изменения. Структурное программирование позволяет проектировать алгоритмы только из трех элементарных алгоритмов.

Эти три элементарных алгоритма являются не только самыми нижними модулями на дереве модулей. Каждый модуль является иерархией этих элементарных алгоритмов. Алгоритм, состоящий только из этих трех элементарных алгоритмов, присутствующих на всех его уровнях, называется структурным.

Основная теорема структурного программирования утверждает, что любой алгоритм можно преобразовать к структурному виду.

Тремя элементарными структурными алгоритмами являются следующие.

1. Следование, или цепочка, или составная инструкция.
2. Выбор, или ветвление, или условная инструкция.
3. Цикл, или возврат, или циклическая инструкция.

7 Объектно-ориентированное и визуальное программирование

ООП - концепция, которая в свое время произвела настоящую революцию в программировании. До появления ООП программа представляла собой логически единый код с ярко выраженными функциональными зависимостями (процедурный подход).

ООП предполагает, что приложение строится из набора независимых по своему внутреннему устройству модулей (для простоты понимания, представьте себе детали из которых можно собрать разные механизмы, по разному объединяя некоторые из них). При этом одной независимой части ничего неизвестно о внутреннем устройстве другой.

Приложения на языках объектно-ориентированного программирования Visual Basic и Visual Basic for Applications строятся из объектов, подобно тому, как из блоков и различных деталей строятся дома. Программные библиотеки готовых объектов входят в эти системы программирования, причем языки Visual Basic и VBA различаются между собой, главным образом, составом программных библиотек.

Системы объектно-ориентированного программирования дают возможность визуализировать процесс создания графического интерфейса разрабатываемого приложения, то есть позволяют создавать объекты и задавать значения их свойств с помощью диалоговых окон системы программирования.

Взаимодействие программных объектов между собой и их изменения описываются с помощью программного кода. Создание программного кода в объектно-ориентированном программировании базируется на использовании алгоритмических структур различных типов (линейной, ветвления, цикла), исполнителями которых выступают программные объекты.

Классы объектов, экземпляры класса и семейства объектов

Основной единицей в объектно-ориентированном программировании является программный объект, который объединяет в себе как описывающие его данные (свойства), так и средства обработки этих данных (методы). Если говорить образно, то объекты — это «существительные», свойства объекта — это «прилагательные», а методы объекта — это «глаголы».

Программные объекты обладают свойствами, могут использовать методы и реагируют на события. Классы объектов являются «шаблонами», определяющими наборы свойств, методов и событий. По этим шаблонам создаются объекты. В языке Visual Basic основными являются классы объектов, реализующие графический интерфейс приложения. В языке VBA еще используются более ста различных классов объектов, которые существуют в среде Windows&Office. В обоих языках существуют возможности

подключения дополнительных библиотек программных объектов, а также создания новых классов объектов самим программистом.

Каждый из классов обладает специфическим набором свойств, методов и событий. Специфика ООП заметно повышает эффективность труда программистов и позволяет им создавать более мощные, масштабируемые и эффективные приложения. Объектно-ориентированное программирование основано на: инкапсуляции; полиморфизме; наследовании.

Инкапсуляция - это механизм, объединяющий данные и обрабатывающий их код как единое целое. Многие преимущества ООП обусловлены одним из его фундаментальных принципов — инкапсуляцией. Инкапсуляцией называется включение различных мелких элементов в более крупный объект, в результате чего программист работает непосредственно с этим объектом. Это приводит к упрощению программы, поскольку из нее исключаются второстепенные детали. Инкапсуляцию можно сравнить с работой автомобиля с точки зрения типичного водителя. Многие водители не разбираются в подробностях внутреннего устройства машины, но при этом управляют ею именно так, как было задумано. Пусть они не знают, как устроен двигатель, тормоз или рулевое управление, — существует специальный интерфейс, который автоматизирует и упрощает эти сложные операции. Сказанное также относится к инкапсуляции и ООП — многие подробности "внутреннего устройства" скрываются от пользователя, что позволяет ему сосредоточиться на решении конкретных задач. В ООП эта возможность обеспечивается классами, объектами и различными средствами выражения иерархических связей между ними

Полиморфизм позволяет использовать одни и те же имена для похожих, но технически разных задач. Главным в полиморфизме является то, что он позволяет манипулировать объектами путем создания стандартных интерфейсов для схожих действий. Полиморфизм значительно облегчает написание сложных программ.

Наследование позволяет одному объекту приобретать свойства другого объекта, не путайте с копированием объектов. При копировании создается точная копия объекта, а при наследовании точная копия дополняется уникальными свойствами, которые характерны только для производного объекта.

Визуальное программирование существенно облегчает программирование для графического интерфейса типа Windows, который состоит из множества графических объектов: кнопок, окошек, меню и т. д. Система визуального программирования предоставляет программисту:

1) готовую объектную визуальную модель, содержащую множество графических диалоговых объектов (кнопки, окошки, меню и т. д.) и программных модулей, которые их реализуют;

2) среду визуального программирования, в которой графические диалоговые объекты, которые будут определять интерфейс программы, просто размещаются на экране мышью.

8 Понятие о языке программирования. Уровни языков программирования

Алгоритм, предназначенный для исполнения на компьютере, должен быть записан на понятном ему языке. Следовательно, язык для записи алгоритмов должен быть формализован. Такой язык принято называть языком программирования, а запись алгоритма на этом языке — программой для компьютера.

В настоящее время в мире существует несколько сотен реально используемых языков программирования. Для каждого есть своя область.

Любой алгоритм, есть последовательность предписаний, выполнив которые можно за конечное число шагов перейти от исходных данных к результату. В зависимости от степени детализации предписаний обычно определяется уровень языка программирования — чем меньше детализация, тем выше уровень языка.

По этому критерию можно выделить следующие уровни языков программирования:

- машинные;
- машинно-ориентированные (ассемблеры);
- машинно-независимые (языки высокого уровня).

Машинные языки и машинно-ориентированные языки — это языки низкого уровня, требующие указания мелких деталей процесса обработки данных.

9 Языки программирования высокого уровня

Языки же высокого уровня имитируют естественные языки, используя некоторые слова разговорного языка и общепринятые математические символы. Эти языки более удобны для человека.

Языки высокого уровня делятся на:

- процедурные (алгоритмические) (Basic, Pascal, C и др.), которые предназначены для однозначного описания алгоритмов; для решения задачи процедурные языки требуют в той или иной форме явно записать процедуру ее решения;
- логические (Prolog, Lisp и др.), которые ориентированы не на разработку алгоритма решения задачи, а на систематическое и формализованное описание задачи с тем, чтобы решение следовало из составленного описания;
- объектно-ориентированные (Object Pascal, C++, Java, FOXPRO, VISUAL BASIC и др.), в основе которых лежит понятие объекта, сочетающего в себе данные и действия над ними. Программа на объектно-ориентированном языке, решая некоторую задачу, по сути описывает часть мира, относящуюся к этой задаче. Описание действительности в форме системы взаимодействующих объектов естественнее, чем в форме взаимодействующих процедур.

Эффективность использования языков программирования можно значительно повысить, если знать, для решения каких задач предназначен тот или иной из них.

Задачи искусственного интеллекта: Lisp, Prolog, Multilisp, Commonlisp, Planner, QA4, FRL, KRL, Qlisp, Рефал.

Параллельные вычисления: Fun, APL, Afl, ParAfl, ML, SML, PPL/1, Hope, Miranda, Occam, PFOR, Glypnir, Actus, параллельный Кобол, OBC – ЛЯПАС, OBC – МНМОКОД, OBC – АЛГОЛ, OBC – Фортран, язык PA (1), язык PA (G).

Задачи вычислительной математики и физики: Occam, PFOR, Glypnir, Actus, параллельный Кобол, OBC – ЛЯПАС, OBC – МНМОКОД, OBC – АЛГОЛ, OBC – Фортран, язык PA (1), язык PA (G)

Разработка интерфейса: Форт, C, C++, Ассемблер, Макроассемблер, Ява, РПГ, SIMULA – 67, Oak SMALLTALK.

Разработка программ – оболочек, разработка систем: Форт, C, C++, Ассемблер, Макроассемблер, Ява, РПГ, SIMULA – 67, Oak SMALLTALK.

Задачи вычислительного характера: Алгол, Фортран, Кобол, Ада, ПЛ/1, Фокал, Бейсик, Паскаль.

Оформление документов, обработка больших текстовых файлов, организация виртуальных трехмерных интерфейсов в Интернете, разработка баз данных:

HTML, Perl, Tcl/Tk, VRML, SQL, PL/SQL, INFORMIX 4GL, NATURAL, DDL, DSDL, SEQUEL, QBE, ISBL.

10 Трансляция и отладка программы

Перевод программы на машинный язык осуществляется с помощью программы - транслятора. Для каждого языка программирования предназначен свой транслятор. Транслятор выполняет следующие основные функции.

Во-первых, производится *синтаксический анализ* исходного текста программы, т.е. выделяются все основные компоненты, из которых состоит программа. В процессе синтаксического анализа выявляются возможные ошибки в исходном тексте. Такие ошибки называют синтаксическими (связанными с неправильной записью программы на

исходном языке программирования, например с использованием недопустимых операторов или недопустимых последовательностей символов, отсутствием необходимых описаний объектов языка и т.п.).

Во-вторых, производится *генерация машинной программы* - непосредственная трансляция исходной программы на машинный язык.

В-третьих, выдается на печать листинг - текст исходной программы вместе со списком всех обнаруженных синтаксических ошибок. По указанию пользователя наряду с листингом исходной программы может быть также отпечатан листинг полученной в результате трансляции машинной программы. Кроме того, по указанию пользователя полученная программа может быть записана в архив операционной системы для дальнейшего использования или передана операционной системе для исполнения.

В результате трансляции получается программа, готовая к исполнению на ЭВМ. Однако в процессе разработки программы могли быть допущены ошибки, которые сохранились и в результирующей программе. Такие ошибки называют содержательными. Эти ошибки принципиально не могут быть устранены на стадии трансляции, так как для их обнаружения необходима информация о существовании решаемой задачи. Поэтому их может устранить только сам пользователь.

Ошибки могут быть допущены по самым разным причинам. Часть ошибок может быть устранена еще на стадии разработки программы, если параллельно с разработкой программы строится также доказательство ее правильности. Но, во-первых, методы аналитического доказательства правильности программы еще недостаточно разработаны, а во-вторых, ошибка могла быть допущена еще на стадии постановки задачи, в результате чего может оказаться, что разработанная программа хотя и работает верно, но решает совсем не ту задачу.

Поэтому любая программа до ее практического использования должна пройти этап отладки. Цель отладки состоит в том, чтобы выявить и устранить ошибки, допущенные на предыдущих этапах, и получить правильную программу, к результатам работы которой можно относиться с доверием.

Суть отладки состоит в том, что пользователь подготавливает систему тестов, с помощью которой проверяется работа программы в различных возможных режимах. Каждый тест содержит набор исходных данных, для которых известен результат. Если в результате работы программы с данным тестом получаются результаты, отличные от ожидаемых, то это свидетельствует о наличии ошибки. Тест стараются выбрать так, чтобы он помог не только установить сам факт ошибки, но и локализовать эту ошибку, т.е. по возможности сузить подозреваемую часть программы, содержащую ошибку.

Для облегчения отладки в состав операционных систем включают специальные сервисные программы - отладчики. При отладке пользователь формулирует задание отладчику, а отладчик выполняет это задание и выдает пользователю необходимую информацию о поведении отлаживаемой программы. Для повышения доверия к результатам необходимы хорошо структурированные программы, поведение которых можно прогнозировать и аналитическими методами. Именно поэтому аналитические методы доказательства правильности программ получают все большее развитие.

11 Эксплуатация программы

После завершения отладки программы передаются в вычислительный центр и используются для решения реальных задач. Эксплуатация таких программ в вычислительном центре осуществляется профессиональными программистами, входящими в группу сопровождения программ.

Необходимость создания специальной службы сопровождения вызвана тем, что программа может эксплуатироваться длительное время. За это время многое может измениться, например, могут измениться требования к решаемой задаче или операционная система, под управлением которой выполняется данная программа. Это может

потребовать внесения изменений в программу, что, в свою очередь, потребует новой отладки. Такие работы и выполняют теперь программисты из группы сопровождения.

1.16 Лекция №16 (2 часа).

Тема: «Локальные и глобальные сети ЭВМ»

1.16.1 Вопросы лекции:

1. Передача информации. Локальные компьютерные сети
2. Аппаратное обеспечение сети. Клиент и сервер
3. Классификация сетей. Типы сетей. Сетевые топологии
4. Глобальная сеть. Технические и программные ресурсы Интернета

1.16.2 Краткое содержание вопросов:

1 Передача информации. Локальные компьютерные сети

Обмен информацией производится по каналам передачи информации. Каналы передачи информации могут использовать различные физические принципы. Общая схема передачи информации включает в себя отправителя информации, канал передачи информации и получателя информации. Если производится двусторонний обмен информацией, то отправитель и получатель информации могут меняться ролями.

Основной характеристикой каналов передачи информации является их пропускная способность (скорость передачи информации). Пропускная способность канала равна количеству информации, которое может передаваться по нему » единицу времени.

Локальная компьютерная сеть – это система взаимосвязанных компьютеров, работающих в пределах одного помещения, здания, одной организации. Локальные сети делятся на учрежденческие (офисные сети фирм, сети организационного управления и другие сети, отличающиеся по терминологии, но практически одинаковые по своей идеологической сути) и сети управления технологическими процессами на предприятиях.

Локальные сети различаются по роли и значению ПК в сети, структуре, методам доступа пользователей к сети, способам передачи данных между компонентами сети и др. Каждой из предлагаемых на рынке сетей присущи свои достоинства и недостатки. Выбор сети определяется числом подключаемых пользователей, их приоритетом, необходимой скоростью и дальностью передачи данных, требуемыми пропускной способностью, надежностью и стоимостью сети.

2 Аппаратное обеспечение сети. Клиент и сервер

Каждый компьютер, подключенный к локальной сети, должен иметь специальную плату (сетевой адаптер).

Основной функцией сетевого адаптера является передача и прием информации из сети. Соединение компьютеров (сетевых адаптеров) между собой производится с помощью кабелей различных типов (коаксиального, витой пары, оптоволоконного). Для подключения к локальной сети портативных компьютеров часто используется беспроводное подключение, при котором передача данных осуществляется с помощью электромагнитных волн.

Сервер или клиент - это функции, которые выполняет компьютер. Любой компьютер в сети может выполнять функции сервера или клиента, а может выполнять обе эти функции одновременно. Все зависит от программного обеспечения. Функции сервера (serve - обслуживать) - выполнять операции по запросам клиентов. Это может быть хранение и передача файлов, выполнение приложений с выдачей результатов, обслуживание принтеров и т.п. Если компьютер выполняет только функции сервера, то

его обычно называют выделенный сервер. Нередко у такого компьютера выключены или вовсе отсутствуют монитор или клавиатура, а все управление им производится с других компьютеров через сеть.

Сервер - это специально выделенная в сети ПЭВМ, в задачу которой входит управление всей сетью или частью сети (например, в комбинированных сетях), прием, хранение, обновление и выдача пользователям общей информации, управление высококачественными принтерами и графопостроителями.

3 Классификация сетей. Типы сетей. Сетевые топологии

Локальные сети, широко используемые в научных, управленческих, организационных и коммерческих технологиях, можно классифицировать по следующим признакам:

1. По роли ПЭВМ в сети:

- сети с сервером;
- одноранговые (равноправные) сети.

2. По структуре (топологии) сети:

- одноузловые («звезда»);
- кольцевые («кольцо»);
- магистральные («шина»);
- комбинированные.

3. По способу доступа пользователей к ресурсам и абонентам сети:

- сети с подключением пользователя по указанным адресам абонентов по принципу коммутации каналов («звезда»);
- сети с централизованным (программным) управлением подключения пользователей к сети («кольцо» и «шина»);
- сети со случайной дисциплиной обслуживания пользователей («шина»).

По виду коммуникационной среды передачи информации:

- сети с использованием существующих учрежденческих телефонных сетей;
- сети на специально проложенных кабельных линиях связи;
- комбинированные сети, совмещающие кабельные линии и радиоканалы.
- По дисциплине обслуживания пользователей (способу доступа пользователей к сети):

- приоритетные, задающиеся ЦУС, когда пользователи получают доступ к сети в соответствии с присвоенными им приоритетами (постоянными или изменяющимися);
- неприоритетные, когда все пользователи сети имеют равные права доступа к сети.

По размещению данных в компонентах сети:

- с центральным банком данных;
- с распределенным банком данных;
- с комбинированной системой размещения данных.

Не смотря на то, что все сети имеют определенное сходство они разделяются на два типа:

1) Одноранговые сети.

В одноранговой сети все компьютеры равноправны: нет иерархии среди компьютеров и нет выделенного сервера. Обычно каждый компьютер функционирует и как клиент и как сервер; иначе говоря, нет отдельного компьютера, ответственного за всю сеть. Пользователи сами решают, какие данные на своем компьютере сделать доступными по сети.

2) Сети, на основе выделенного сервера

Выделенным называется такой сервер, который функционирует только как сервер и не используется в качестве клиента или рабочей станции. Он оптимизирован для быстрой

обработки запросов от сетевых клиентов и для повышения защищенности файлов или каталогов. Сети на основе серверов стали промышленным стандартом.

Топология - физическая или электрическая конфигурация кабельного хозяйства и соединений сети. Топология - это скелет сети. Существует несколько основных типов: шина; звезда; кольцо; древовидная.

4 Глобальная сеть. Технические и программные ресурсы Интернета

Глобальная сеть — это система объединенных в сеть компьютеров, расположенных на больших расстояниях друг от друга. Отдельные участки Интернет представляют собой сети различной архитектуры, которые связываются между собой с помощью маршрутизаторов. Передаваемые данные разбиваются на небольшие порции, называемые пакетами. Каждый пакет перемещается по сети независимо от других пакетов. Сети в Интернет неограниченно коммутируются (т.е. связываются) друг с другом, потому что все компьютеры, участвующие в передаче данных, используют единый протокол коммуникации TCP/IP:

а) протокол TCP (Transmission Control Protocol) — протокол управления передачей данных, использующий автоматическую повторную передачу пакетов, содержащих ошибки; этот протокол отвечает за разбиение передаваемой информации на пакеты и правильное восстановление информации из пакетов получателя;

б) протокол IP (Internet Protocol) — протокол межсетевого взаимодействия, отвечающий за адресацию и позволяющий пакету на пути к конечному пункту назначения проходить по нескольким сетям.

Организация-владелец узла глобальной сети называется провайдером сетевых услуг. Английское слово provider обозначает «поставщик», «снабженец». Пользователь заключает договор с провайдером на подключение к его узлу и в дальнейшем оплачивает ему предоставляемые услуги (подобно тому, как мы оплачиваем услуги телефонной сети).

Узел содержит один или несколько мощных компьютеров, которые находятся в состоянии постоянного подключения к сети. Они называются хост-компьютерами (англ. host — хозяин).

Программные ресурсы Интернета. Работа Сети поддерживается определенным программным обеспечением (ПО). Это ПО функционирует на хост-компьютерах и на персональных компьютерах пользователей. Программное обеспечение узловых компьютеров очень разнообразно. Условно его можно разделить на базовое (системное) и прикладное. Базовое ПО обеспечивает поддержку работы сети по протоколу TCP/IP — базовому протоколу Интернета, то есть оно решает проблемы рассылки и приема информации. Прикладное ПО занимается обслуживанием разнообразных информационных услуг Сети, которые принято называть службами Интернета. Такие программы называются серверами. Для каждой службы существует своя сервер-программа: для электронной почты, для телеконференций, для WWW и пр.

1.17 Лекция №117 (2 часа).

Тема: «Основы защиты информации и сведений»

1.17.1 Вопросы лекции:

1. Лицензионные, условно бесплатные и бесплатные программы. Правовая охрана информации
2. Электронная подпись. Защита доступа к компьютеру.
3. Криптографические средства защиты информации

4. Компьютерные вирусы. Антивирусные программы

1.17.2 Краткое содержание вопросов:

1 Лицензионные, условно бесплатные и бесплатные программы. Правовая охрана информации

Программы по их юридическому статусу можно разделить на три большие группы: лицензионные, условно бесплатные (shareware) и свободно распространяемые программы (freeware). Дистрибутивы лицензионных программ (дискеты или диски CD-ROM, с которых производится установка программ на компьютеры пользователей) распространяются разработчиками на основании, договоров с пользователями на платной основе, т.е., лицензионные программы продаются. Некоторые фирмы — разработчики программного обеспечения предлагают пользователям условно бесплатные программы в целях их рекламы и продвижения на рынок. Пользователю предоставляется версия программы с ограниченным сроком действия (после истечения указанного срока программа перестает работать, если за нее не произведена оплата) или версия программы с ограниченными функциональными возможностями (в случае оплаты пользователю сообщается код, включающий все функции).

Многие производители программного обеспечения и компьютерного оборудования заинтересованы в широком бесплатном распространении программного обеспечения. К таким программным средствам можно отнести следующие:

- новые недоработанные (бета) версии программных продуктов (это позволяет провести их широкое тестирование);
- программные продукты, являющиеся частью принципиально новых технологий (это позволяет завоевать рынок);
- дополнения к ранее выпущенным программам, исправляющие найденные ошибки или расширяющие возможности;
- устаревшие версии программ;
- драйверы к новым устройствам или улучшенные драйверы к уже существующим.

Правовая охрана программ для ЭВМ и баз данных впервые в полном объеме введена в Российской Федерации Законом РФ «О правовой охране программ для электронных вычислительных машин и баз данных», который вступил в силу в 1992 году. Предоставляемая настоящим законом правовая охрана распространяется на все виды программ для ЭВМ (в том числе на операционные системы и программные комплексы), которые могут быть выражены на любом языке и в любой форме, включая исходный текст на языке программирования и машинный код. Однако правовая охрана не распространяется на идеи и принципы, лежащие в основе программы для ЭВМ, в том числе на идеи и принципы организации интерфейса и алгоритма. Для оповещения о своих правах разработчик программы может, начиная с первого выпуска в свет программы, использовать знак охраны авторского права, состоящий из трех элементов: буквы С в окружности или круглых скобках ©; наименования (имени) правообладателя; года первого выпуска программы в свет.

Автору программы принадлежит исключительное право осуществлять воспроизведение и распространение программы любыми способами, а также модификацию программы. Организация или пользователь, правомерно владеющий экземпляром программы (купивший лицензию на ее использование), вправе без получения дополнительного разрешения разработчика осуществлять любые действия, связанные с функционированием программы, в том числе ее запись и хранение в памяти ЭВМ. Запись и хранение в памяти ЭВМ допускаются в отношении одной ЭВМ или одного пользователя в сети, если другое не предусмотрено договором с разработчиком.

Необходимо знать и выполнять существующие законы, запрещающие нелегальное копирование и использование лицензионного программного обеспечения.

2 Электронная подпись. Защита доступа к компьютеру

В 2002 году был принят Закон РФ «Об электронно-цифровой подписи», который стал законодательной основой электронного документооборота в России. По этому закону электронная цифровая подпись в электронном документе признается юридически равнозначной подписи в документе на бумажном носителе.

При регистрации электронно-цифровой подписи в специализированных центрах корреспондент получает два ключа: секретный и открытый. Секретный ключ хранится на дискете или смарт-карте и должен быть известен только самому корреспонденту. Открытый ключ должен быть у всех потенциальных получателей документов и обычно рассылается по электронной почте. Процесс электронного подписания документа состоит в обработке с помощью секретного ключа текста сообщения. Далее зашифрованное сообщение посылается по электронной почте абоненту. Для проверки подлинности сообщения и электронной подписи абонент использует открытый ключ.

Для предотвращения, несанкционированного доступа к данным, хранящимся на компьютере, используются пароли. Компьютер разрешает доступ к своим ресурсам только тем пользователям, которые зарегистрированы и ввели правильный пароль. Каждому конкретному пользователю может быть разрешен доступ только к определенным информационным ресурсам. При этом может производиться регистрация всех попыток несанкционированного доступа.

В настоящее время для защиты от несанкционированного доступа к информации все более часто используются биометрические системы авторизации и идентификации пользователей. Используемые в этих системах характеристики являются неотъемлемыми качествами личности человека и поэтому не могут быть утерянными и подделанными.

Защита программ от нелегального копирования и использования. Компьютерные пираты, нелегально тиражируя программное обеспечение, обесценивают труд программистов, делают разработку программ экономически невыгодным бизнесом. Для того чтобы программное обеспечение компьютера могло функционировать, оно должно быть установлено (инсталлировано). Программное обеспечение распространяется фирмами-производителями в форме дистрибутивов на CD-ROM. Каждый дистрибутив имеет свой серийный номер, что препятствует незаконному копированию и установке программ.

Для предотвращения нелегального копирования программ и данных, хранящихся на CD-ROM, может использоваться специальная защита. На CD-ROM может быть размещен закодированный программный ключ, который теряется при копировании и без которого программа не может быть установлена.

Защита данных на дисках. Каждый диск, папка и файл локального компьютера, а также компьютера, подключенного к локальной сети, может быть защищен от несанкционированного доступа. Для них могут быть установлены определенные права доступа (полный, только чтение, по паролю), причем права могут быть различными для различных пользователей.

Защита информации в Интернете. Если компьютер подключен к Интернету, то в принципе любой пользователь, также подключенный к Интернету, может получить доступ к информационным ресурсам этого компьютера. Если сервер имеет соединение с Интернетом и одновременно служит сервером локальной сети (Инtranет-сервером), то возможно несанкционированное проникновение из Интернета в локальную сеть.

- Механизмы проникновения из Интернета на локальный компьютер и в локальную сеть могут быть разными:

- загружаемые в браузер Web-страницы могут содержать активные элементы ActiveX или Java-апплеты, способные выполнять деструктивные действия на локальном компьютере;

- некоторые Web-серверы размещают на локальном компьютере текстовые файлы cookie, используя которые можно получить конфиденциальную информацию о пользователе локального компьютера;
- с помощью специальных утилит можно получить доступ к дискам и файлам локального компьютера и др. Для того чтобы этого не происходило, устанавливается программный или аппаратный барьер между Интернетом и Интранетом с помощью брандмауэра (firewall — межсетевой экран). Брандмауэр отслеживает передачу данных между сетями, осуществляет контроль текущих соединений, выявляет подозрительные действия и тем самым предотвращает несанкционированный доступ из Интернета в локальную сеть.

3 Криптографические средства защиты информации

Криптографические методы защиты информации - это специальные методы шифрования, кодирования или иного преобразования информации, в результате которого ее содержание становится недоступным без предъявления ключа криптограммы и обратного преобразования. Криптографический метод защиты, безусловно, самый надежный метод защиты, так как охраняется непосредственно сама информация, а не доступ к ней (например, зашифрованный файл нельзя прочесть даже в случае кражи носителя). Данный метод защиты реализуется в виде программ или пакетов программ.

Современная криптография включает в себя четыре крупных раздела:

Симметричные криптосистемы. В симметричных криптосистемах и для шифрования, и для дешифрования используется один и тот же ключ. (Шифрование - преобразовательный процесс: исходный текст, который носит также название открытого текста, заменяется шифрованным текстом, дешифрование - обратный шифрованию процесс. На основе ключа шифрованный текст преобразуется в исходный);

Криптосистемы с открытым ключом. В системах с открытым ключом используются два ключа - открытый и закрытый, которые математически связаны друг с другом. Информация шифруется с помощью открытого ключа, который доступен всем желающим, а расшифровывается с помощью закрытого ключа, известного только получателю сообщения. (Ключ - информация, необходимая для беспрепятственного шифрования и дешифрования текстов.);

Электронная подпись. Системой электронной подписи называется присоединяемое к тексту его криптографическое преобразование, которое позволяет при получении текста другим пользователем проверить авторство и подлинность сообщения.

Управление ключами. Это процесс системы обработки информации, содержанием которых является составление и распределение ключей между пользователями.

Основные направления использования криптографических методов - передача конфиденциальной информации по каналам связи (например, электронная почта), установление подлинности передаваемых сообщений, хранение информации (документов, баз данных) на носителях в зашифрованном виде.

4 Компьютерные вирусы. Антивирусные программы

Компьютерные вирусы являются программами, которые могут «размножаться» и скрытно внедрять свои копии в файлы, загрузочные секторы дисков и документы. Активизация компьютерного вируса может вызывать уничтожение программ и данных. Разнообразны последствия действия вирусов; по величине вредных воздействий вирусы можно разделить на:

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

По «среде обитания» вирусы можно разделить на файловые, загрузочные, макровирусы и сетевые.

Файловые вирусы. Файловые вирусы различными способами внедряются в исполнимые файлы (программы) и обычно активизируются при их запуске.

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

Загрузочные вирусы. Загрузочные вирусы записывают себя в загрузочный сектор диска. При загрузке операционной системы с зараженного диска вирусы внедряются в оперативную память компьютера. В дальнейшем загрузочный вирус ведет себя так же, как файловый, то есть может заражать файлы при обращении к ним компьютера. Профилактическая защита от таких вирусов состоит в отказе от загрузки операционной системы с гибких дисков и установке в BIOS вашего компьютера защиты загрузочного сектора от изменений.

Макровирусы. Макровирусы заражают файлы документов Word и электронных таблиц Excel. Макровирусы являются фактически макрокомандами (макросами), которые встраиваются в документ.

Профилактическая защита от макровирусов состоит в предотвращении запуска вируса. При открытии документа в приложениях Word и Excel сообщается о присутствии в них макросов (потенциальных вирусов) и предлагается запретить их загрузку..

Сетевые вирусы. По компьютерной сети могут распространяться и заражать компьютеры любые обычные вирусы..

Интернет-черви (worm) — это вирусы, которые распространяются в компьютерной сети во вложенных в почтовое сообщение файлах. Профилактическая защита от интернет-червей состоит в том, что не рекомендуется открывать вложенные в почтовые сообщения файлы, полученные из сомнительных источников.

Особой разновидностью вирусов являются активные элементы (программы) на языках JavaScript или VBScript, которые могут выполнять разрушительные действия, то есть являться вирусами (скрипт-вирусами). Такие программы передаются по Всемирной паутине в процессе загрузки Web-страниц с серверов Интернета в браузер локального компьютера.

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

Антивирусные программы могут использовать различные принципы для поиска и лечения зараженных файлов.

Полифаги. Самыми популярными и эффективными антивирусными программами являются антивирусные программы полифаги (например, Kaspersky Anti-Virus, Dr.Web). Принцип работы полифагов основан на проверке файлов, загрузочных секторов дисков и оперативной памяти и поиске в них известных и новых (неизвестных полифагу) вирусов.

Ревизоры. Принцип работы ревизоров (например, ADInf) основан на подсчете контрольных сумм для присутствующих на диске файлов. Эти контрольные суммы затем сохраняются в базе данных антивируса, как и некоторая другая информация: длины файлов, даты их последней модификации и пр.

При последующем запуске ревизоры сверяют данные, содержащиеся в базе данных, с реально подсчитанными значениями. Если информация о файле, записанная в базе данных, не совпадает с реальными значениями, то ревизоры сигнализируют о том, что файл был изменен или заражен вирусом.

Блокировщики. Антивирусные блокировщики — это программы, перехватывающие «вирусоопасные» ситуации и сообщающие об этом пользователю.

2. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

2.1 Лабораторная работа №1 (2 часа).

Тема: «Правила техники безопасности. Введение в предмет. Входной контроль»

2.1.1 Цель работы: познакомить и заинтересовать студентов предметом;

показать, чему должны научиться студенты и какое значение имеет информатика в жизни; познакомиться с правилами техники безопасности; определить уровень знаний студентов.

2.1.2 Задачи работы:

1. познакомить с техникой безопасности и организацией рабочего места
2. выполнить работу используя ПО Microsoft Office

2.1.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.1.4 Описание (ход) работы:

Техника безопасности и организация рабочего места:

ЗАПОМНИТЕ! К каждому рабочему месту подведено опасное для жизни напряжение.

Во время работы следует быть предельно внимательным.

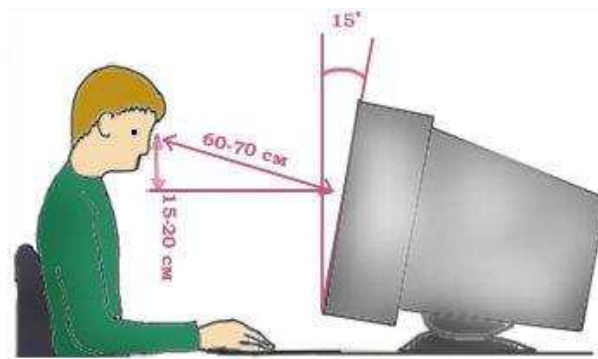
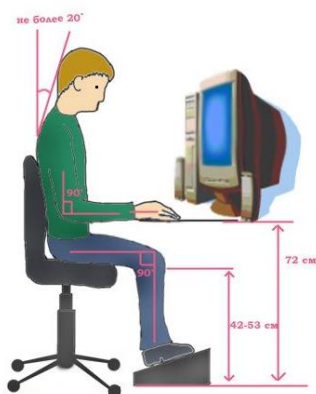
Во избежание несчастного случая, поражения электрическим током, поломки оборудования рекомендуется выполнять следующие правила:

- Входите в компьютерный класс спокойно, не торопясь, не толкаясь, не задевая мебель и оборудование и только с разрешения преподавателя.
- Не включайте и не выключайте компьютеры без разрешения преподавателя.
 - Не трогайте питающие провода и разъёмы соединительных кабелей.
 - Не прикасайтесь к экрану и тыльной стороне монитора.
 - Не размещайте на рабочем месте посторонние предметы.
 - Не вставайте со своих мест, когда в кабинет входят посетители.
- Не пытайтесь самостоятельно устранять неисправности в работе аппаратуры; при неполадках и сбоях в работе компьютера немедленно прекратите работу и сообщите об этом преподавателю.
- Работайте на клавиатуре чистыми, сухими руками; легко нажимайте на клавиши, не допуская резких ударов и не задерживая клавиши в нажатом положении.
 - Запрещается работать во влажной одежде и влажными руками.

При появлении запаха гари немедленно прекратите работу, выключите аппаратуру и сообщите об этом преподавателю.

ЗАПОМНИТЕ! Если не принимать мер предосторожности, работа за компьютером может оказаться вредной для здоровья.

Чтобы не навредить своему здоровью, необходимо соблюдать ряд простых рекомендаций:



- Неправильная посадка за компьютером может стать причиной боли в плечах и пояснице. Поэтому садитесь свободно, без напряжения, не сутулясь, не наклоняясь и не наваливаясь на спинку стула. Ноги ставьте прямо на пол, одна возле другой, не вытягивайте их и не подгибайте.
- Если стул с регулируемой высотой, то её следует отрегулировать так, чтобы угол между плечом и предплечьем был чуть больше прямого. Туловище должно находиться от стола на расстоянии 15-16 см. Линия зрения должна быть направлена в центр экрана. Если вы имеете очки для постоянного ношения, работайте в них.
- Плечи при работе должны быть расслаблены, локти – слегка касаться туловища. Предплечья должны находиться на той же высоте, что и клавиатура.
- При напряжённой длительной работе глаза переутомляются, поэтому каждые 5 минут отрывайте взгляд от экрана и смотрите на что-нибудь, находящееся вдали.

Входной контроль

Задание № 1

Создать презентацию, являющуюся визитной карточкой студента, содержащую информацию о ФИО, месте жительства, курсе, направлении подготовки и профиле (использовать команды панели инструментов, вставку графических объектов, различные приемы форматирования).

Задание № 2

В текстовом процессоре набрать предложенный текст.

Задание № 3

В табличном процессоре Excel создать таблицы, содержащую не менее 5 записей о наименованиях товара и его количестве и оптовой цене в у.е. (можно использовать свои цифры и наименования). Добавить к ячейкам столбца **Наименование** примечания, содержащие информацию о стране производителя каждого товара. В верхней части таблицы организовать ячейки, содержащие текущую дату и курс у.е. (выбрать по своему усмотрению).

Для каждого наименования товара рассчитать оптовую цену в рублях. Рассчитать розничную цену товара в рублях, используя формулу:

$$\text{Розничная цена} = \text{Оптовая цена} + N \cdot 20\% \text{ оптовой цены,}$$

где N – номер варианта.

С помощью встроенных статистических функций определить максимальную, среднюю и минимальную цену товара. Подсчитать итоговую сумму стоимости всего товара.

Изменить шрифты исходной таблицы, изменить цвет шрифта и цвет фона.

Создать два вида диаграмм на основе имеющихся данных. Предусмотреть названия диаграмм и легенду.

2.2 Лабораторная работа №2 (2 часа).

Тема: «Способы сбора информации»

2.2.1 Цель работы: сформировать у учащихся понятие «информационные процессы», «информационная культура», «информатизация общества»;

2.2.2 Задачи работы:

1. научиться обрабатывать информацию;
2. научиться хранить информацию;
3. научиться передавать информацию.;

2.2.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.2.4 Описание (ход) работы:

Подготовить доклад по теме:

1. Составные части информатики.
2. История развития информатики как дисциплины.
3. Место информатики в ряду других фундаментальных наук.
4. Социальные аспекты информатики.
5. Правовые аспекты информатики.
6. Этические аспекты информатики.
7. Роль информации в развитии общества.
8. Этапы развития информационного общества. Его информатизация.
9. Национальные информационные ресурсы России.
10. Информационная культура человека.
11. История развития компьютеров.
12. Основные структуры данных.

2.3 Лабораторная работа №3 (2 часа).

Тема: «Кодирование информации»

2.3.1 Цель работы: сформировать понимание процесса обмена информацией; показать различные виды кодирования информации; выявить преимущество двоичного кодирования различных видов информации.

2.3.2 Задачи работы:

1. знакомство с понятиями код, кодирование, двоичное кодирование, бит;
2. рассмотреть различные способы кодирования информации
3. научиться восстанавливать информацию по ее кодовому представлению.

2.3.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.3.4 Описание (ход) работы:

1. Используя базовую таблицу кодировки ASCII, закодировать слова: Alt Shift Enter

2. Перевести на русский язык закодированные слова:

102 117 110 99 116 105 111 117
112 114 111 99 101 100 117 114 101

3. Раскодировать слово:

11010011 11101111 11110000 11100000 11100010 11101011 11100101 11101101
11101000 11100101

Закодировать ее азбукой Морзе

4. Раскодировать пословицу

211 247 229 237 232 229 150 247 229 235 238 226 229 234 243 238 230 229 240 229 235 2
52 229

Закодировать ее азбукой Морзе и в двоичном коде.

5. Дана кодировочная таблица

	0	1	2	3	4	5	6	7	8	9
0	а	б	в	г	д	е	ё	ж	з	и
1	й	к	л	м	н	о	п	р	с	т
2	у	ф	х	ц	ч	ш	щ	ъ	ы	ь
3	э	ю	я							

1) Раскодировать текст: 15041400172011000902120004152509140501290619

Предусмотреть вариант для отделения слов друг от друга.

2) Закодировать фразу: я изучаю информатику

6. Используя азбуку Морзе, Windows-1251 (прописные и строчные) и двоичный код закодировать слова:

а) информация б) кодирование в) технология г) регистрация д)
изображение

7. Раскодировать фразу:

255 226 229 240 237 238 226 251 239 238 235 237 232 238 231
224 228 224 237 232 229 232
239 238 235 243 247 232 235 53

Любому молодцу скромность к лицу
Близко то время, ребята, когда в музей пойдет лопата
Без хорошего друга человек не знает своих ошибок
Длинный язык с умом не в родстве
Дружные сороки и верблюда одолевают
Настойчивый найдет, он и в камень гвоздь забьет
Зря похвалишь – человека испортишь

2.4 Лабораторная работа №4 (2 часа).

Тема: «Измерение информации»

2.4.1 Цель работы: подсчитывать объем занимаемой данными информации и
уметь переводить значения количества информации из одних единиц измерения в другие.

2.4.2 Задачи работы:

1. уметь находить количество информации

2. уметь переводить значения из одних единиц измерения информации в другие.

2.4.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.4.4 Описание (ход) работы:

Задачи на вероятностный подход

Задача 1. Конфеты находятся в одной из 10 коробок. Определить информационную неопределенность.

Задача 2. Шарик находится в одной из трех урн: А, В или С. Определить информационную неопределенность.

Задача 3. После реализации одного из возможных событий получили количество информации равное 9 бит. Какое количество возможных событий было первоначально?

Задача 4. Сколько вопросов следует задать и как их нужно сформулировать, чтобы узнать с какого из 16 путей отправляется ваш поезд?

Задача 5. Найти количество информации в однозначном сообщении.

Задача 6. Какое количество информации потребуется для кодирования одного шахматного поля?

Задача 7. Получено сообщение, объемом 10 бит. Какое количество сообщений возможно составить из полученных данных?

Задача 8. В корзине лежит 32 шаров разного цвета. Сколько информации несет сообщение, что достали белый шар?

Задача 9. Какое количество слов получится из фразы в 8 бит?

Задача 10. Сообщение о том, что ваш друг живет на 6 этаже несет 4 бита информации. Сколько этажей в доме?

Задача 11. При игре в кости используется кубик с шестью гранями. Сколько бит информации получает игрок при каждом бросании кубика?

Задача 12. В библиотеке 16 стеллажей. На каждом стеллаже по 8 полок Библиотекарь сказал Оле, что интересующая ее книга находится на 3 стеллаже, на 2-й сверху полке. Какое количество информации получила Оля?

Задача 13. Информационная емкость сообщения о том, что из корзины, где лежало некоторое количество разноцветных шаров, достали зеленый шар, несет в себе 0,375 байта информации. Сколько в корзине было шаров.

Задача 14. В мешке находятся 30 шаров, из них 10 белых и 20 черных. Какое количество информации несет сообщение о том, что достали белый шар, черный шар?

Задача 15. В классе 30 человек. За контрольную работу по математике получено 6 пятерок, 15 четверок, 8 троек и 1 двойка. Какое количество информации в сообщении о том, что Иванов получил тройку? Он получил двойку?

Задача 16. В корзине лежат черные и белые шары. Среди них 18 черных шаров. Сообщение о том, что достали белый шар, несет 2 бита информации. Сколько всего шаров в корзине?

Алфавитный подход к измерению информации

№1 Сообщение, записанное буквами из 64-х символьного алфавита, содержит 20 символов. Какой объем информации оно несет?

№2 Информационное сообщение объемом 1,5 Кбайта содержит 3072 символа. Сколько символов содержит алфавит, при помощи которого было записано это сообщение?

№3 Объем сообщения, содержащего 2048 символов, составил 1/512 часть Мбайта.

Каков размер алфавита, с помощью которого записано сообщение?

№4 Сколько символов содержит сообщение, записанное с помощью 16-ти символьного алфавита, если объем его составил 1/16 часть Мбайта?

№5 Сколько килобайтов составляет сообщение, содержащее 12288 битов?

№6 Сколько килобайтов составит сообщение из 384 символов 16-ти символьного алфавита?

№7 Для записи текста использовался 256-символьный алфавит. Каждая страница содержит 30 строк по 70 символов в строке. Какой объем информации содержат 5 страниц текста?

№8 Сообщение занимает 3 страницы по 25 строк. В каждой строке записано по 60 символов. Сколько символов в использованном алфавите, если все сообщение содержит 1125 байтов?

№9 Для записи сообщения использовался 64-х символьный алфавит. Каждая страница содержит 30 строк. Все сообщение содержит 8775 байтов информации и занимает 6 страниц. Сколько символов в строке?

№10 Сообщение занимает 2 страницы и содержит 1/16 Кбайта информации. На каждой странице записано 256 символов. Какова мощность использованного алфавита?

№11 Сообщение записано с помощью алфавита, содержащего 8 символов. Какое количество информации несет одна буква этого алфавита?

№12 Информационное сообщение объемом 4 Кбайта содержит 4096 символов. Сколько символов содержит алфавит, при помощи которого было записано это сообщение?

№13 Для записи сообщения использовался 64 – символьный алфавит. Каждая страница содержит 30 строк. Все сообщение содержит 8775 байтов информации и занимает 6 страниц. Сколько символов в строке?

№14 Пользователь вводит текст с клавиатуры со скоростью 90 знаков в минуту. Какое количество информации будет содержать текст, который он набирал 15 минут (используется компьютерный алфавит)?

№15 Пользователь вводил текст с клавиатуры 10 минут. Какова его скорость ввода информации, если информационный объем полученного текста равен 1 Кбайт.

№16 Ученик 9 класса читает со скоростью 250 символов в минуту. При записи текста использовался алфавит, содержащий 64 символа. Какой объем информации получит ученик, если будет непрерывно читать 20 минут?

2.5 Лабораторная работа №5(2 часа).

Тема: «Информационно-логические основы построения персонального компьютера»

2.5.1 Цель работы: научиться переводить числа в те системы счисления, которые использует ЭВМ, сформировать навыки выполнения арифметических действий с двоичными числами

2.5.2 Задачи работы:

1. знать основные приемы работы с позиционными системами счисления;
2. уметь переводить числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную;

3. производить обратный перевод из этих систем в десятичную;

2.5.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.5.4 Описание (ход) работы:

Общие теоретические сведения

Система счисления – это способ представления чисел цифровыми знаками и соответствующие ему правила действий над числами.

Системы счисления можно разделить: непозиционные системы счисления; позиционные системы счисления.

В непозиционной системе счисления значение (величина) символа (цифры) не зависит от положения в числе.

Самой распространенной непозиционной системой счисления является **римская**. Алфавит римской системы записи чисел состоит из символов: I – один, V – пять, X – десять, L – пятьдесят, C – сто, D – пятьсот, M – тысяча.

Величина числа определяется как сумма или разность цифр в числе (например, II – два, III – три, XXX – тридцать, CC – двести).

Если же большая цифра стоит перед меньшей цифрой, то они складываются (например, VII – семь), если наоборот – вычитаются (например, IX – девять).

В позиционных системах счисления значение (величина) цифры определяется ее положением в числе.

Любая позиционная система счисления характеризуется своим основанием.

Основание позиционной системы счисления – количество различных цифр, используемых для изображения чисел в данной системе счисления.

Основание 10 у привычной десятичной системы счисления (десять пальцев на руках).

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

За основание можно принять любое натуральное число – два, три, четыре и т. д., образовав новую позиционную систему: двоичную, троичную, четверичную и т. д.

Позиция цифры в числе называется **разрядом**.

Представим развернутую форму записи числа:

$A_q = a_{n-1} \cdot q^{n-1} + \dots + a_1 \cdot q^1 + a_0 \cdot q^0 + a_{-1} \cdot q^{-1} + \dots + a_{-m} \cdot q^{-m}$, где

q – основание системы счисления (количество используемых цифр)

A_q – число в системе счисления с основанием q

a – цифры многоразрядного числа A_q

n (m) – количество целых (дробных) разрядов числа A_q

Пример

порядковый номер

2 1 0 -1 -2

2 3 9, 4 $5_{10} = 2 \cdot 10^2 + 3 \cdot 10^1 + 9 \cdot 10^0 + 4 \cdot 10^{-1} + 5 \cdot 10^{-2}$

a_2 a_1 a_0 , a_{-1} a_{-2}

Двоичная система счисления

Официальное «рождение» двоичной системы счисления (в её алфавите два символа: 0 и 1) связывают с именем Готфрида Вильгельма Лейбница. В 1703 г. он опубликовал статью, в которой были рассмотрены все правила выполнения арифметических действий над двоичными числами.

Преимущества:

1. для её реализации нужны технические устройства с двумя устойчивыми состояниями:

есть ток — нет тока;
намагничен — не намагничен;

- представление информации посредством только двух состояний надежно и помехоустойчиво;
- возможно применение аппарата булевой алгебры для выполнения логических преобразований информации;
- двоичная арифметика намного проще десятичной.

Недостаток: быстрый рост числа разрядов, необходимых для записи чисел.

Перевод чисел (8) → (2), (16) → (2)

Перевод восьмеричных и шестнадцатеричных чисел в двоичную систему: каждую цифру заменить эквивалентной ей двоичной триадой (тройкой цифр) или тетрадой (четверкой цифр).

Примеры:

$$5371_8 = 101\ 011\ 111\ 001_2;$$

5 3 7 1

$$1A3F_{16} = 1\ 1010\ 0011\ 1111_2$$

1 A 3 F

Задание для самостоятельного выполнения

Переведите:

- $3754_8 \rightarrow X_2$
- $2ED_{16} \rightarrow X_2$

Перевод чисел (2) → (8), (2) → (16)

Чтобы перевести число из двоичной системы в восьмеричную или шестнадцатеричную, его нужно разбить влево и вправо от запятой на триады (для восьмеричной) или тетрады (для шестнадцатеричной) и каждую такую группу заменить соответствующей восьмеричной (шестнадцатеричной) цифрой.

Примеры:

$$1101010000111_2 = 1\ 5\ 2\ 0\ 7_8;$$

1 101 010 000 111

$$110111000001101_2 = 6\ E\ 0\ D_{16}$$

110 1110 0000 1101

Задание для самостоятельного выполнения

Переведите:

- $1011111010101100_2 \rightarrow X_8$
- $1011010100000110_2 \rightarrow X_{16}$

Перевод чисел (q) → (10)

Запись числа в развернутой форме и вычисление полученного выражения в десятичной системе.

Примеры:

$$1. \quad 110110_2 = 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 54_{10};$$

$$2. \quad 237_8 = 2 \cdot 8^2 + 3 \cdot 8^1 + 7 \cdot 8^0 = 128 + 24 + 7 = 159_{10};$$

$$3. \quad 3FA_{16} = 3 \cdot 16^2 + 15 \cdot 16^1 + 10 \cdot 16^0 = 768 + 240 + 10 = 1018_{10}.$$

Задание для самостоятельного выполнения

Переведите:

- $1100011010_2 \rightarrow X_{10}$
- $162_8 \rightarrow X_{10}$
- $E23_{16} \rightarrow X_{10}$

Перевод чисел (10) → (q)

Последовательное **целочисленное деление** десятичного числа на основание системы q , пока последнее частное не станет меньше делителя.
Затем остатки от деления записываются в порядке, обратном порядку их получения.

$$2009_{10} = 31014_5$$

$$75_{10} = 1001011_2$$

$$75_{10} = 113_8$$

$$75_{10} = B_{16}$$

Задание для самостоятельного выполнения

Переведите:

1. $141_{10} \rightarrow X_2$

2. $141_{10} \rightarrow X_8$

3. $141_{10} \rightarrow X_{16}$

Для перевода правильных дробей из десятичной системы счисления в произвольную используется метод последовательного умножения на основание системы счисления дробных цифр числа до тех пор, пока не получим в дробной части всех нулей или не достигнем заданной точности (если число не переводится точно).

Пример. Перевести из десятичной системы счисления в двоичную число 0,325.

$$0,375$$

$$\times 2$$

$$0,750$$

Выделяем целую часть: 0

$$0,750$$

$$\times 2$$

$$1,500$$

Выделяем целую часть: 1

$$0,500$$

$$\times 2$$

$$1,000$$

В дробной части получили все нули, т. е. число перевелось в двоичную систему счисления точно: $0,011_2$.

Двоичная арифметика

1. Таблица сложения

$$0 + 0 = 0$$

$$1 + 0 = 1$$

$$0 + 1 = 1$$

$$1 + 1 = 10$$

2. Таблица вычитания

$$0 - 0 = 0$$

$$1 - 0 = 1$$

$$1 - 1 = 0$$

$$10 - 1 = 1$$

3. Таблица умножения

$$0 \cdot 0 = 0$$

$$1 \cdot 0 = 0$$

$$1 \cdot 1 = 1$$

Пример. Сложить два числа в двоичной системе счисления.

$$\begin{array}{r} 11011 \\ + 101101 \\ \hline 1001000 \end{array}$$

Выполнение арифметических операций в различных системах счисления

1. Вычислить: а) $1011001+1111$; б) $11001,01+11001,11$; в) $10110011,1011+1001,01$
2. Выполнить вычитание: а) $100111101-101110$; б) $1100110,101-1011,1111$
3. Выполнить: а) $100111 \cdot 110$; б) $1100011,1 \cdot 101,1001$; в) 111^{11}
4. Вычислить: а) $1101101:111$; б) $101110,001:11,01$
5. В четверичной системе счисления записать таблицу сложения и умножения. Выполнить сложение и умножение следующих пар чисел:
а) 12 и 3; б) 120 и 11; в) 313,2 и 21; г) 32,01 и 0,213

2.6 Лабораторная работа №6 (2 часа).

Тема: «Логические основы построения персонального компьютера»

2.6.1 Цель работы: Изучить основы алгебры логики.

2.6.2 Задачи работы:

1. студент должен знать:

- о определения основных понятий (простое и сложное высказывания, логические операции, логические выражения, логическая функция);
- о порядок выполнения логических операций;
- о алгоритм построения таблиц истинности;
- о схемы базовых логических элементов;
- о законы логики и правила преобразования логических выражений;

2. студент должен уметь:

- о применять законы логики для упрощения логических выражений;
- о строить таблицы истинности;
- о строить логические схемы сложных выражений.

2.6.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.6.4 Описание (ход) работы:

Общие теоретические сведения

Основные понятия алгебры логики

Логической основой компьютера является алгебра логики, которая рассматривает логические операции над высказываниями.

Алгебра логики – это раздел математики, изучающий высказывания, рассматриваемые со стороны их логических значений (истинности или ложности) и логических операций над ними.

Логическое высказывание – это любое повествовательное предложение, в отношении которого можно однозначно сказать, истинно оно или ложно.

Пример. «3 – простое число» является высказыванием, поскольку оно истинно. Не всякое предложение является логическим высказыванием.

Пример. предложение «Давайте пойдем в кино» не является высказыванием. Вопросительные и побудительные предложения высказываниями не являются.

Высказывательная форма – это повествовательное предложение, которое прямо или косвенно содержит хотя бы одну переменную и становится высказыванием, когда все переменные замещаются своими значениями.

Пример. « $x+2>5$ » - высказывательная форма, которая при $x>3$ является истинной, иначе ложной.

Алгебра логики рассматривает любое высказывание только с одной точки зрения – является ли оно истинным или ложным. Слова и словосочетания «не», «и», «или», «если..., то», «тогда и только тогда» и другие позволяют из уже заданных высказываний строить новые высказывания. Такие слова и словосочетания называются **логическими связками**.

Высказывания, образованные из других высказываний с помощью логических связок, называются **составными** (сложными). Высказывания, которые не являются составными, называются **элементарными** (простыми).

Пример. высказывание «Число 6 делится на 2» - простое высказывание. Высказывание «Число 6 делится на 2, и число 6 делится на 3» - составное высказывание, образованное из двух простых с помощью логической связки «и».

Истинность или ложность составных высказываний зависит от истинности или ложности элементарных высказываний, из которых они состоят.

Чтобы обращаться к логическим высказываниям, им назначают имена.

Пример. Обозначим через А простое высказывание «число 6 делится на 2», а через В простое высказывание «число 6 делится на 3». Тогда составное высказывание «Число 6 делится на 2, и число 6 делится на 3» можно записать как «А и В». Здесь «и» – логическая связка, А, В – логические переменные, которые могут принимать только два значения – «истина» или «ложь», обозначаемые, соответственно, «1» и «0».

Каждая логическая связка рассматривается как операция над логическими высказываниями и имеет свое название и обозначение (табл. 1).

Таблица 1. Основные логические операции

Обозначение	Читается	Название операции	Альтернативные
-------------	----------	-------------------	----------------

операции			обозначения
\neg	НЕ	Отрицание (инверсия)	Черта сверху
\wedge	И	Конъюнкция (логическое умножение)	\cdot &
\vee	ИЛИ	Дизъюнкция (логическое сложение)	+
\rightarrow	Если ... то	Импликация	\supset
\leftrightarrow	Тогда и только тогда	Эквиваленция	\sim
XOR	Либо ...либо	Исключающее ИЛИ (сложение по модулю 2)	\oplus

НЕ Операция, выражаемая словом «не», называется **отрицанием** и обозначается чертой над высказыванием (или знаком \neg). Высказывание $\neg A$ истинно, когда A ложно, и ложно, когда A истинно.

Пример. Пусть A =«Сегодня пасмурно», тогда $\neg A$ =«Сегодня не пасмурно».

И Операция, выражаемая связкой «и», называется **конъюнкцией** (лат. conjunctio – соединение) или логическим умножением и обозначается точкой « \cdot » (может также обозначаться знаками \wedge или $\&$). Высказывание $A \cdot B$ истинно тогда и только тогда, когда оба высказывания A и B истинны.

Пример. Высказывание «Число 6 делится на 2, и число 6 делится на 3» - истинно, а высказывание «Число 6 делится на 2, и число 6 больше 10» - ложно.

ИЛИ Операция, выражаемая связкой «или» (в неисключающем смысле этого слова), называется **дизъюнкцией** (лат. disjunctio – разделение) или логическим сложением и обозначается знаком \vee (или плюсом). Высказывание $A \vee B$ ложно тогда и только тогда, когда оба высказывания A и B ложны.

Пример: Высказывание «Число 6 делится на 2 или число 6 больше 10» - истинно, а высказывание «Число 6 делится на 5 или число 6 больше 10» - ложно.

ЕСЛИ ... ТО Операция, выражаемая связками «если ..., то», «из ... следует», «... влечет ...», называется **импликацией** (лат. implico – тесно связаны) и обозначается знаком \rightarrow . Высказывание $A \rightarrow B$ ложно тогда и только тогда, когда A истинно, а B ложно.

Пример. Высказывание «если студент сдал все экзамены на «отлично», то он получит стипендию». Очевидно, эту импликацию следует признать ложной лишь в том случае, когда студент сдал на «отлично» все экзамены, но стипендии не получил. В остальных случаях, когда не все экзамены сданы на «отлично» и стипендия получена (например, в силу того, что студент проживает в малообеспеченной семье) либо когда экзамены вообще не сданы и о стипендии не может быть и речи, импликацию можно признать истинной.

РАВНОСИЛЬНО Операция, выражаемая связками «тогда и только тогда», «необходимо и достаточно», «... равносильно ...», называется **эквиваленцией** или **двойной**

импликацией и обозначается знаком \leftrightarrow или \sim . Высказывание $A \leftrightarrow B$ истинно тогда и только тогда, когда значения A и B совпадают.

Пример: Высказывание «Число является четным тогда и только тогда, когда оно делится без остатка на 2» является истинным, а высказывание «Число является нечетным тогда и только тогда, когда оно делится без остатка на 2» - ложно.

ЛИБО ... ЛИБО Операция, выражаемая связками «Либо ... либо», называется **исключающее ИЛИ** или **сложением по модулю 2** и обозначается XOR или \oplus .

Высказывание $A \oplus B$ истинно тогда и только тогда, когда значения A и B не совпадают.

Пример. Высказывание «Число 6 либо нечетно либо делится без остатка на 2» является истинным, а высказывание «Либо число 6 четно либо число 6 делится на 3» – ложно, так как истинны оба высказывания входящие в него.

Замечание. Импликацию можно выразить через дизъюнкцию и отрицание:

$$A \rightarrow B = \neg A \vee B$$

Эквиваленцию можно выразить через отрицание, дизъюнкцию и конъюнкцию:

$$A \leftrightarrow B = (\neg A \vee B) \wedge (\neg B \vee A)$$

Исключающее ИЛИ можно выразить через отрицание, дизъюнкцию и конъюнкцию:

$$A \text{ XOR } B = (\neg A \wedge B) \vee (\neg B \wedge A)$$

Вывод. Операций отрицания, дизъюнкции и конъюнкции достаточно, чтобы описывать и обрабатывать логические высказывания.

Порядок выполнения логических операций задается круглыми скобками. Но для уменьшения числа скобок договорились считать, что сначала выполняется операция отрицания («не»), затем конъюнкция («и»), после конъюнкции – дизъюнкция («или») и исключающего или и в последнюю очередь – импликация и эквиваленция.

С помощью логических переменных и символов логических операций любое высказывание можно формализовать, то есть заменить логической формулой (логическим выражением).

Логическая формула - это символическая запись высказывания, состоящая из логических величин (констант или переменных), объединенных логическими операциями (связками).

Логическая функция - это функция логических переменных, которая может принимать только два значения: 0 или 1. В свою очередь, сама логическая переменная (аргумент логической функции) тоже может принимать только два значения: 0 или 1.

Пример. $F(A, B) = A \& B \vee A$ – логическая функция двух переменных A и B.

Значения логической функции для разных сочетаний значений входных переменных – или, как это иначе называют, наборов входных переменных – обычно задаются специальной таблицей. Такая таблица называется **таблицей истинности**.

Приведем таблицу истинности основных логических операций (табл. 2)

Таблица 2

A	B	$\neg A$	$A \& B$	$A \vee B$	$A \rightarrow B$	$A \leftrightarrow B$	$A \text{ XOR } B$
1	1	0	1	1	1	1	0
1	0	0	0	1	0	0	1

0	1	1	0	1	1	0	1
0	0	1	0	0	1	1	0

Опираясь на данные таблицы истинности основных логических операций можно составлять таблицы истинности для более сложных формул.

Алгоритм построения таблиц истинности для сложных выражений:

1. Определить количество строк:

- количество строк = 2^n + строка для заголовка,
- n - количество простых высказываний.

2. Определить количество столбцов:

- количество столбцов = количество переменных + количество логических операций;
- определить количество переменных (простых выражений);
- определить количество логических операций и последовательность их выполнения.

Пример 1. Составить таблицу истинности для формулы И–НЕ, которую можно записать так: $\neg(A \& B)$.

1. Определить количество строк:

На входе два простых высказывания: А и В, поэтому n=2 и количество строк = $2^2+1=5$.

2. Определить количество столбцов:

Выражение состоит из двух простых выражений (А и В) и двух логических операций (1 инверсия, 1 конъюнкция), т.е. количество столбцов таблицы истинности = 4.

3. Заполнить столбцы с учетом таблиц истинности логических операций (табл. 3).

Таблица 3. Таблица истинности для логической операции

A	B	$A \& B$	$\neg(A \& B)$
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	1

Подобным образом можно составить таблицу истинности для формулы ИЛИ–НЕ, которую можно записать так:

$\neg(A \vee B)$.

Таблица 4. Таблица истинности для логической операции

A	B	$A \vee B$	$\neg(A \vee B)$
1	1	1	0
1	0	1	0
0	1	1	0

0	0	0	1
---	---	---	---

Примечание: И–НЕ называют также «штрих Шеффера» (обозначают $|$) или «антиконъюнкция»; ИЛИ–НЕ называют также «стрелка Пирса» (обозначают \downarrow) или «антидизъюнкция».

Пример 2. Составить таблицу истинности логического выражения $C = \neg A \& B \vee A \& \neg B$.

Решение:

1. Определить количество строк:

На входе два простых высказывания: А и В, поэтому $n=2$ и количество строк $=2^2+1= 5$.

2. Определить количество столбцов:

Выражение состоит из двух простых выражений (А и В) и пяти логических операций (2 инверсии, 2 конъюнкции, 1 дизъюнкция), т.е. количество столбцов таблицы истинности = 7.

Сначала выполняются операции инверсии, затем конъюнкции, в последнюю очередь операция дизъюнкции.

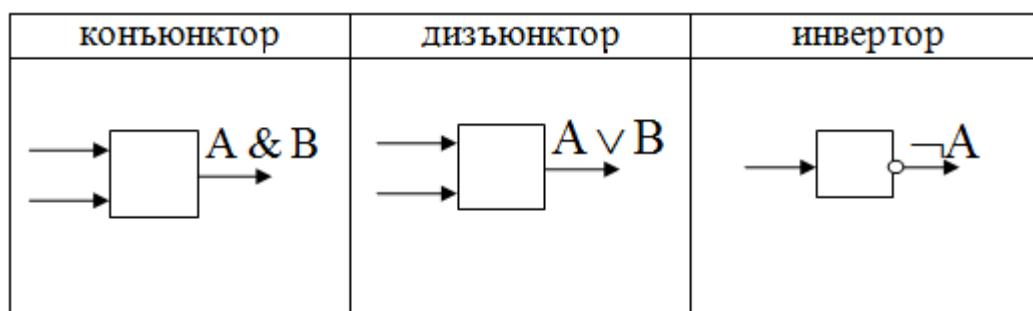
3. Заполнить столбцы с учетом таблиц истинности логических операций (табл. 5).

Таблица 5. Таблица истинности для логической операции $C = \neg A \& B \vee A \& \neg B$

A	B	$\neg A$	$\neg B$	$\neg A \& B$	$A \& \neg B$	C
1	1	0	0	0	0	0
1	0	0	1	0	1	1
0	1	1	0	1	0	1
0	0	1	1	0	0	0

Логические формулы можно также представлять с помощью языка логических схем. Существует три базовых логических элемента, которые реализуют три основные логические операции:

- логический элемент «И» – логическое умножение – конъюнктор;
- логический элемент «ИЛИ» – логическое сложение – дизъюнктор;
- логический элемент «НЕ» – инверсию – инвертор.



Поскольку любая логическая операция может быть представлена в виде комбинации трех

основных, любые устройства компьютера, производящие обработку или хранение информации, могут быть собраны из базовых логических элементов, как из “кирпичиков”.

Логические элементы компьютера оперируют с сигналами, представляющими собой электрические импульсы. Есть импульс – логический смысл сигнала – 1, нет импульса – 0. На входы логического элемента поступают сигналы-значения аргументов, на выходе появляется сигнал-значение функции.

Преобразование сигнала логическим элементом задается таблицей состояний, которая фактически является таблицей истинности, соответствующей логической функции, только представлена в форме логических схем. В такой форме удобно изображать цепочки логических операций и производить их вычисления.

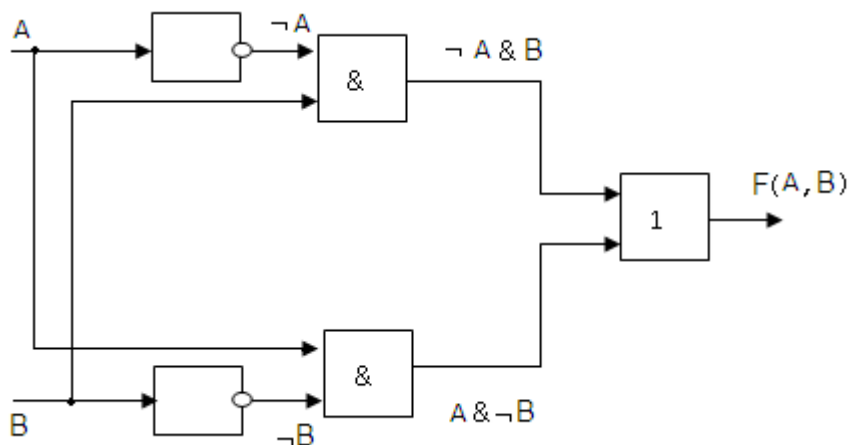
Алгоритм построения логических схем.

1. Определить число логических переменных.
2. Определить количество логических операций и их порядок.
3. Изобразить для каждой логической операции соответствующий ей логический элемент.
4. Соединить логические элементы в порядке выполнения логических операций.

Пример. По заданной логической функции $F(A,B) = \neg A \& B \vee A \& \neg B$ построить логическую схему.

Решение.

1. Число логических переменных = 2 (A и B).
2. Количество операций = 5 (2 инверсии, 2 конъюнкции, 1 дизъюнкция). Сначала выполняются операции инверсии, затем конъюнкции, в последнюю очередь операция дизъюнкции.
3. Схема будет содержать 2 инвертора, 2 конъюнктора и 1 дизъюнктор.
4. Построение надо начинать с логической операции, которая должна выполняться последней. В данном случае такой операцией является логическое сложение, следовательно, на выходе должен быть дизъюнктор. На него сигналы подаются с двух конъюнкторов, на которые, в свою очередь, подаются один входной сигнал нормальный и один инвертированный (с инверторов).



Логические законы и правила преобразования логических выражений

Если две формулы A и B одновременно, то есть при одинаковых наборах значений входящих в них переменных, принимают одинаковые значения, то они называются **равносильными**.

В алгебре логики имеется ряд законов, позволяющих производить равносильные преобразования логических выражений.

1. Закон двойного отрицания: $A = \neg(\neg A)$;

2. Переместительный (коммутативный) закон:

- для логического сложения: $A \vee B = B \vee A$;
- для логического умножения: $A \wedge B = B \wedge A$;

3. Сочетательный (ассоциативный) закон:

- для логического сложения: $(A \vee B) \vee C = A \vee (B \vee C)$;
- для логического умножения: $(A \wedge B) \wedge C = A \wedge (B \wedge C)$;

4. Распределительный (дистрибутивный) закон:

- для логического сложения: $(A \vee B) \wedge C = (A \wedge C) \vee (B \wedge C)$;
- для логического умножения: $(A \wedge B) \vee C = (A \vee C) \wedge (B \vee C)$;

5. Законы де Моргана:

- для логического сложения: $\neg(A \vee B) = \neg A \wedge \neg B$;
- для логического умножения: $\neg(A \wedge B) = \neg A \vee \neg B$;

6. Закон идемпотентности:

- для логического сложения: $A \vee A = A$;
- для логического умножения: $A \wedge A = A$;

7. Законы исключения констант:

- для логического сложения: $A \vee 1 = 1$, $A \vee 0 = A$;
- для логического умножения: $A \wedge 1 = A$, $A \wedge 0 = 0$;

8. Закон противоречия: $A \wedge \neg A = 0$;

9. Закон исключения третьего: $A \vee \neg A = 1$;

10. Закон поглощения:

- для логического сложения: $A \vee (A \wedge B) = A$;
- для логического умножения: $A \wedge (A \vee B) = A$;

11. Правило исключения импликации: $A \rightarrow B = \neg A \vee B$;

12. Правило исключения эквиваленции: $A \leftrightarrow B = (A \rightarrow B) \wedge (B \rightarrow A)$.

Справедливость этих законов можно доказать составив таблицу истинности выражений в правой и левой части и сравнив соответствующие значения.

Основываясь на законах, можно выполнять упрощение сложных логических выражений. Такой процесс замены сложной логической функции более простой, но равносильной ей, называется минимизацией функции.

Пример. Упростить логическое выражение $\neg(A \vee B) \wedge (A \& \neg B)$.

Решение:

Согласно закону де Моргана:

$$\neg(A \vee B) \wedge (A \& \neg B) \vee A = \neg A \& \neg B \& (A \& \neg B) \vee A$$

Согласно сочетательному закону:

$$\neg A \& \neg B \& (A \& \neg B) \vee A = \neg A \& A \& \neg B \& \neg B \vee A$$

Согласно закону противоречия и закону идемпотентности:

$$\neg A \& A \& \neg B \& \neg B \vee A = 0 \wedge \neg B \& \neg B = 0 \& \neg B \vee A$$

Согласно закону исключения 0:

$$0 \& \neg B = 0$$

Окончательно получаем

$$\neg(A \vee B) \wedge (A \& \neg B) \vee A = 0 \vee A = A$$

Задания

Содержание отчета

1. Текст задания (с данными своего варианта).
2. Представление по каждому пункту задания подробного решения.

Технология выполнения работы

В данной работе необходимо составить таблицу истинности логического выражения, построить схему логической функции и упростить логическое выражение заданные каждому студенту в соответствии с его вариантом, записать ход рассуждений и полученные результаты.

Вопросы для защиты работы

1. Что такое высказывание (приведите пример)?
2. Что такое составное высказывание (приведите пример)?
3. Как называются и как обозначаются (в языке математики) следующие операции: ИЛИ, НЕ, И, ЕСЛИ ... ТО, ТОГДА И ТОЛЬКО ТОГДА, ЛИБО ... ЛИБО?
4. Укажите приоритеты выполнения логических операций.
5. Составьте таблицу истинности для следующих операций: отрицание, конъюнкция, дизъюнкция, импликация, эквиваленция.
6. Изобразите функциональные элементы: конъюнктор, дизъюнктор, инвертор.
7. Какие логические выражения называются равносильными?
8. Записать основные законы алгебры логики.

2.7 Лабораторная работа №7 (2 часа).

Тема: «Первоначальные сведения и правила работы в операционные системы Windows»

2.7.1 Цель работы: Освоить основные приемы и методы работы с объектами операционной системы Windows

2.7.2 Задачи работы:

1. Освоение основных приемов настройки и работы с объектами, используя Главное меню операционной системы Windows

2. Освоение методов работы с объектами, используя служебную программу «Проводник» операционной системы Windows

2.7.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.7.4 Описание (ход) работы:

Задание 1

- 1 Разместите значки, находящиеся на Рабочем столе, в произвольном порядке.
- 2 При помощи контекстного меню Рабочего стола упорядочьте их размещение. Для упорядочения размещения значков на Рабочем столе вызовите контекстное меню. Для вызова контекстного меню щелкните на свободном месте Рабочего стола правой кнопкой мыши.
- 3 Запустите на выполнение несколько стандартных программ на выбор используя Главное меню: ПУСК/ ПРОГРАММЫ/СТАНДАРТНЫЕ, например, Блокнот, WordPad, Paint, Калькулятор (см. рисунок 1.1).

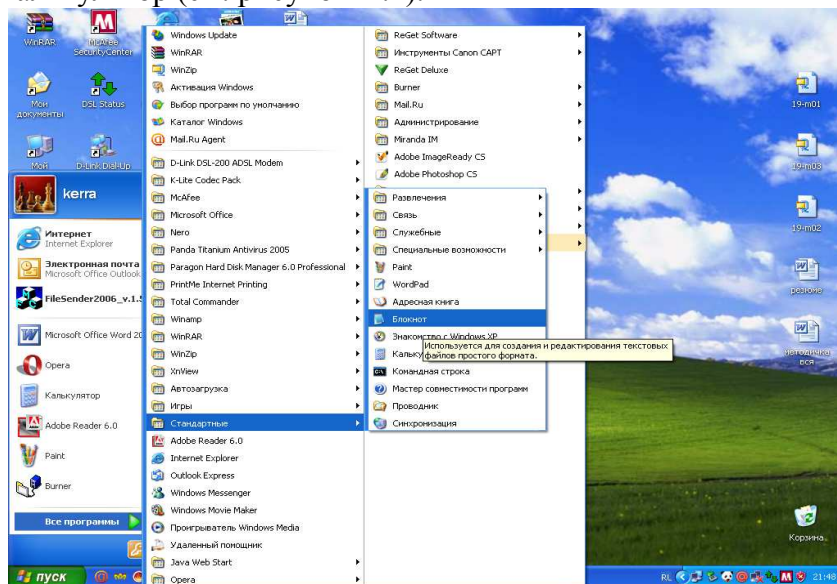



Рисунок 1.1 – Вид главного меню

4 Отобразите окна всех программ в полноэкранном режиме. Для этого необходимо нажать кнопку <Развернуть> 

5 Упорядочите окна всех программ каскадом, слева направо, сверху вниз. Для того чтобы упорядочить размещение открытых окон на Рабочем столе, воспользуйтесь контекстным меню **Панели задач** (т.е. щелкните на свободном месте Панели задач правой кнопкой мыши), опции которого позволяют на Рабочем столе разместить окна, а также

определить их свойства: каскадом, слева направо, сверху вниз, свернуть все (показать Рабочий стол).

6 Сверните окно одной из программ, а окно второй восстановите до первоначального размера. Измените с помощью мыши размеры окна. Для переключения программы в различные режимы воспользуйтесь кнопками в правом верхнем углу окна программы. Для регулирования окна вручную подведите к краю окна указатель мыши, который превратится в маркер растяжки. Затем, удерживая нажатой левую кнопку мыши, перетащите маркер в необходимое положение.

7 Закройте открытые программы, кроме документа программы Блокнот. Закройте программу можно одним из способов: щелкнув по кнопке <Заккрыть>, выполнив команду Выход меню ФАЙЛ, нажав комбинацию клавиш <ALT> и <F4>.

8 Найдите в справочной системе Windows три термина: «мультимедиа», «настройка Главного меню», «прокрутка». Скопируйте по очереди их описание и поместите сводный текст в документ БЛОКНОТ. Для этого:

- запустите программу «Справка», используя Главное меню;
- выберите закладку УКАЗАТЕЛЬ;
- в пустом поле введите ключевое слово и щелкните по кнопке **Вывести** или

Показать;

- выделите содержание справки и скопируйте в буфер обмена при помощи опции контекстного меню Копировать;
- активируйте окно программы Блокнот вставьте скопированный материал в Блокнот;
- сохраните выбранный материал при помощи команды Файл/Сохранить, выберите свою папку и наберите имя файла: **справка.txt**, закрыть все окна.

9 Найти путь нахождения файла **справка.txt**. Для поиска объекта вызовите команду Пуск/Поиск(Найти)/Файлы и папки: в окне **Найти** в закладках **имя и размещение, дата изменения, дополнительно** сделайте соответствующие установки. Сверните окно поиска.

10 Создайте ярлык для файла **справка.txt** и поместите его на Рабочий стол. Запустите программу при помощи созданного ярлыка.

11 Поместите файл справка.txt в Главное меню ПРОГРАММЫ. Запустите файл из Главного меню. Для размещения файла (или программы) в Главном меню Программы необходимо:

- выбрать в Главном меню пункт Настройка;
- выбрать пункт Панель задач и меню «Пуск» и откройте вкладку Настройка меню;
- щелкнуть по кнопке Добавить, а затем - по кнопке Обзор;
- найти нужный файл и дважды щелкните по нему;
- щелкнуть по кнопке Далее и затем дважды щелкните по нужной папке (программы) в структуре меню;
- ввести в поле название новой команды, добавляемой в меню, и щелкните по кнопке Готово.

12 Удалите программу-приложение из меню ПРОГРАММЫ. Для удаления программы из Главного меню или меню Программы:

- выберите в главном меню пункт Настройка;
- выберите пункт Панель задач и откройте вкладку Настройка меню;
- щелкните по кнопке Удалить, а затем найдите в структуре меню удаляемую программу;
- для того чтобы открыть меню, щелкните стоящий рядом с ним знак «+».

Выделите удаляемый пункт меню и щелкните по кнопке Удалить.

Задание 2. Работа в служебной программе «Проводник»

1 Запустите приложение программы «Проводник» одним из способов. Запуск программы Проводник можно выполнить с помощью:

- кнопки Пуск, используя контекстное меню;
- папки Мой компьютер, используя контекстное меню.

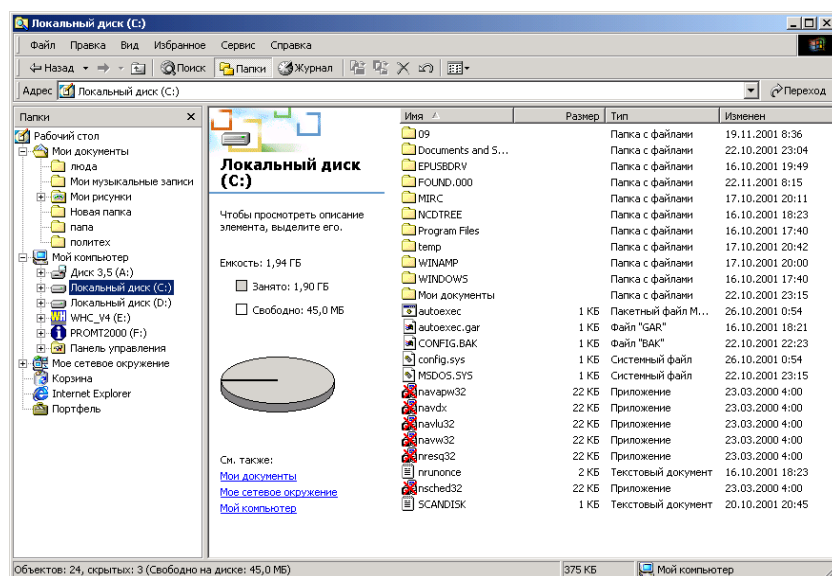


Рисунок 1.2 – Вид программы Проводник

2 Выполните с окном программы Проводник (см. рисунок 1.2) стандартные операции: уменьшение размеров окна, перемещение окна в пределах Рабочего стола, сворачивание окна программы Проводник, разворачивание окна программы Проводник.

3 Измените соотношения внутренних окон программы. Для этого необходимо подвести указатель мыши к границе между окнами и левой кнопкой мыши перетащить на нужное расстояние влево или вправо.

4 Изучите состав меню окна Проводник. Ознакомьтесь с командами меню ФАЙЛ и переведите указатель мыши на очередное меню.

5 Ознакомьтесь с названием кнопок Панели инструментов, указывая поочередно указателем мыши на каждую из них. Названия кнопок поочередно появляются под кнопками Панели инструментов.

6 Щелкните по любой папки, щелкните по любой папки оцените результат.

7 Отобразите содержимое любой папки, щелкнув по ней. Упорядочите объекты данной папки по имени, типу, размеру или дате.

8 Выберите один из видов отображения папок: плитка (крупные значки), эскизы страниц, список, таблица. Обратите внимание чем отличаются друг от друга отображения.

9 Для получения информации об объекте используйте: команду **Свойства** меню **Файл** или контекстное меню. Получите информацию про файл: справка.txt.

10 Ознакомьтесь с содержанием рабочего логического диска C:, просмотрев все ветви на соответствующей панели программы Проводник, и получите информацию о свойствах логического диска.

11 Откройте личную папку и создайте одним из способов папку **Персональная**.

12 Скопируйте документ **справка.txt** в папку Персональная на Рабочем столе, используя при этом метод перетаскивания.

13 Создайте две копии папки Персональная и поместите одну из них на Рабочий стол, используя при этом метод перетаскивания.

14 Переименуйте копию папки Персональная в личной папке на **Персональная_1** на Рабочем столе в **Персональная_2**.

15 Скопируйте папку Персональная_2 в личную папку, используя при этом метод перетаскивания.

- 16 Откройте содержимое папки Персональная_2. Создайте в ней несколько разного типа файлов: точечный рисунок, текстовый документ, документ MS Word.
- 17 Удалите из папки Персональная_1 файл справка.txt.
- 18 Просмотрите содержимое Корзины. Восстановите только что удаленный файл.
- 19 Закройте все окна.

2.8 Лабораторная работа №8 (2 часа).

Тема: «Работа с сервисными программами в операционной системе Windows»

2.8.1 Цель работы: Получить практические навыки работы с дискетами и сервисными программами Windows

2.8.2 Задачи работы:

1. Освоение основных приемов работы с программами, обслуживающими диски в операционные системы Windows

2. Создание архивного файла и работа в программе-архиваторе WinRar

2.8.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.8.4 Описание (ход) работы:

Задание 1. Отформатируйте 2 дискеты, задав им метки *диск 1* и *диск 2*

1 Выделите значок **Диск 3,5 (A:)** в папке **Мой компьютер**. Если содержимое диска отображается в папке **Мой компьютер**, отформатировать диск будет невозможно. Диск может быть отформатирован только в том случае, если на нем нет открытых файлов.

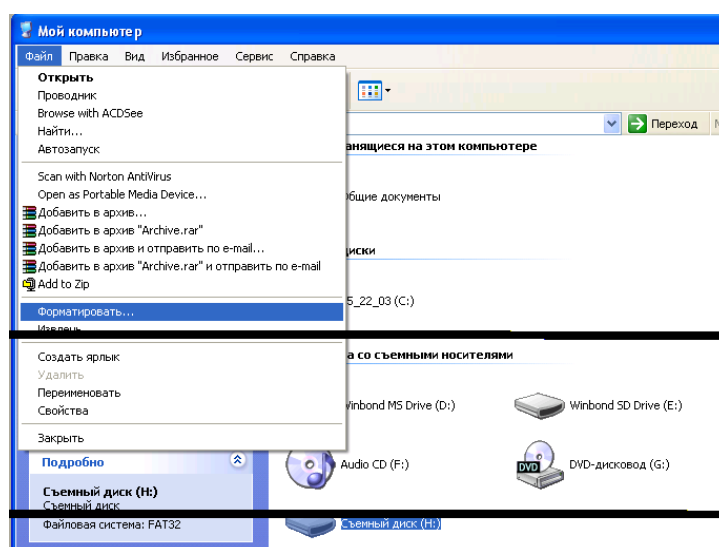


Рисунок 2.1 - Вид меню при выборе пункта Форматировать

2 Выберите команду **Форматировать** в меню **Файл** (см. рисунок 2.1) или в контекстном меню.

3 В открывшемся диалоговом окне Форматирование установите вид форматирования **Полное** и запишите метку дискеты **диск 1** в соответствующем окне. Поставьте флажок в опции **Вывести отчет в результатах**.

Помните!!! Форматирование приводит к полному уничтожению всех данных на диске.

4 Повторите аналогичные действия со второй дискетой, задав метку дискеты **диск 2**.

5 Скопируйте на первую дискету несколько различных объектов (папок, файлов, ярлыков, приложений). Для копирования данных на дискету можно воспользоваться любым способом: командами меню, кнопками Панели инструментов, методом *drag and drop*. Наиболее быстрым и удобным способом является команда контекстного меню *Отправить, Диск 3,5(A)*.

Задание 2. Проверьте поверхность дискеты диск 1 и информацию, содержащуюся на ней, программой Проверка диска. Получите итоговую информацию о сканировании

1 Необходимо запустить программу проверки диска. Для этого необходимо в папке **Мой компьютер** выделить диск **A:**.

2 В контекстном меню выбрать **Свойства**. В диалоговом окне **Свойства диска** (рисунок 2.2) выбрать вкладку **Сервис** и пункт **Выполнить проверку**.

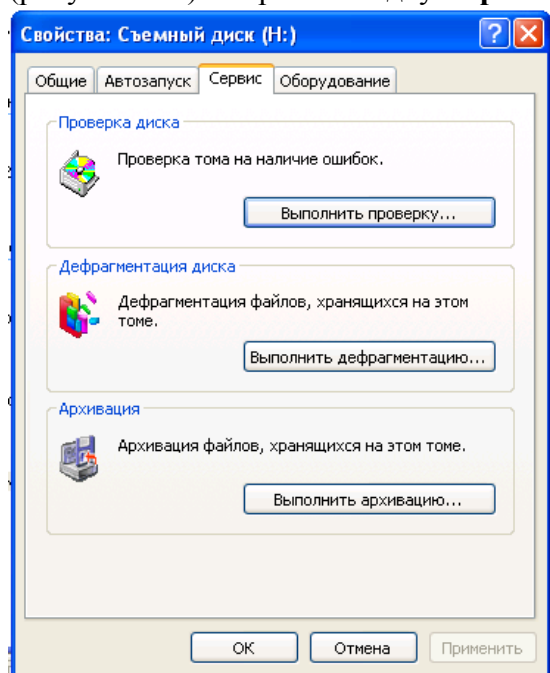


Рисунок 2.2 - Вид окна «Свойства диска».

3 Установите параметры проверки в соответствии с рисунком 2.3 и нажмите кнопку **Запуск**.

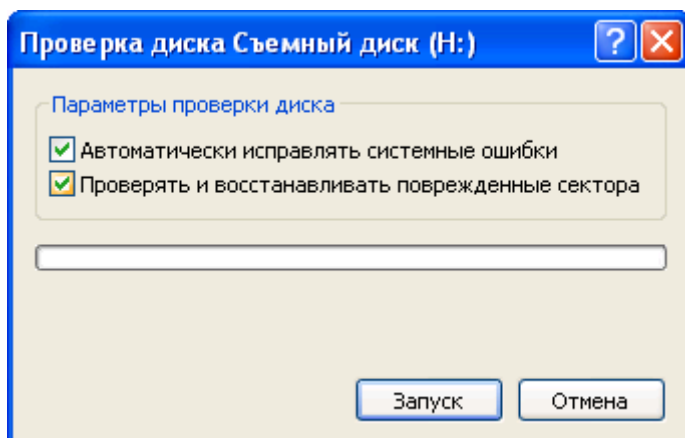


Рисунок 2.3 - Параметры проверки диска.

Задание 3. Выполните дефрагментацию дискеты *диск 1*

1 Необходимо запустить программу дефрагментации диска. Для этого необходимо в папке **Мой компьютер** выделить **диск A:**.

2 В контекстном меню выбрать **Свойства**. В диалоговом окне **Свойства диска** (рисунок 2.2) выбрать вкладку **Сервис** и пункт **Выполнить дефрагментацию**.

3 Установите параметры фрагментации и нажмите **Старт**.

Задание 4. Сделайте копию дискеты *диск 1* на дискете *диск 2*

Создание копии диска позволяет перенести все объекты с *диска 1* на *диск 2*, не используя промежуточные операции, что экономит время и ресурсы.

Для создания копии необходимо выполнить следующие действия:

1 Вставьте в дисковод дискету *диск 1*.

2 Выделите значок Диск 3,5 (A:) в папке Мой компьютер или на правой панели программы Проводник. Выберите в меню **Файл** команду **Копировать диск**.

3 В открывшемся окне **Копирование диска** (рисунок 2.4) щелкните по кнопке **Начать**.

4 После чтения содержимого *диск 1* и копирования его в буфер последует предложение о вставке диска 2 (рисунок 2.5). Вставьте *диск 2* и щелкните по кнопке **ОК**.

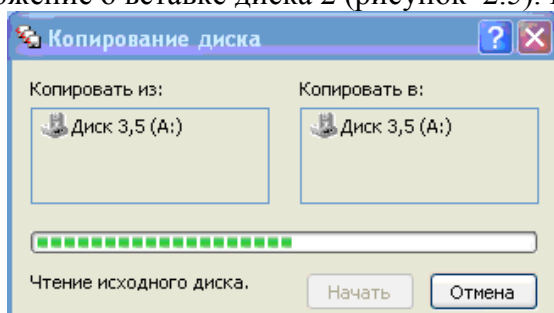


Рисунок 2.4 - Окно «Копирование диска».

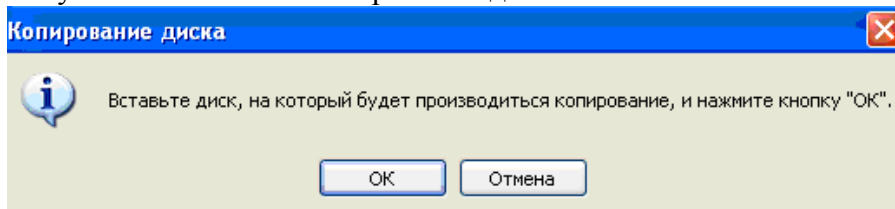


Рисунок 2.5 - Сообщение после копирования содержимого первого диска.

Задание 5. Получите информацию о дискете *диск 2*

Для того чтобы получить информацию о диске необходимо:

— выделите необходимый диск и выберите команду **Свойства** в меню **Файл** или контекстного меню;

— просмотреть все необходимые вкладки, согласно рисунка 2.6.

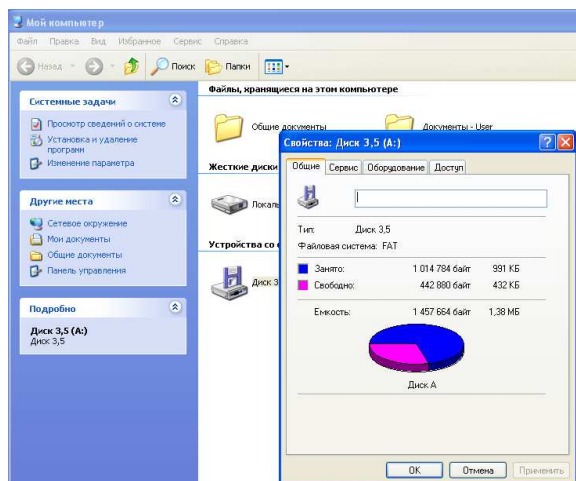


Рисунок 2.6 – Вид окна Свойства Диск 3,5 (A)

Задание 6. Архивирование файлов

1 Подготовьте объекты архивирования. Для этого необходимо создать в личной папке папку **Архив_1**. В эту папку поместите несколько различных объектов, например, 5 файлов формата документ MS Word, имеющих расширение .doc, и 5 графических файлов с расширением .bmp, .jpg.

2 Запустите программу **WinRAR**. В его рабочем окне (см. рисунок 2.7) найдите и откройте папку **Архив_1**.

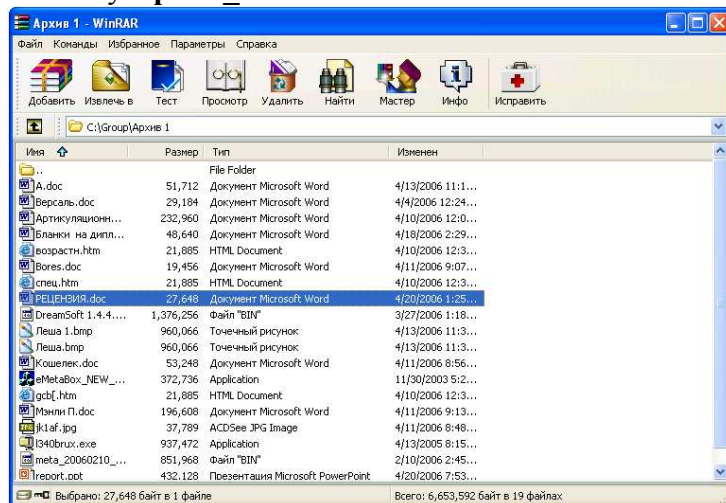


Рисунок 2.7 – Вид рабочего окна программы WinRAR

3 Выделите файлы формата **Microsoft Word**. Для этого можно воспользоваться следующим способом: выбрать команду меню **Файл Выделить группу** (рисунок 2.8).

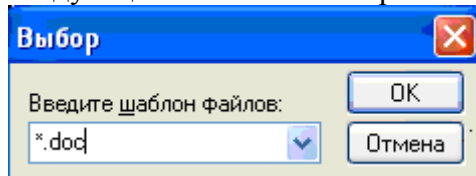


Рисунок 2.8 - Окно выделения группы файлов

4 Заархивировать данные файлы. Для этого можно воспользоваться командой меню **Команды, Добавить файлы в архив** (рисунок 2.9) или кнопкой на Панели инструментов **Добавить** или командой контекстного меню **Добавить файлы в архив**.

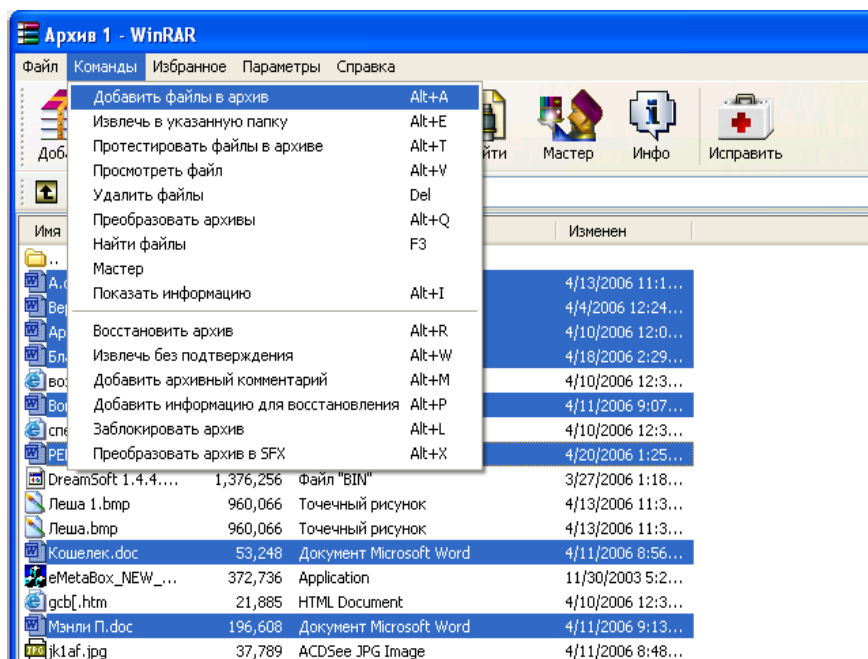


Рисунок 2.9 - Добавление файлов в архив

5 Задать необходимые параметры: **Имя архива** – **doc.rar**, **Метод сжатия** - «**Обычный**», **Размер словаря** – по умолчанию 256 Кбайт. Щелкните по кнопке **ОК**. В папке **Архив_1** появился архивный файл **doc.rar**

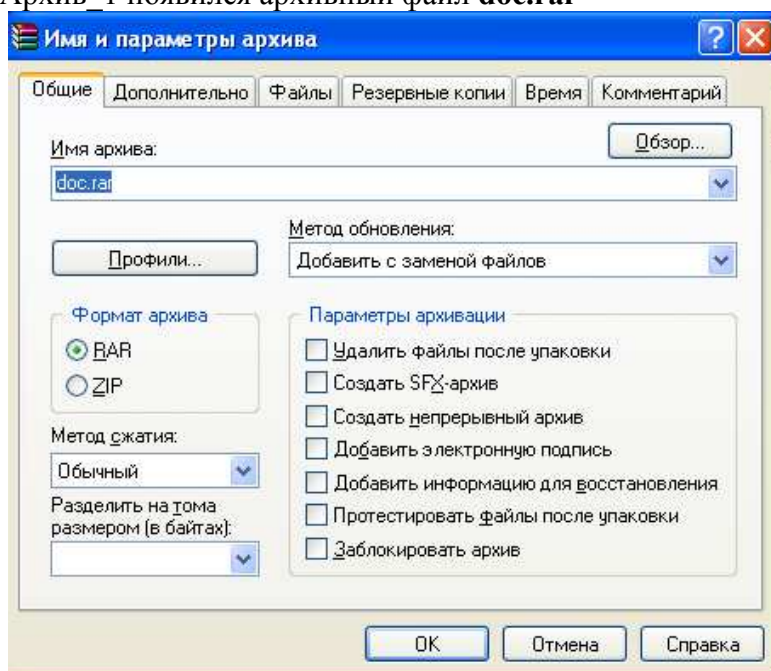


Рисунок 2.10 – Вид диалогового окна Имя и параметры архива

6 Сравните объем архивного файла **doc.rar** и объем всех файлов в папке **Архив_1** с расширением .doc Оцените результат.

7 Создайте в личной папке папку **Архив_2**, а в нем архивный файл **Картинки.rar**, в который заархивируйте все графические файлы из папки **Архив_2**. Для этого:

- выделите все графические файлы в папке **Архив_1**;
- выберите в контекстном меню команду **Добавить файлы в архив**;
- в окне **Имя и параметры архива** выберите кнопку **Обзор**, в открывшемся окне выберите папку **Архив_2** и задайте имя **Картинки**;
- задайте необходимые параметры и нажмите кнопку **ОК**.

8 Создайте в личной папке любой файл, назвав его «Дополнение». Добавьте данный файл в уже существующий архивный файл **doc.rar** Для этого:

- выделите его и выполните команду **Добавить**;
- в окне **Имя и параметры архива** выберите кнопку **Обзор**, в открывшемся окне выберите архивный файл **doc.rar**;
- задайте необходимые параметры и нажмите кнопку **ОК**.

9 Для того чтобы просмотреть информацию о файлах, помещенных в архив, откройте его в окне программы **WinRAR** двойным щелчком мыши или нажатием клавиши **Enter**, выделите необходимый файл и воспользуйтесь командой **Информация об архиве**;

10 В личной папке создайте папку **Извлеченные файлы**. Извлеките в нее из архива **doc.rar** и **Картинки.rar** несколько файлов. Для этого:

- откройте архивный файл;
- выделите нужные файлы и нажмите кнопку **Извлечь в**;
- в открывшемся окне **Путь и параметры извлечения** в правой части выбрать папку **Извлеченные файлы**, проверить в верхней строке путь к папке (см. рисунок 2.11), установить другие параметры, например, поставить галочку **Показать файлы в Проводнике**, нажать **ОК**.

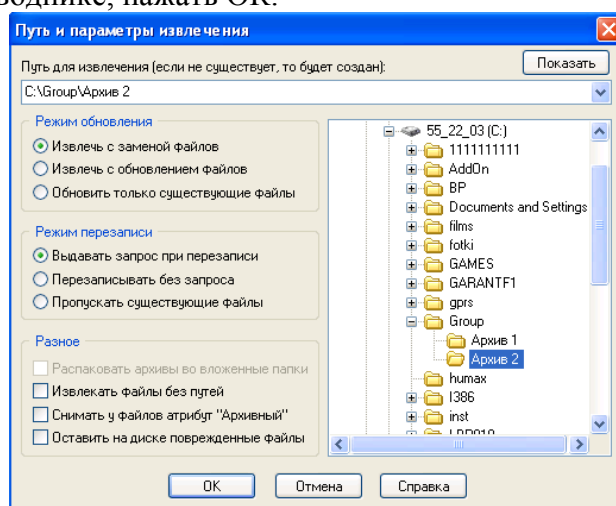


Рисунок 2.11 – Вид диалогового окна Путь и параметры извлечения

11 Просмотрите результат. Закройте все программы.

2.9. Лабораторная работа №9 (2 часа).

Тема: «Возможности графического редактора PAINT и текстового редактора WordPad»

2.9.1 Цель работы: Изучение приемов создания и обработки графических изображений средствами стандартных программ; изучение мультипрограммного режима на примере стандартных программ, «Графический редактор Paint», «Текстовый редактор WordPad»

2.9.2 Задачи работы:

1. Изучение интерфейса приложения Paint
2. Изучение приемов создания рисунков в Paint
3. Ввод и форматирование текста в текстовом редакторе WordPad
4. Создание рисунка и копирование его в редактор WordPad

2.9.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.9.4 Описание (ход) работы:

Задание 1. Запустить графический редактор PAINT: Пуск – Все программы - Стандартные - PAINT

1. Выбираем инструмент «эллипс» и, при нажатой клавише {Shift}, рисуем маленькую окружность. Это будет ягода рябины.
2. С помощью инструмента «заливка» закрашиваем ягоду. Увеличиваем ягоду при помощи «лупы» и прорисовываем детали. Возвращаем ягоде исходный размер.
3. На панели инструментов выбираем «выделение» — не важно, какое — прямоугольной области или произвольной. Внизу, под панелью инструментов, выбираем опцию «без фона».

4. Мышью копируем ягоду желаемое количество раз.

Следующий этап — рисуем листья.

1. Первоначально положение листочков выбираем несколько в стороне от основного рисунка. По окончании работы перенесем их в нужное место. С помощью «эллипса», «заливки» и «лупы» прорисовываем один листок таким же образом, как и ягоду. Здесь надо учесть, что эллипс нужно рисовать сильно вытянутым по вертикали.

2. «Paint» дает возможность вытягивать эллипс только по вертикали или по горизонтали. После того, как закончили прорисовку листочка, нужно придать ему наклонное положение. Выделяем листок, в главном меню выбираем пункт [*Рисунок — Растянуть/Наклонить*].

3. В открывшемся окне выбираем [наклонить — по горизонтали — 10 градусов]

4. Далее копируем рисунок 5...7 раз без фона, таким образом, чтобы все листочки находились в ряд по одной горизонтальной прямой. У нас получилась половинка листа рябины.

5. Последнюю копию листочка отставляем в сторону — из нее мы потом сделаем центральный лист.

6. Теперь нужно сделать вторую половинку листа. Выделяем первую половину и копируем ее вниз. Выделение не снимаем.

7. Выбираем в меню пункт [*Рисунок — Отразить/повернуть*].

8. [Отразить сверху вниз]. Получилась вторая половинка листа. Совмещаем ее с первой.

9. Выделяем теперь весь лист, открываем в меню [*Рисунок — Растянуть/Наклонить*].

10. Выбираем [наклонить — по вертикали — 10 градусов]. Теперь лист практически готов.

11. Выделяем его и перетаскиваем мышью поближе к ягодам.

12. С помощью инструмента «карандаш» или «кисть» прорисовываем веточки. Здесь можно воспользоваться «лупой», чтобы получилось аккуратней.

13. Теперь берем последнюю, оставленную копию листочка. Ее нужно выделить (без фона) и повернуть в горизонтальном направлении. После этого совмещаем листочек с основным рисунком.

14. Инструментом Кисть нарисовать ветку, поместить на ней гроздь рябины (Предварительно выделив и скопировав) (см. Рис.1)

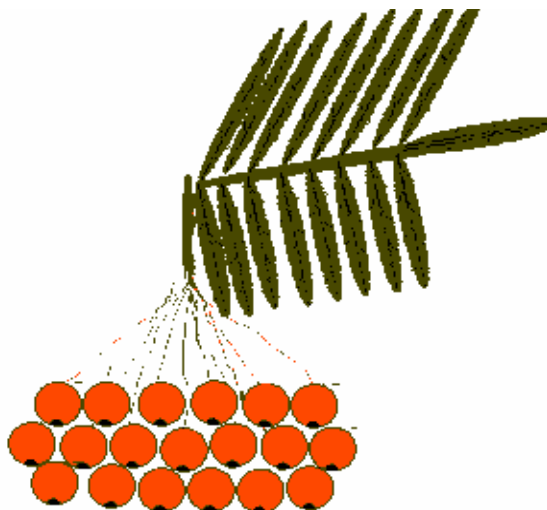


Рисунок 1 – Ветка рябины

Задание 2. Запустите текстовый процессор WordPad (Пуск – Все программы - Стандартные - WordPad).

Введите текст любого стихотворения на тему «Осень, природа» (не менее 8 строк) и каждое четверостишие выделите отдельным цветом (Формат – Шрифт).

Задание 3. Переключиться в PAINT, выделить полученный рисунок, скопировать, и вставить в документ WordPad, ниже Вашего текста. Полученную открытку показать преподавателю.

Задание 4. Письменно ответить на следующие вопросы:

- 1) Каково разрешение экрана Вашего монитора?
- 2) Определите качество цветопередачи дисплея?
- 3) Охарактеризуйте цветовую модель RGB (HSL).
- 4) Что такое видеопамять, что в ней хранится и от чего зависит ее размер?
- 5) Какой объем видеопамати необходим для хранения вашей открытки?

2.10. Лабораторная работа №10-11(4 часа).

Тема: «Текстовый редактор»

2.10.1 Цель работы: Освоить основные приемы работы с документами и форматирования текста в MS Word; освоить основные приемы создания, редактирования и форматирования таблиц в документах текстового процессора MS Word

2.10.2 Задачи работы:

1. Выполнение основных операций по редактированию текстовых документов
2. Выполнение форматирования текстовых документов в MS Word
3. Создание списков в текстовых документах
4. Создание и форматирование таблиц
5. Создание комплексного текстового документа

2.10.3 Перечень приборов, материалов, используемых в лабораторной работе:

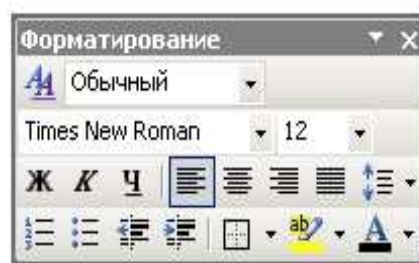
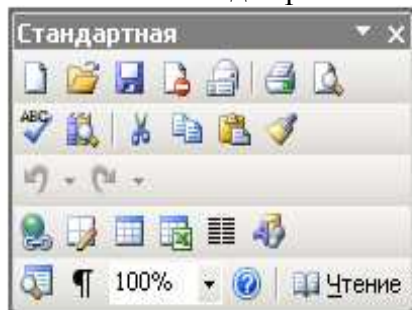
1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.10.4 Описание (ход) работы:

Форматирование шрифтов

Порядок работы.

1. Создайте в папке «Мои документы» папку с именем номера Вашей группы, а в ней свою папку (в качестве имени папки наберите свою фамилию). Для создания папки используйте окно *Мои документы (Пуск/Мои документы)*.
2. Откройте текстовый редактор Microsoft Word (*Пуск /Все программы/ Microsoft Word* или выполните двойной щелчок мыши по ярлыку MS Word).
3. Изучите кнопки панелей инструментов *Стандартная* и *Форматирование* программы *Microsoft Word*, подводя к ним курсор мыши.
4. Установите вид экрана — *Обычный* (меню *Вид* команда *Обычный* (см. Рис. 2.1)).



Р и с у н о к 1. – Панели инструментов программы Microsoft Word (Стандартная и Форматирование)

Задание 1 Набор текста.

Наберите два абзаца текста по приведенному образцу, расположенному ниже (кнопками панелей инструментов установите гарнитуру шрифта — Times New Roman, размер шрифта 14 пт., начертание — обычное, выравнивание — по ширине).

Образец для набора

В перечне перспективных специальностей направление информационные технологии (Information Technologies, IT, ИТ), вычислительная техника и средства коммуникации занимают особое место.

Без компьютеров и ИТ нам уже не обойтись, и люди, в них разбирающиеся, — везде нарахват. Канада, Германия и другие страны дают зеленый свет иностранным ИТ-специалистам, лишь бы заполнить имеющиеся вакансии.

Задание 2 Изменение вида экрана.

Порядок работы

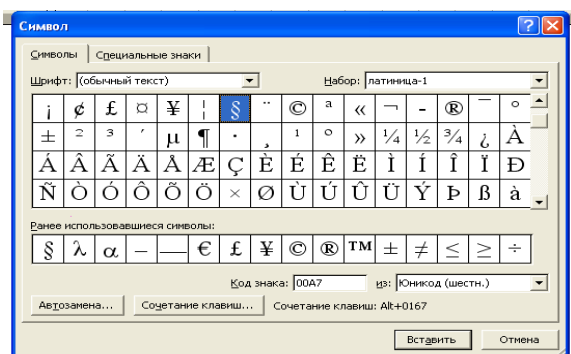
1. Установите режим *Разметка страницы* (меню *Вид/Разметка страницы*).
2. Для выбора оптимального размера документа на экране установите в порядке указанной очередности, ниже перечисленные виды масштабов (*Вид/Масштаб*):
 - стандартный 75 %;
 - произвольный 46 %;
 - по ширине страницы;
 - по ширине текста;

- целая страница.

3. Оставьте для работы с документом последний установленный вид масштаба «По ширине страницы».

Задание 3 Вставка символов.

Вставьте после текста следующие символы (меню *Вставка*, команда *Символ*) (см. Рис. 2.2).



Р и с у н о к 2 – Вставка символов в текст

Задание 4. Форматирование текста. Порядок работы

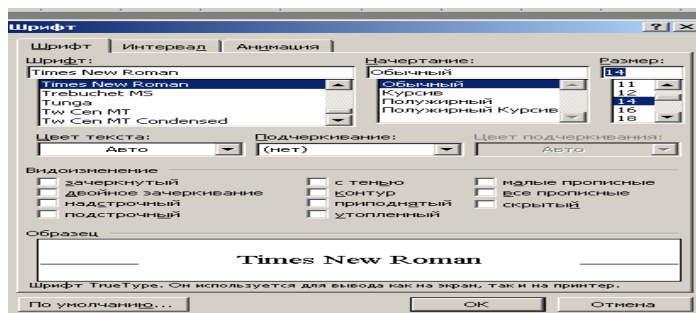
1. Установите в первом абзаце напечатанного текста различные размеры шрифта (выделяя слова мышкой или клавишами [Shift], [Ctrl] и →): первое слово — 10 пт., второе — 14 пт., третье — 18 пт., четвертое — 22 пт. (*Формат/Шрифт* /вкладка *Шрифт*) (рис. 2.3)

2. Оформите во втором абзаце в первой строке каждые два слова разным цветом (*Формат/Шрифт* /вкладка *Шрифт*).

3. Произведите во втором абзаце следующие преобразования, выделяя нужные слова (*Формат/Шрифт/вкладка Шрифт*):

первые два слова оформите полужирным шрифтом;

4. Задайте в первом абзаце разные виды подчеркивания (*Формат/Шрифт/вкладка Шрифт*) (см. Рис. 2.3)



Р и с у н о к 3 – Задание параметров шрифта

первое слово — с одинарным подчеркиванием;

второе — с двойным подчеркиванием;

третье — с пунктирным подчёркиванием.

5. Наберите слово эффект. Скопируйте его пять раз (*Правка/Копировать*, *Правка/Вставить*) и наложите следующие видоизменения (*Формат/Шрифт/вкладка Шрифт*):

эффе^кт (зачеркнутый);

эф^кт (верхний индекс);

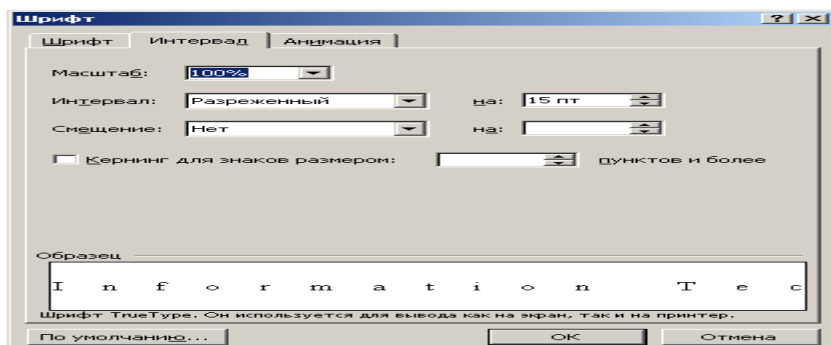
эф_кт (нижний индекс);

ЭФФЕКТ (малые прописные);

ЭФФЕКТ (прописные + контур + полужирный).

6. Наложите на первую строку первого абзаца эффект анимации «Фейерверк» (*Формат/Шрифт*, вкладка *Анимация*).

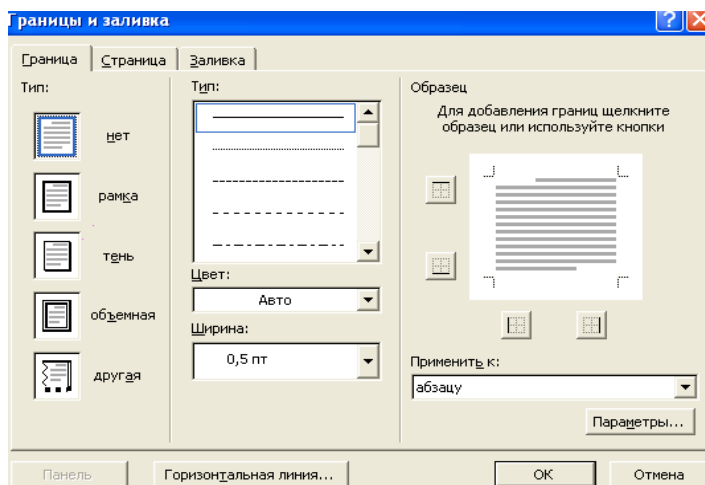
7. В исходном тексте на слова «Information Technologies» установите интервал разрежения на 15 пт. (*Формат/ Шрифт/вкладка Интервал/интервал разреженный на 15 пт.*) (см. Рис. 2.4).



Р и с у н о к 4 – Задание разреженного вида текста

Задание 5 Обрамление и заливка текста.

1. Произведите обрамление первой строки текста. Для этого выделите первую строку, в меню *Формат* выберите команду *Границы и заливка*, на вкладке *Границы* задайте цвет линии — зеленый, толщину — 1,5 пт., тип линии — сплошная линия; применить — к тексту, тип границ — рамка (см. Рис. 2.5).



Р и с у н о к 5 – Обрамление текста рамкой

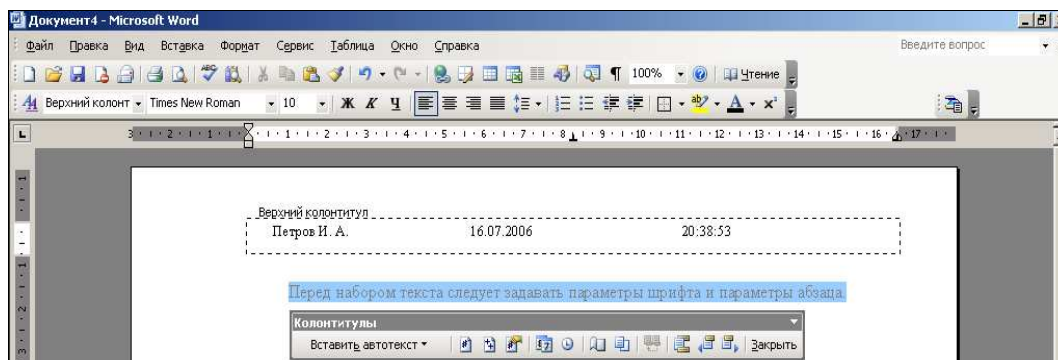
2. Произведите заливку цветом второго абзаца текста. Для этого выделите второй абзац, в меню *Формат* выберите команду *Границы и заливка*, на вкладке *Заливка* выберите цвет и нажмите на кнопку *ОК*.

3. Сохраните набранный документ в свою папку с именем «Фамилия1.doc» (*Файл/Сохранить*).

Задание 6. Задание колонтитулов.

1. Установите вид документа — «Разметка страницы» (*Вид/Разметка страницы*).

2. Задайте колонтитулы документа (*Вид/Колонтитулы*) (см. Рис.2.6).





Р и с у н о к 6. – Задание верхнего колонтитула

Изучите назначение кнопок панели *Колонтитулы*, подводя к ним курсор мыши. В колонтитулы

введите следующую информацию:

в верхний колонтитул — дата, время, ФИО;

в нижний колонтитул — название учебного заведения и номера страниц.

Дата и время задаются кнопками  панели *Колонтитулы*. Переключение на нижний колонтитул производится кнопкой , — *Верхний/Нижний колонтитул*. Обратите внимание, что при вводе колонтитулов основной текст приобрел бледный цвет и стал недоступен.

Окончание работы с колонтитулами производится кнопкой *Заккрыть* панели *Колонтитулы*. Колонтитулы видны только в виде *Разметка страницы*.

3. Установите параметры страницы, (*Файл/Параметры страницы/вкладка Поля*). Установите расстояние от края до колонтитула равное 1,25 см (*Файл/Параметры страницы/вкладка. Источник бумаги*)

4. Измените вид экрана на обычный (*Вид/Обычный*). Обратите внимание, что колонтитулы не видны в обычном виде документа.

Создание списков в текстовых документах

К р а т к а я с п р а в к а . При создании списков можно использовать два способа: задавать параметры списка в процессе набора текста или наложить вид списка после набора текста.

Первый способ: задание параметров списка в процессе набора текста.

Образец текста с нумерованным списком

Элементарные операции информационного процесса включают в себя:

1. Сбор, преобразование информация, ввод в компьютер.

2. Передачу информации.

3. Хранение и обработку информации.

4. Предоставление информации пользователю.

Порядок работы


1. Откройте текстовый редактор Microsoft Word.

2. Наберите первую строку образца текста, нажмите [Enter].

3. Нажмите на панели инструментов кнопку *Нумерация* , появится цифра 1.

4. Напечатайте текст первого пункта и нажмите [Enter]. Точка ввода переместится на следующую строку, которая сразу получает порядковый номер (2, 3 и т.д.) (или появляется новый значок маркера).

5. Для прекращения списка в очередной строке еще раз нажмите на кнопку *Нумерация* (или *Маркер*), чтобы убрать из строки соответствующий элемент списка.

6. Скопируйте набранный текст с нумерованным списком, преобразуйте уже готовый список из нумерованного в маркированный. Для этого выделите все пункты списка (как набор строк) и нажмите кнопку *Маркер* . Обратите внимание, как изменился вид списка.

Второй способ: наложение параметров списка после набора текста.

1. Наберите текст по приведенному образцу, расположенному ниже, при этом 3 — 6 строки (будущие элементы списка) введите как отдельные абзацы, нажимая клавишу [Enter] в конце каждой строки.

Образец текста

Элементарные операции информационного процесса включают в себя:

сбор, преобразование информации, ввод в компьютер;

передачу информации;


хранение и обработку информации;


предоставление информации пользователю.

2. Скопируйте набранный фрагмент текста три раза *Правка/Копировать*, *Правка/Вставить*).

3. Сформируйте одноуровневый нумерованный список. Для этого выделите списочную часть первого фрагмента (3 — 6 строка), задайте команду *Формат/Список*, выберете вкладку *Нумерованный* и вид цифровой нумерации с круглой скобкой, после чего нажмите *OK*

4. Выделите списочную часть второго фрагмента (3 — 6 строка) и сформируйте одноуровневый маркированный список. Для этого используйте команду *Формат/Список*, выберите вкладку *Маркированный* и задайте вид маркера списка.

5. Выделите списочную часть третьего фрагмента (3 — 6 строка) и сформируйте многоуровневый нумерованный список. Для этого используйте команду *Формат/Список*, вкладку *Многоуровневый* и выберите вид многоуровневого нумерованного списка. Произойдет нумерация в первом уровне списка. Чтобы увидеть нумерацию второго уровня, необходимо выделить вторую строку и увеличить отступ кнопкой панели инструментов *Увеличить отступ* . Аналогично выделите третью строку списка и увеличьте отступ.

Краткая справка. При работе с многоуровневым списком нужно пользоваться кнопками  панели инструментов, которые позволяют присвоить выделенным элементам списка соответствующий уровень.

6. Сохраните документ в своей папке с именем «Документ 3» (*Файл/Сохранить как*), закройте документ и программу.


Создание и форматирование таблиц в MS WORD

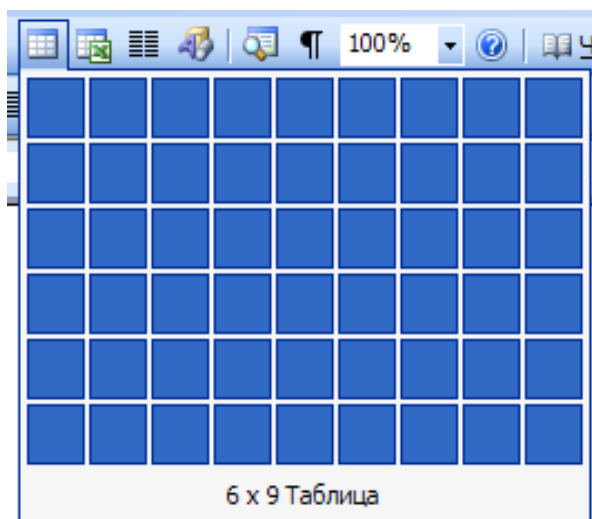
Порядок работы

1. Откройте текстовый редактор Microsoft Word.

2. Установите параметры страницы (размер бумаги — А 4, ориентация — книжная, поля: левое — 3см, правое — 1,5 см, верхнее — 2,5 см, нижнее — 2см), используя команду *Файл/Параметры страницы*.

3. Установите формат абзаца (первая строка — отступ, межстрочный интервал — полуторный).

4. Создайте таблицу 9 x 6, используя команду *Таблица/Вставить/Таблица* (см. Рис.2.7) или кнопку панели инструментов *Добавить таблицу*  (нажатием и продвижением по таблице с помощью левой кнопки мыши)




Р и с у н о к 7– Задание таблицы из панели инструментов

5. Измените ширину колонок: наведите стрелку мыши на вертикальный разделитель таблицы, при этом стрелка мыши примет вид разделителя (двусторонней стрелки); нажатием и продвижением разделителя левой кнопкой задайте нужную ширину столбцов таблицы (см. Табл. 2.1)


Т а б л и ц а 1 -- Образец создания таблицы

Дата	Товарооборот		Выручка	Секции			Состав	Итого
	План	Факт		1	2	3		
2004	13552	13459	4598632	4562	1549	1249	25	1249
2005	15654	15486	5989642	9852	1255	2525	45	1554
2006	13658	14358	1259896	1554	1236	6459	96	15599
2007	56983	58962	125 584	2336	1255	2155	89	12544

6. Произведите объединение ячеек по вертикали и горизонтали как на образце. Для объединения или разбиения ячеек выделите группу ячеек и примените команду *Таблица/Объединить-разбить ячейки* или кнопками 

7. Заполните таблицу данными, перемещаясь по ней мышью или с помощью клавиш [Tab], [Shift]-[Tab].

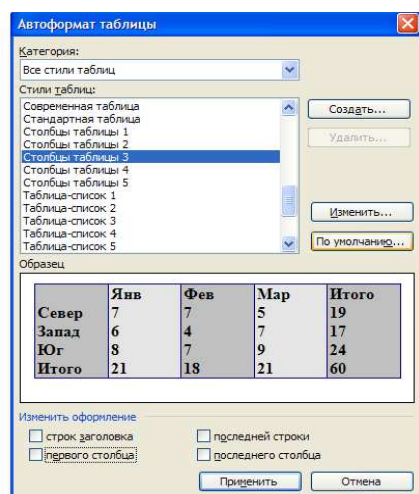
8. Добавьте в таблицу новую строку, для чего поместите курсор в правую ячейку нижней строки таблицы и нажмите клавишу [Tab] или воспользуйтесь командой *Таблица/Добавить/Строки. выше/ниже*, предварительно установив курсор в любую ячейку нижней строки таблицы.

9. Выделите всю таблицу, для чего установите курсор в любую ячейку таблицы и выполните команду *Таблица/Выделить/Таблица* или щелкните левой кнопки мыши по крестообразному указателю мыши в левом верхнем углу таблицы за ее контуром .

10. Произведите обрамления таблицы по образцу, используя команду *Формат/Границы и заливка*.

11. Удалите добавленную пустую нижнюю строку таблицы и сохраните файл в вашей папке с именем «Таблица 1».

12. Скопируйте набранную таблицу в виде нового документа и произведите ее автоформатирование. Для этого установите курсор внутри таблицы, выберите в меню *Таблица* команду *Автоформат* и выберите формат таблицы — «Столбцы таблицы 3» (см. Рис. 2.8).



Р и с у н о к 8– Автоформат таблицы

13. Сохраните отформатированную таблицу в вашей папке с именем «Таблица 2» (*Файл/Сохранить как*).

Задание 2 Произвести вычисления успеваемости студентов в таблице.

Технология выполнения:

1. Создать таблицу по образцу (см. Табл. 2.2)

Т а б л и ц а 2 – Образец таблицы

Сведения об успеваемости студентов								
	Учебная лина	Группа	Средний балл	Всего авало	Отл	Хор	Удовл.	Неуд.
	<i>KCEa</i>							
1		MO11		<i>E4</i>	11	8	6	3
2		MO12			12	7	6	2
3		MO13			10	8	3	1
ИТОГО								

2. Введите формулы для расчета числа студентов каждой группы, сдавших экзамен по дисциплине «Информатика». Для этого установите курсор в ячейку E4 и введите формулу: SUM(RIGHT), предварительно убрав имеющуюся в ней запись. Ввести формулу надо, используя меню *Таблица\Формула*....

3. Произведите те же действия для ячеек E5 — E6.

Введите формулу для расчетов среднего балла по дисциплине «Информатика» для группы MO11. Для этого установите курсор в ячейку D4 и введите формулу: $= (F4*5 + G4*4 + H4*3 + I4*2) / E4$. Выберите формат числа 0,00.

4. Введите аналогичные формулы в ячейки D5 — D6.

Введите формулу для расчета общего числа студентов, сдавших экзамен по каждой дисциплине на отлично, хорошо и т.д. Для этого установите курсор в ячейку F7 и введите формулу: SUM(ABOVE), затем аналогично в ячейку G7 и т.д.

Создание комплексного текстового документа

Задание.1. Создать гипертекстовый документ «Программное обеспечение компьютера»

Порядок работы

На первой странице документа разместить текст:

Программное обеспечение компьютера – это множество программ хранящихся в долговременной памяти компьютера. Множество программ можно разделить на три группы: системные, прикладные, системы программирования.

1. На второй странице :

Назначение системных программ – управление работой компьютера.

2. На третьей странице :

Назначение прикладных программ – решение информационных задач.

3. На четвертой странице :

Назначение система программирования – создание новых программ.

2. Установить закладку к тому месту в документе, куда необходимо осуществить переход.

Вставка – Закладка – Имя закладки

3 Выделить текст, который предполагается использовать как гиперссылку. (например системные)

4. Вставка – Гиперссылка

5. В списке *Связать с* выберите параметр *местом в этом документе.*

Выберите в списке *заголовков или закладку* для ссылки. ОК

6. Гиперссылка выделена цветным подчеркнутым текстом. Подвести курсор, отследить работу гиперссылки.

Задание 2. Записать макрос для вставки (создания) таблицы: 6 столбцов, 7 строк ; толщина линий рамки ячейки D1- 2,25пт ,толщина линий рамки таблицы - 1,5пт, цвет рамки ячейки A4 – синий, цвет заливки столбца E – желтый , цвет заливки ячейки B3 – зелёный

Тщательно продумать порядок действия.

a.В меню Сервис выделите пункт Макрос, а затем выберите команду Начать запись.

b.В поле Имя макроса введите имя нового макроса «Таблица».

c.Нажмите кнопку ОК, чтобы начать запись макроса.

Выполните действия, которые следует включить в макрос, используя пункты меню (Таблица, Формат и т.д.) в текстовом редакторе. Для перемещения курсора, а также выделение, копирование и перемещение текста, необходимо использовать клавиатуру.

Остановить запись макроса с помощью меню Сервис - Макрос.

Установить курсор мыши на новую строку. Выполнить команду для проверки макроса: Сервис – Макрос – Макросы – Имя макроса – Выполнить

2.11 Лабораторная работа №12-13(4 часа).

Тема: «Программные средства решения задач презентационного представления документации.»

2.11.1 Цель работы: Изучить функции и возможности среды Microsoft Power Point.

2.11.2 Задачи работы:

1. Изучение информационной технологии разработки презентации

2. Задание эффектов и демонстрация презентации

2.11.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер

2. Операционная система Microsoft Windows.

3. Microsoft Office

2.11.4 Описание (ход) работы:

Запустите PowerPoint. Выберите в меню Справка команду Справка: Microsoft PowerPoint. В раскрывшемся затем окне Справка Microsoft PowerPoint выберите вкладку Содержание и

изучите тему Приступая к работе, раздел Новые возможности в Microsoft PowerPoint. Открыв тему Получение справки, изучите разделы справочной информации о способах получения справочной информации во время работы. В разделе Создание презентаций изучите справку о различных способах создания презентаций.

Выбрав вкладку Мастер ответов, задайте вопрос «создание презентаций» и щелкните кнопку «Найти». В списке найденных разделов изучите вариант Создание новой презентации на основе существующей. Выбрав вкладку Указатель, введите ключевое слово «шаблон», затем щелкните кнопку «Найти». Выбрав в списке ключевых слов шаблон, щелкните в списке найденных разделов на разделе Применение шаблона оформления. Изучив справочную информацию об использовании шаблонов, образцов, цветовых схем и макетов слайдов для управления внешним видом слайдов, закройте окно справки.

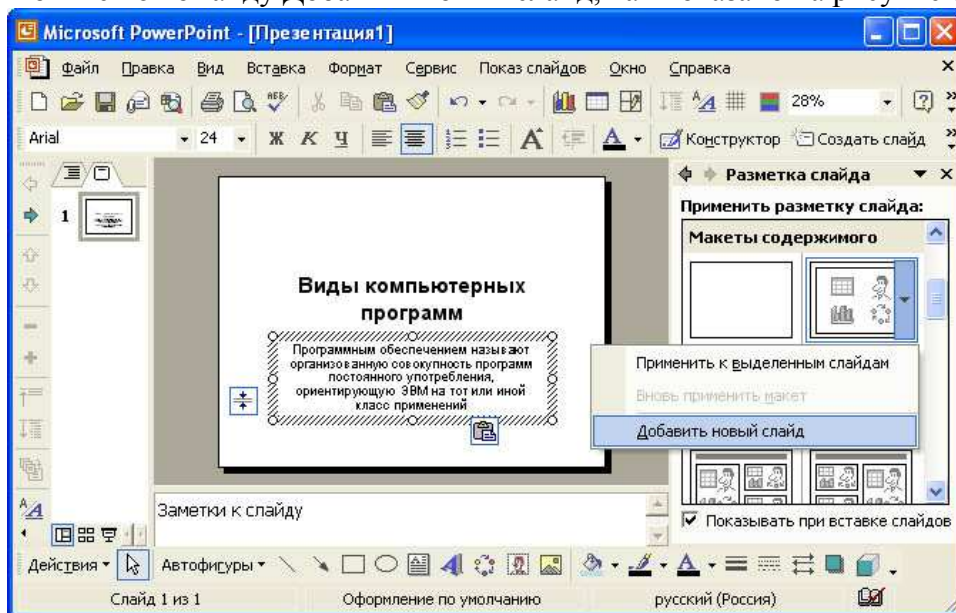
Задание 1. Создать презентацию на тему Виды компьютерных программ. Для этого выполните следующую последовательность действий.

1. Для создания презентации выберите команду Создать в меню Файл, а затем в панели задачи Создать презентацию щелкните ссылку Новая презентация. В окне Создание слайда выберите автомакет Титульный слайд и щелкните кнопку «ОК».

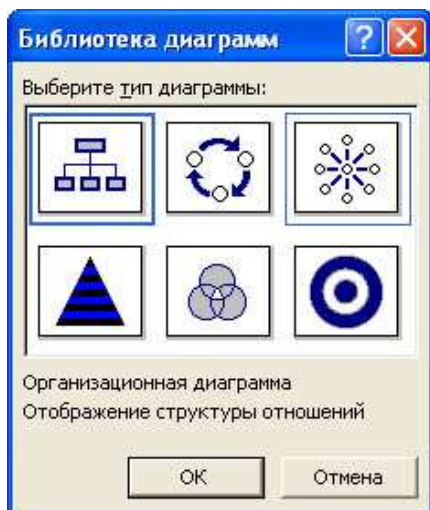
2. Щелкнув мышью в области заголовка слайда, введите заголовок презентации «Виды компьютерных программ». В подзаголовок слайда введите текст «Программным обеспечением называют организованную совокупность программ постоянного употребления, ориентирующую ЭВМ на тот или иной класс применений».

Используя инструменты панели Форматирование, установите нужные параметры текстам заголовка и подзаголовка.

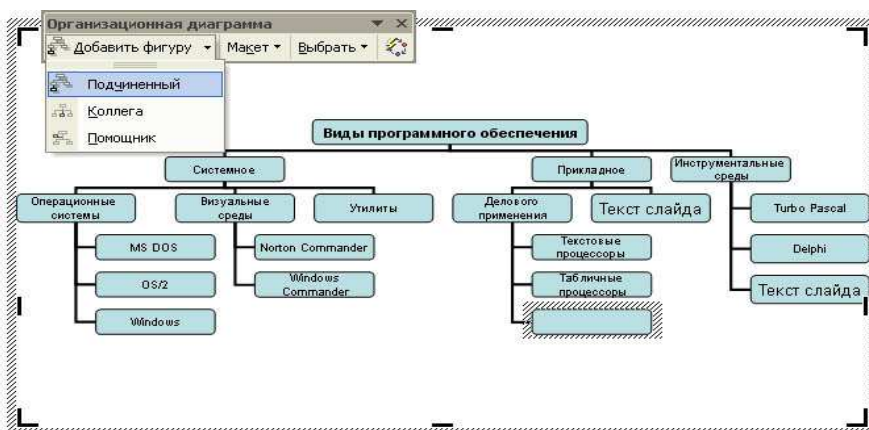
3. Для вставки нового слайда с организационной диаграммой выберите в панели задач Разметка слайда макет Объект. Щелкнув на макете Объект стрелку справа, выберите в контекстном меню команду Добавить новый слайд, как показано на рисунке ем, и многое другое.



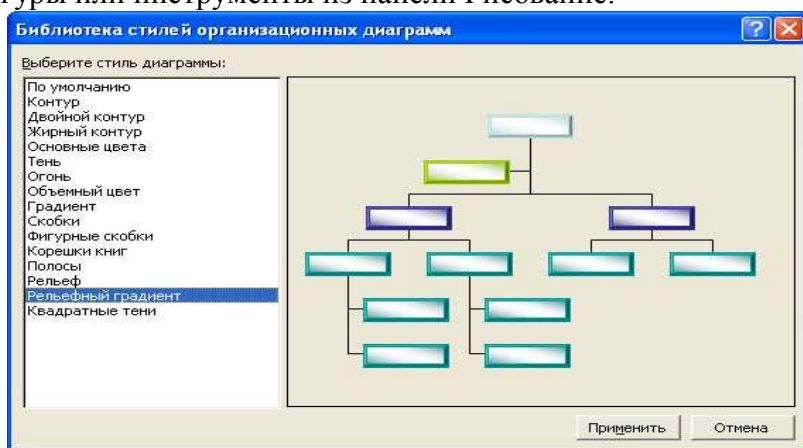
Щелкнув на новом слайде в панели Рисование по кнопке «Добавить организационную диаграмму». В появившемся окне Библиотека диаграмм выберите тип диаграммы, как показано на рис.



Введите заголовок диаграммы «Типы программного обеспечения», затем добавьте нужные фигуры и введите элементы диаграммы, как показано на рис.



Для определения параметров текста можно использовать команды меню Текст. Для определения параметров элементов диаграммы применяйте команды контекстного меню Формат Автофигуры или инструменты из панели Рисование.



Для изменения стиля организационной диаграммы щелкните кнопку «Автоформат» в панели инструментов Организационная диаграмма и выберите стиль, как показано на рис. 4. Завершите создание организационной диаграммы и установите нужный размер вставленной диаграммы, используя маркеры объекта.

4. Для вставки нового слайда выберите в панели задачи Разметка слайда макет Заголовок, текст и графика и, щелкнув стрелку справа, выберите команду Добавить новый слайд. В область заголовка введите текст «Прикладные программы». В область текстовой рамки слайда введите текст о назначении прикладных программ, например, «Прикладное программное обеспечение - программы для решения класса задач в определенной области применения систем обработки данных. Они непосредственно обеспечивают выполнение необходимых пользователям

работ».

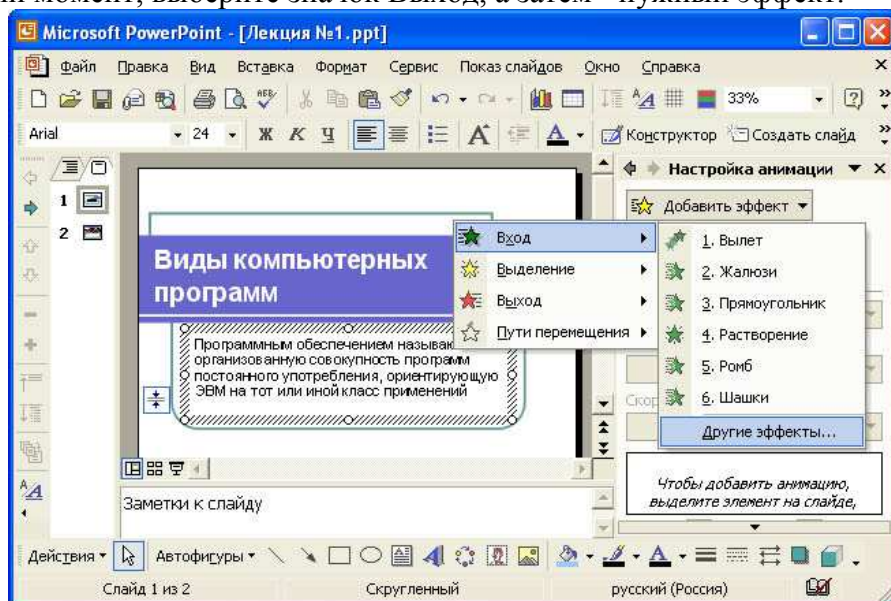
В область графики вставьте картинку, для чего дважды щелкните кнопку «Вставка картинки» на слайде, а затем в раскрывшемся окне Microsoft Clip Gallery выберите нужный рисунок. Можно вставить рисунок из файла, для чего, указав область рисунка на слайде, выберите в меню Вставка команду Рисунок, а затем опцию Из файла. В окне Добавить рисунок найдите нужную папку и файл. Щелкнув кнопку «Вставка», завершите вставку рисунка и задайте нужную позицию и размеры вставленному рисунку.

5. Действуя аналогично, добавьте слайды и введите текстовую и графическую информацию об остальных типах программного обеспечения.

Задание 2. Оформите презентацию и настройте анимацию объектов. Для этого выполните следующую последовательность действий.

Оформите презентацию с использованием готовых шаблонов оформления, для чего откройте в области задач панель Дизайн слайда. Выбирая в поле Применить шаблон оформления различные шаблоны оформления, просматривая варианты дизайна слайда, выберите подходящий шаблон. После этого выполнится настройка цветовой схемы, и все слайды презентации будут переоформлены в соответствии с выбранным шаблоном.

Для настройки анимации слайдов откройте в области задач панель задачи Настройка анимации. Выбрав слайд и выбирая элемент слайда, настройте эффекты анимации. Для этого щелкните кнопку «Добавить эффект» и выполните одно или нескольких следующих действий. Если во время показа слайдов требуется ввести текст или объект в сопровождении определенного визуального эффекта, укажите значок Вход, а затем выберите эффект. Если требуется добавить определенный визуальный эффект в текст или объект, находящиеся на самом слайде, укажите значок Выделение, а затем выберите нужный эффект. Если требуется добавить определенный визуальный эффект в текст или объект, который вызывает удаление текста или объекта со слайда в заданный момент, выберите значок Выход, а затем - нужный эффект.



Для просмотра заданного эффекта анимации щелкните кнопку «Просмотр». Для изменения порядка появления анимации или ряда анимированных фрагментов, выбрав параметр в списке настроек анимации, перетащите его в другое место списка.

Для точной настройки параметров анимационных эффектов, щелкнув правой кнопкой мыши на стрелке справа строки с надписью эффекта в списке настройки анимации, откройте контекстное меню и выберите в нем команду Параметры эффектов. После этого откроется окно эффекта с вкладками Эффект, Время и Анимация объекта. Для уточнения назначения параметров воспользуйтесь подсказкой «Что это такое?». Выбирая вкладки Эффект, Время и Анимация объекта, задайте нужные параметры и щелкните кнопку «ОК».

Для ввода заметок докладчика, указав слайд, щелкните в области заметок и введите текст заметок.

Закончив работу над презентацией, выберите в меню Файл команду Сохранить. Открыв нужную папку, присвойте презентации имя, например, Виды компьютерных программ, и нажмите кнопку «Сохранить».

Для демонстрации презентации щелкните кнопку «Показ слайдов» на панели инструментов. По окончании просмотра презентации закройте окно PowerPoint.

2.12 Лабораторная работа №14 (2 часа).

Тема: «Табличный процессор Excel: работа с листами и графиками»

2.12.1 Цель работы: Изучение информационной технологии организации работы с листами и графиками в таблицах MS Excel.

2..2 Задачи работы:

1. Научиться выполнять элементарные вычисления
2. Выполнять работу с листами и графиками

2.12.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.12.4 Описание (ход) работы:

1.Элементарные вычисления

Задание:

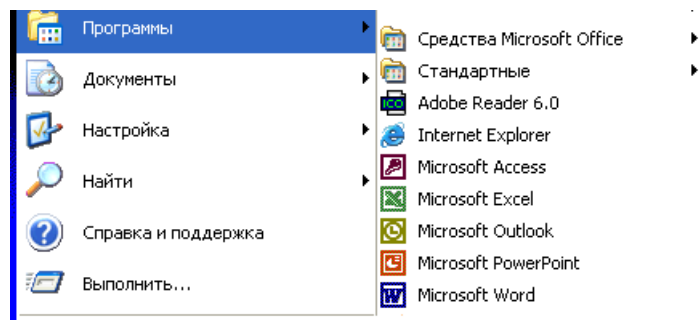
1. Открыть Excel и создать новый документ.
2. Заполнить столбец значений аргумента с заголовком.
3. Вычислить значения заданной функции во втором столбце.
4. Открыть второй лист.
5. Заполнить значения аргументов для функции двух переменных в два столбца.
6. Вычислить значение функции двух переменных в третьем столбце.
7. Открыть третий лист.
8. Заполнить значения аргумента x в первый столбец, начиная со второй строки, а значения аргумента y в первую строку, начиная со второго столбца.
9. Вычислить значения функции двух переменных.
10. Переименовать листы.
11. Сохранить документ.

Пример выполнения задания:

Задание:

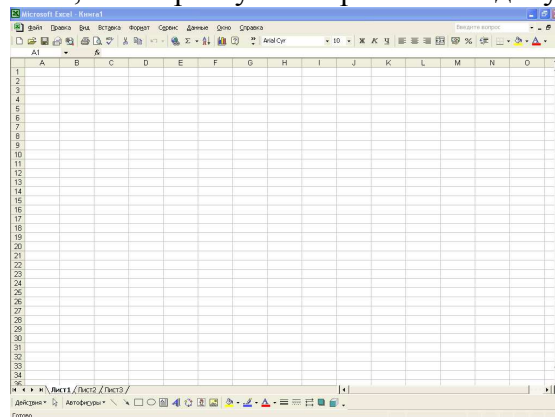
1. $f(x)=x^2+x+3$, где x меняется от 0 до 5 с шагом 0,5.
2. $F(x,y)=x^2+y^2-10$, где x меняется от 0 до 4 с шагом 0,5, а y меняется от 0 до 4 с шагом 0,25.

1. Для выполнения задания 1 найдем в меню «Пуск» строчку «Программы», вы увидите выпадающее меню:

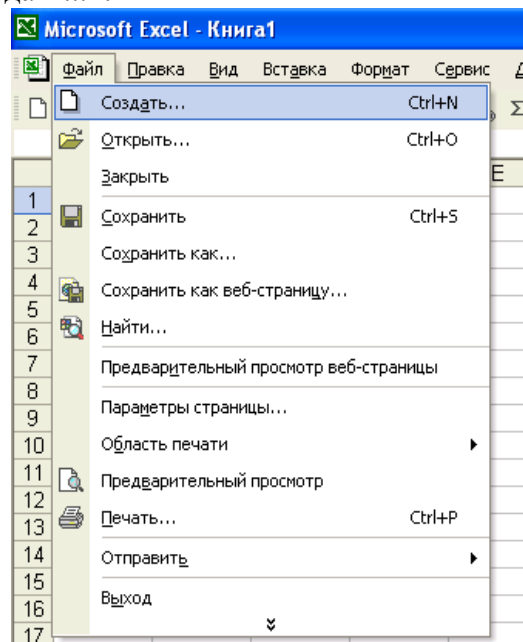


в котором надо выбрать строку  Microsoft Excel .

В результате вы увидите окно, в котором уже открыт новый документ.



Если в окне нет документа или открыт чужой документ, необходимо выбрать пункт меню «Файл» и там найти команду «Создать...»:

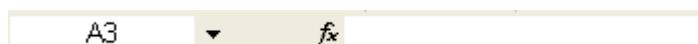


Откроется новый документ.


Отметим, что чистый документ представляет из себя таблицу, у которой проименованы столбцы и пронумерованы строки. Имена столбцов задаются латинскими буквами в алфавитном порядке до столбца с именем **Z**, следующий столбец имеет имя **AA**, затем **AB**, **AC** и т.д.

Примечание. Если имена ячеек задаются в виде **RIC1**, изменить на латинский алфавит можно с помощью **Сервис\Параметры...**, там выбрать закладку «Общие» и убрать флажок из поля «Стиль ссылок **RIC1**» в разделе «Параметры».


В остальном оформление окна не отличается от окна Word, за исключением строки формул:



Слева в ней указывается адрес активной ячейки, справа набираются формулы начиная со знака «=», а также имеется ряд кнопок:

 , которая вызывает встроенные функции Excel и присутствует всегда;

 , которая стирает формулу из строки формул;

 – вычисляет формулу.

***Примечание.** Рассматривать работу с панелью инструментов «Стандартная» и «Форматирование» не будем, так как обе эти панели подробно разобраны в пособии «Лабораторный практикум. Часть 1. Word». Отметим только, что применять форматирование к ячейкам можно или до набора в них текст, или к уже набранному тексту. Для этого надо выйти из режима набора текста в ячейку (щелчок левой кнопки мыши на любой другой ячейке) и установить на нее курсор ячейки, теперь можно применять форматирование.*

В Excel имеется несколько видов курсоров, рассмотрим их все:

| – текстовый курсор, позволяющий набирать текст в ячейки строку формул.

 – курсор мыши, позволяющий производить выбор и выделение ячеек.

+ – курсор мыши, позволяющий производить копирование функций, появляется в нижнем правом углу выделенной (рабочей) ячейки (на маленьком квадратике).

 – курсор ячейки.

 – курсор мыши, позволяющий перемещать ячейки.

↔ и ⇕ – курсоры мыши, позволяющие расширять строки и столбцы. Появляются только в заголовке столбцов и нумерации строк.

↓ и → – курсоры мыши, позволяющие выделять целиком столбец или строку.

2. Теперь мы можем начать работать в Excel.

Заполним столбец значений аргумента x , который изменяется от 0 до 5 с шагом 0,5.

Это можно сделать несколькими способами:

Способ 1. В ячейку **A1** запишем « x » и отформатируем его по центру. В ячейку **A2** запишем 0. В ячейку **A3** – 0,5. В ячейку **A4** – 1 и т.д. до ячейки **A12**, в которую записывается 5.

Способ 2. В ячейку **A1** запишем « x » и отформатируем его по центру. В ячейку **A2** запишем 0. В ячейку **A3** – 0,5. Затем выделим ячейки **A2** и **A3**, установим курсор мыши в нижний правый угол (он примет вид «+»), нажмем левую кнопку мыши и, удерживая ее, перемещаем до ячейки **A12** включительно. (Отметим, что справа от курсора появляется желтенький прямоугольник, в котором при движении мыши меняются цифры. Эти цифры указывают значение в ячейке, мимо которой движется указатель мыши.) Данный способ эффективен, если параметр изменяется с постоянным шагом.

Способ 3. В ячейку **A1** запишем « x » и отформатируем его по центру. В ячейку **A2** запишем 0. В ячейку **A3** установим курсор и наберем формулу:

=A2+0,5

нажмем **Enter**. Затем установим курсор ячейки на **A3**, а курсор мыши – на нижний правый угол («+») и, удерживая левую кнопку мыши, растянем до **A12**.

В результате получим:

	A1	
	A	B
1	x	
2	0	
3	0,5	
4	1	
5	1,5	
6	2	
7	2,5	
8	3	
9	3,5	
10	4	
11	4,5	
12	5	
13		

3. Вычислим значения заданной функции $f(x)=x^2+x+3$.

Для этого в ячейке **B1** наберем текст «f(x)» и центрируем его. В ячейку **B2** наберем формулу:

=A2*A2+A2+3

т.е. вместо «х» вы набираете адрес ячейки, в которой находится соответствующий аргумент.

Нажмите кнопку «Вычисление формулы».

Затем, установив курсор мыши в нижний правый угол (+), растяните формулу до ячейки **B12**.

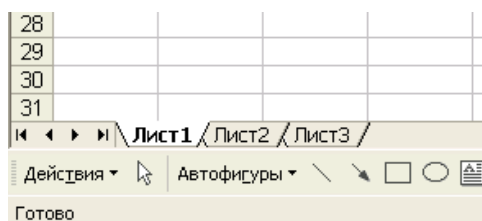
В результате получим:

	A	B	C	D
1	x	f(x)		
2	0	3		
3	0,5	3,75		
4	1	5		
5	1,5	6,75		
6	2	9		
7	2,5	11,75		
8	3	15		
9	3,5	18,75		
10	4	23		
11	4,5	27,75		
12	5	33		

Как видите, курсор мыши стоит на ячейке **B2**, а в строке формул указаны адрес этой ячейки и формула, набранная в ней, так можно просмотреть все формулы в столбце **B**. Вы можете заметить, что формулы отличаются только номером строки, в остальном они одинаковы.

Примечание. Если растягивать формулу вниз или вверх, то изменяется номер строки, а если растягивать формулу вправо или влево, то изменяется буква столбца.

4. У Excel есть еще одно отличие от редактора Word. Каждый документ Excel состоит из нескольких отдельных *листов*, выполняющих роль отдельных документов, но при этом с возможностью ссылки друг на друга. Закладки листов находятся в нижней части окна Excel, сразу над строкой состояния:



Вычислим функцию двух переменных на листе 2. Для этого откроем второй лист (щелкнув левой кнопкой мыши на закладке «Лист2»).

5. Заполним значения аргумента в столбцах **A** и **B**. Напомним, что у функции двух аргументов при одном значении x может быть много различных значений y .

В ячейку **A1** запишем «х» по центру. В ячейку **B1** – «у» по центру. В ячейку **A2** – 0. В ячейку **B2** – 0. В ячейку **A3** – 0. В ячейку **B3** – 0,25.

Теперь выделим четыре ячейки от **A2** до **B3**, установим курсор мыши в нижний правый угол и, удерживая левую кнопку мыши, растягиваем до строки **18** включительно, получим:

	A	B
1	x	y
2	0	0
3	0	0,25
4	0	0,5
5	0	0,75
6	0	1
7	0	1,25
8	0	1,5
9	0	1,75
10	0	2
11	0	2,25
12	0	2,5
13	0	2,75
14	0	3
15	0	3,25
16	0	3,5
17	0	3,75
18	0	4
19		

Теперь посчитаем для второго значения x . В ячейку **A19** набираем 0,5. В ячейку **B19** – 0. В **A20** – 0,5. В **B20** – 0,25. Выделяем ячейки **A19–B20** и растягиваем черным крестиком (+), получаем:

16	0	3,5
17	0	3,75
18	0	4
19	0,5	0
20	0,5	0,25
21	0,5	0,5
22	0,5	0,75
23	0,5	1
24	0,5	1,25
25	0,5	1,5
26	0,5	1,75
27	0,5	2
28	0,5	2,25
29	0,5	2,5
30	0,5	2,75
31	0,5	3
32	0,5	3,25
33	0,5	3,5
34	0,5	3,75
35	0,5	4
36		
37		

И так заполним для каждого x до значения 4 с шагом 0,5. Вы получите длинную таблицу из 154 строк.

Примечание. Отметим, что если вы растянули недостаточно или, наоборот, слишком много, то, пока не снято выделение с ячеек, вы можете увеличить или уменьшить таблицу вычислений, взявшись за нижний правый угол всего выделения и потянув вниз или вверх.

6. Вычислим функцию $f(x,y)=x^2+y^2-10$ в третьем столбце. Для этого в ячейку **C1** наберем «f(x,y)» и центрируем. В ячейку **C2** наберем:

$$=A2*A2+B2*B2-10$$

и вычислим формулу.

После чего растянем формулу за нижний правый угол и получим:

	A	B	C
1	x	y	f(x,y)
2	0	0	-10
3	0	0,25	-9,9375
4	0	0,5	-9,75
5	0	0,75	-9,4375
6	0	1	-9
7	0	1,25	-8,4375
8	0	1,5	-7,75
9	0	1,75	-6,9375
10	0	2	-6
11	0	2,25	-4,9375
12	0	2,5	-3,75
13	0	2,75	-2,4375
14	0	3	-1
15	0	3,25	0,5625
16	0	3,5	2,25
17	0	3,75	4,0625
18	0	4	6
19	0,5	0	-9,75
20	0,5	0,25	-9,6875
21	0,5	0,5	-9,5
22	0,5	0,75	-9,1875
23	0,5	1	-8,75
24	0,5	1,25	-8,1875
25	0,5	1,5	-7,5
26	0,5	1,75	-6,6875
27	0,5	2	-5,75

7. Теперь откроем лист 3, для этого надо щелкнуть левой кнопкой мыши по закладке «Лист3».

8. Заполним значения аргумента для функции двух переменных, только теперь значения x расположим в столбец, а значения y – в строку.

Ячейку **A1** оставим пустой. В ячейку **A2** запишем 0. В **A3** – 0,5. Теперь выделим ячейки **A2** и **A3** и растянем за нижний правый угол до строки 10.

Заполняем значения y . Для этого в ячейку **B1** запишем 0. В **C1** – 0,25. Выделим ячейки **B1** и **C1**, а далее растянем за нижний правый угол до ячейки **R1**. Мы получили:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		0	0,25	0,5	0,75	1	1,25	1,5	1,75	2	2,25	2,5	2,75	3	3,25	3,5	3,75	4
2		0																
3		0,5																
4		1																
5		1,5																
6		2																
7		2,5																
8		3																
9		3,5																
10		4																
11																		

9. Теперь заполним значения функции в ячейку **B2**:

$$=A2*A2+B1*B1-10$$

Вычислим ее. Но теперь, поскольку ссылки у нас относительные, если мы будем растягивать формулу, то при растягивании вниз у нас будут смещаться ссылки для значений y , а при растягиванию вправо будут смещаться буквы в ссылках значений x .

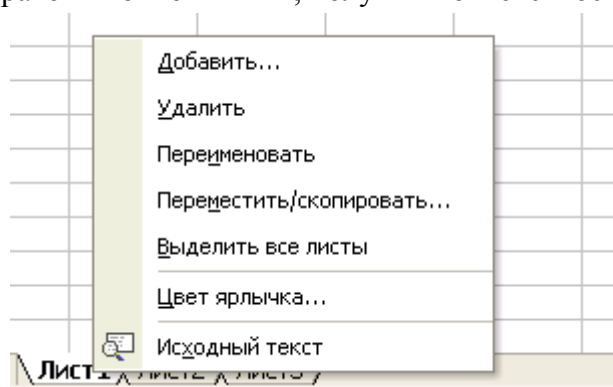
Нам необходимо зафиксировать в ссылках для x столбец **A** (сделать абсолютную ссылку на столбец **A**), а в ссылках для y – строку **1** (сделать абсолютную ссылку на строку **1**). Для того чтобы фиксировать строки и столбцы в ссылках, в Excel предназначен знак «\$». Тогда в нашей формуле правильно записать ссылки для x – **\$A2**, а ссылки для y – **\$B1**. Мы получим:

$$= \$A2* \$A2+ \$B1* \$B1-10$$

Вот эту формулу уже можно растянуть. Заметим, что растягивать по диагонали нельзя, данное действие выполняется в два приема: сначала растягивают вниз и отпускают кнопку мыши, далее, не снимая выделения, снова берут за нижний правый угол и растягивают вправо. В результате получаем:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		0	0,25	0,5	0,75	1	1,25	1,5	1,75	2	2,25	2,5	2,75	3	3,25	3,5	3,75	4
2	0	-10	-9,938	-9,75	-9,438	-9	-8,438	-7,75	-6,9375	-6	-4,938	-3,75	-2,438	-1	0,5625	2,25	4,0625	6
3	0,5	-9,75	-9,688	-9,5	-9,188	-8,75	-8,188	-7,5	-6,6875	-6,75	-4,688	-3,5	-2,188	-0,75	0,8125	2,5	4,3125	6,25
4	1	-9	-8,938	-8,75	-8,438	-8	-7,438	-6,75	-5,9375	-5	-3,938	-2,75	-1,438	0	1,5625	3,25	5,0625	7
5	1,5	-7,75	-7,688	-7,5	-7,188	-6,75	-6,188	-5,5	-4,6875	-3,75	-2,688	-1,5	-0,188	1,25	2,8125	4,5	6,3125	8,25
6	2	-6	-5,938	-5,75	-5,438	-5	-4,438	-3,75	-2,9375	-2	-0,938	0,25	1,5625	3	4,5625	6,25	8,0625	10
7	2,5	-3,75	-3,688	-3,5	-3,188	-2,75	-2,188	-1,5	-0,6875	0,25	1,3125	2,5	3,8125	5,25	6,8125	8,5	10,3125	12,25
8	3	-1	-0,938	-0,75	-0,438	0	0,563	1,25	2,0625	3	4,0625	5,25	6,5625	8	9,5625	11,25	13,0625	15
9	3,5	2,25	2,3125	2,5	2,813	3,25	3,813	4,5	5,3125	6,25	7,3125	8,5	9,8125	11,25	12,813	14,5	16,3125	18,25
10	4	6	6,0625	6,25	6,563	7	7,563	8,25	9,0625	10	11,063	12,25	13,563	15	16,563	18,25	20,0625	22

10. Теперь переименуем листы Excel. Для этого установим курсор на закладку «Лист1» и щелкнем один раз правой кнопкой мыши, получим контекстное меню вида:



в котором выберем пункт «Переименовать» и запишем туда « $f(x)$ ».

Повторим процедуру для второго и третьего листа, назвав их « $f1(x,y)$ » и « $f2(x,y)$ » соответственно. Мы получим:

30																		
31																		
	«	»	«	»	«	»	«	»	«	»	«	»	«	»	«	»	«	»

11. Для сохранения документа выполним одно из следующих действий:

- **\Файл\Сохранить...** В открывшемся окне в поле «Папка» указываем диск и папку, куда сохраняем файл. В поле «Имя» набираем имя сохраняемого документа. Нажимаем

кнопку «Сохранить».

- Нажимаем кнопку  на панели инструментов «Стандартная» и в полученном окне проводим вышеуказанные действия.

2. Работа с листами и графиками

Задание:

1. Открыть Excel и созданный ранее документ.
2. Создать новый лист (Лист4) и переименовать его в $f3(x)$.
3. Вычислить значения заданной функции в столбце **A** на новом листе, взяв значения аргумента с Листа1, названного $f(x)$.
4. Построить график полученной функции на этом же листе.
5. По заданной таблице построить поверхность на отдельном листе.
6. Сохранить документ.


Пример выполнения задания:

Задание:

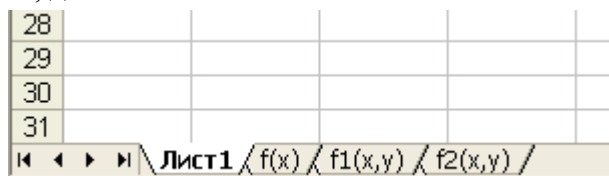
1. $f(x)=x^2+x+3$, где x меняется от 0 до 5 с шагом 0,5.
2. $F(x,y)=x^2+y^2-10$, где x меняется от 0 до 4 с шагом 0,5, а y меняется от 0 до 4 с шагом 0,25.

1. Откроем Excel и загрузим в него уже имеющийся файл, для чего выполним одно из следующих действий:

- \Файл\Открыть... . Открывается диалоговое окно «Открыть», очень похожее на окно «Сохранить», в поле «Папка» открываем диск и папку, где был записан файл, в окне выбора находим свой файл и дважды щелкаем по нему левой кнопкой мыши (или один щелчок по файлу левой кнопкой мыши, а когда его имя появится в поле «Имя», нажать кнопку «Открыть»).

- Нажать кнопку  на панели инструментов «Стандартная», далее появится диалоговое окно «Открыть», действия в котором мы уже описывали.

2. Создадим новый лист, для этого выполним \Вставка\Лист и получим:



Так как мы уже переименовали листы, то новый лист опять носит название Лист1. Переименуем его в $f3(x)$.

3. Произведем вычисление значений функции $f(x)$. Для этого в ячейку **A1** наберем « $f(x)$ » и центрируем, а в ячейку **A2** наберем:

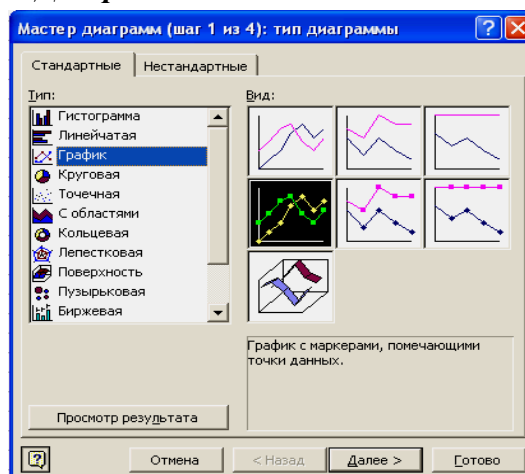
$$='f(x)!'A2*'f(x)!'A2+'f(x)!'A2+3$$

где ' $f(x)$ '! – ссылка на лист с именем $f(x)$.

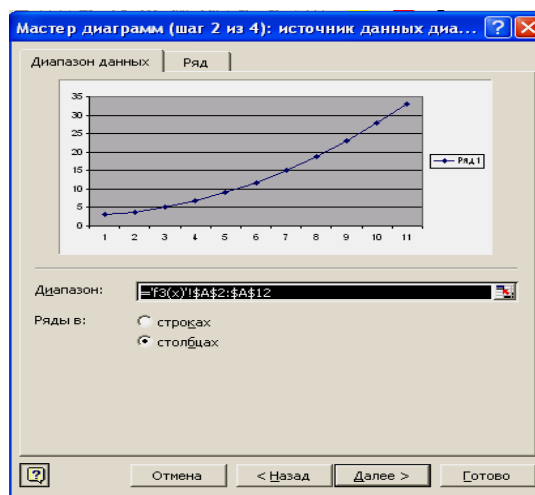
Теперь вычислим формулу и растянем ее до строки **12**, поскольку значения аргумента у нас посчитаны только до этой строки. Получим:

	A2	fx = f(x)!A2*f(x)!A2+f(x)!A2+3			
	A	B	C	D	E
1	f(x)				
2	3				
3	3,75				
4	5				
5	6,75				
6	9				
7	11,75				
8	15				
9	18,75				
10	23				
11	27,75				
12	33				

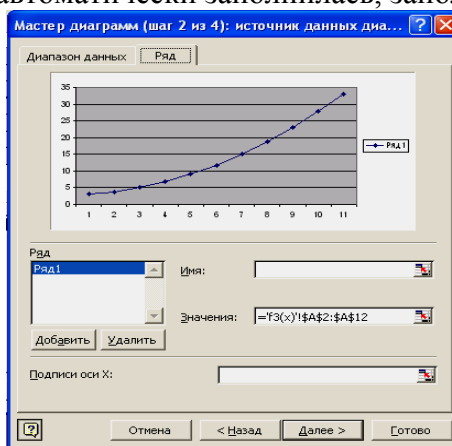
4. Теперь построим график данной функции на этом же листе, для этого выделим ячейки от A2 до A12 и выполним **Вставка\Диаграмма...** Появится диалоговое окно вида:



На закладке «Стандартные» в поле выбора «Тип» выберем «График», в поле выбора «Вид» выберем первый во втором ряду и нажмем кнопку «Далее», получим окно:



Первая закладка этого окна автоматически заполнилась, заполним вторую закладку:



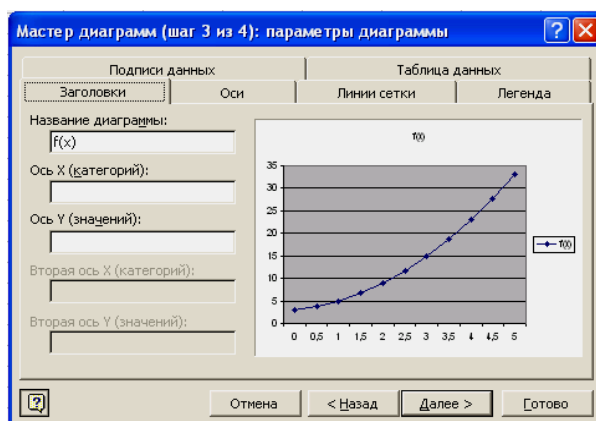
Для этого в поле «Имя» запишем

$=f_3(x)!$ A\$1

в поле «Подписи оси X» заполним

$=f_3(x)!$ A\$2:A\$12

Нажмем кнопку «Далее» и получим окно:



Данное окно позволяет произвести оформление графика. Начнем с первой закладки «Заголовки». Здесь в поле ввода «Название диаграммы:» вы вводите название вашего графика, назовем его «График функции», в поле «Ось X (категорий)...» введем «x», в поле «Ось Y (значений)» вводим «f(x)».

Открываем вторую закладку «Оси»:



Здесь можно отключить оси с разбиением и подписями («ось X» и «ось Y»), а также выбрать один из трех видов осей.

На третьей закладке «Линии сетки» вы можете включить и выключить основные и дополнительные линии сетки, как по x , так и по y :



На закладке «Легенда» вы можете добавить или убрать легенду, а также выбрать ее месторасположение на графике:

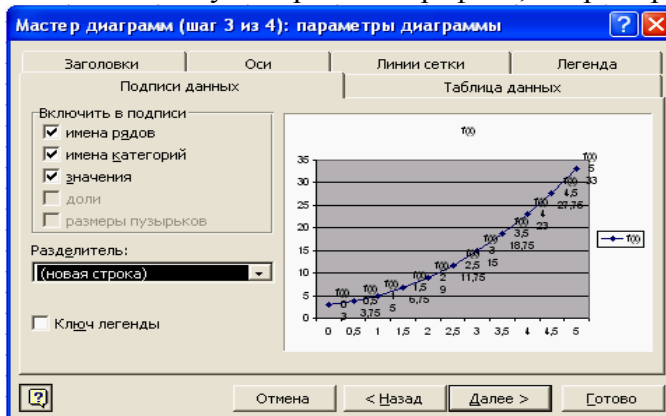


(Легендой в Excel называют прямоугольник, в котором сообщается, что нарисовано и какого цвета линия отрисовки, например: .)

На закладке «Подписи данных»: вы можете добавить имена рядов (в нашем случае ряд один – это $f(x)$), имена категорий (в данном случае значения x), значения самой функции, а также поставить различные величины между ними.



Заметим, что все эти величины пишутся прямо на графике, например:



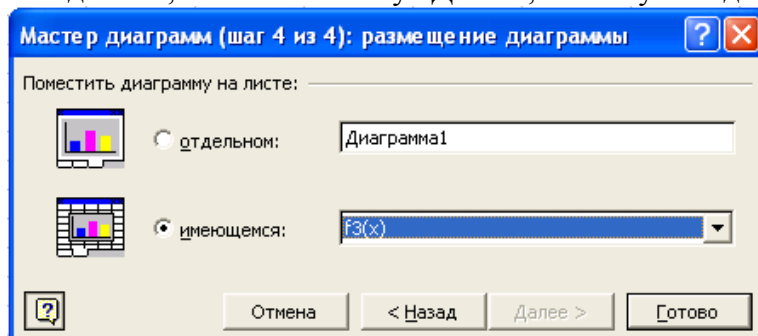
Закладка «Таблица данных»:



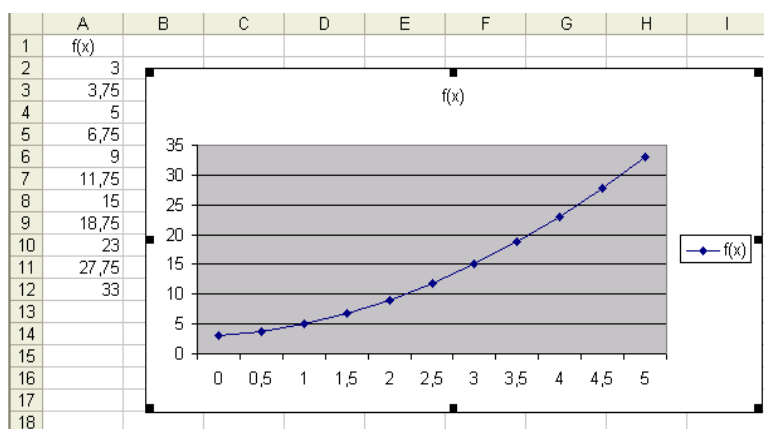
На ней вы можете включить таблицу данных в график, например:



Когда настройки все сделаны, нажмите кнопку «Далее», вы получите диалоговое окно:



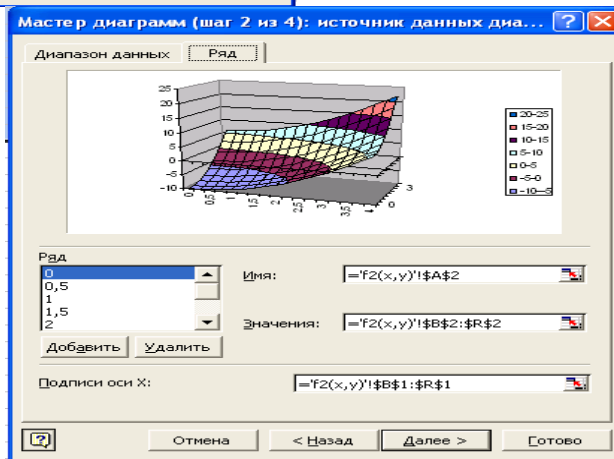
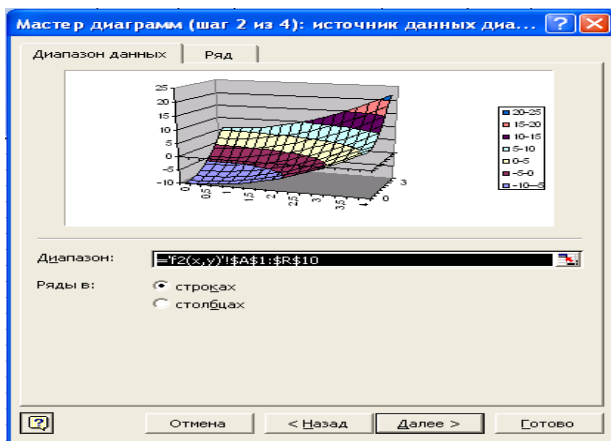
В этом окне вы можете выбрать положение графика (на отдельном листе или на том же самом, где и данные). Выберем имеющийся лист и нажмем кнопку «Готово», получим график вида:



5. Откроем лист **f2(x,y)**.

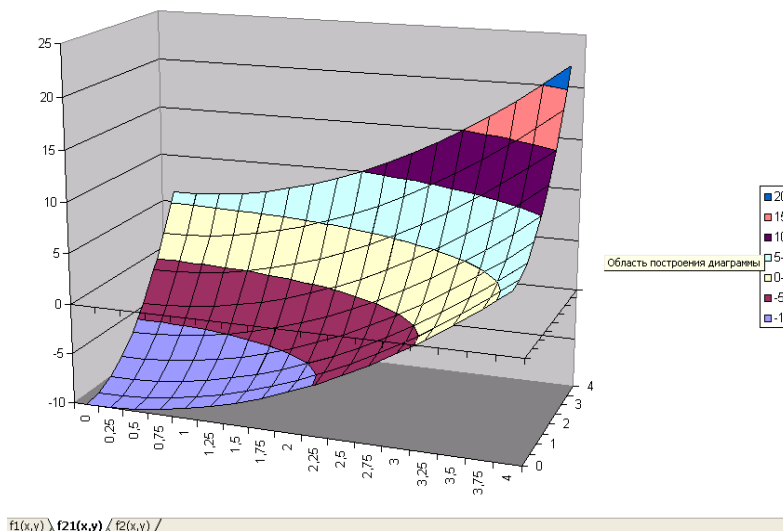
6. Для построения поверхности выделим ячейки от **A1** до **R10** и выберем **Вставка\Диаграмма...** В появившемся диалоговом окне, на закладке «Стандартные» в поле выбора «Тип» выберем «Поверхность», а в поле выбора «Вид» выберем вид поверхности (имеются три вида изображения поверхности: 1) классическое изображение поверхности с выделением уровней цветами; 2) изображение поверхности черными контурными линиями; 3) плоское изображение с выделением высот и впадин цветом и четкой отрисовкой линий уровня (напоминает школьный атлас по географии); 4) плоское изображение состоящие только из линий уровня(как в контурной карте)).

Когда вы выбрали нужный вид поверхности, нажмите кнопку «Далее», получите окно вида:



Обращаем ваше внимание на то, что при таком способе построения в этом окне заполнены все необходимые поля на обеих закладках. Такое возможно, только если ячейка **A1** пуста.

Нажимаем кнопку «Далее» и получаем диалоговое окно для настройки заголовков, осей, легенды и т.д. Опять нажимаем кнопку «Далее» и получаем окно для выбора размещения диаграммы. На этот раз выберем размещение на отдельном листе, в поле ввода вместо «Диаграмма 1» поместим другое название этого листа: **f21(x,y)** и нажмем кнопку «Готово». В результате получим:



Как видим, появился отдельный лист, на котором есть только данная поверхность.

2.13 Лабораторная работа №15 (2 часа).

Тема: «Табличный процессор Excel: операции с условием»

2.13.1 Цель работы: Научиться пользоваться математическими, некоторыми статистическими и логическими функциями в MS Excel

2.13.2 Задачи работы:

1. Вычисление значения заданной функции с условием
2. Применение математических, статистических и логических функций в вычислениях

2.13.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.13.4 Описание (ход) работы:

Задание:

1. Открыть Excel и созданный ранее документ. Создать новый лист и назвать его *if(x)*.
2. Вычислить значение заданной функции одной переменной *f1* с условием.
3. Вычислить количество точек функции, попадающих в заданный интервал.
4. Вычислить значения заданной функции одной переменной *f2*.
5. Вычислить сумму тех значений функции, аргументы которых лежат в заданном интервале.
6. Вычислить значение функции двух переменных.
7. Вычислить максимальное и минимальное значение функции.
8. Вычислить количество положительных и сумму отрицательных элементов функции.
9. Посчитать произведение тех значений функции, которые меньше 2.
10. Сохранить документ.

Пример выполнения задания:

Задание:

1.
$$f1(x) = \begin{cases} x+1, & x \leq 0 \\ x-1, & x > 0 \end{cases},$$

где x меняется от -2 до 2 с шагом $0,2$. Интервал: $[-0,2; 0,2]$.

2.
$$f2(x) = \begin{cases} x^2 + 1, & x \leq -2 \\ 1, & -2 < x < 2 \\ x - 3, & x \geq 2 \end{cases},$$

где x меняется от -4 до 4 с шагом $0,5$. Интервал $[-1; 1]$

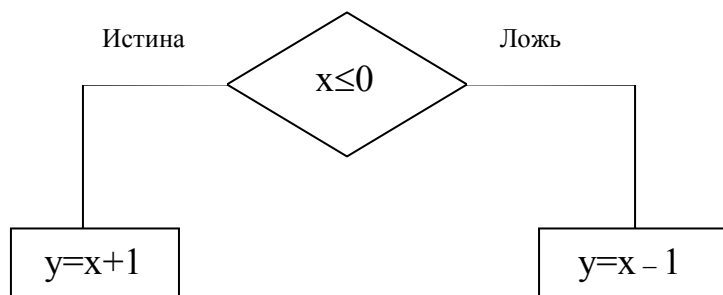
3.
$$f(x, y) = \begin{cases} x^2 + y^2 + 1, & \text{т. } (x, y) \text{ лежит в круге радиусом } 3 \\ x^2 + y^2 - 1, & \text{если т. } (x, y) \text{ лежит в круге с радиусом } 3, \end{cases}$$

где x меняется от -4 до 4 с в противном случае, $,25$.

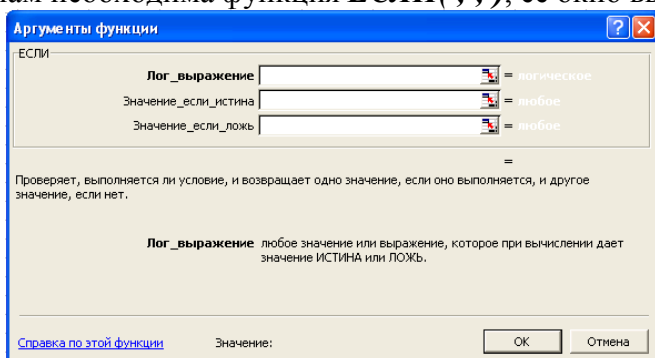
1. Откроем документ (**Файл\Открыть...**) и создадим новый лист (**Вставка\Лист**). Переименуем лист при помощи контекстного меню в *if(x)*.

2. Рассмотрим алгоритм вычисления данной функции.

Для этого построим блок-схему:



Для записи таких выражений в Excel есть встроенные логические функции. В данном случае нам необходима функция **ЕСЛИ(; ;)**, ее окно ввода выглядит следующим образом:




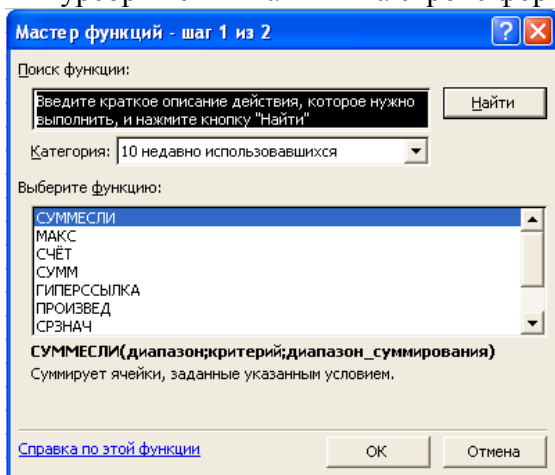
В первое поле ввода «Лог_выражение» вводится условие, записанное в нашей блок-схеме в ромбе. Во второе поле ввода «Значение_если_истина» вводится, то что присваивается в случае истинности условия. В третье поле ввода записывается оставшееся выражение, которое присваивается в случае невыполнения условия.

Данную функцию можно записать и без вызова ее окна, для этого записываем:
 =ЕСЛИ(условие;выр_истина;выр_ложь)

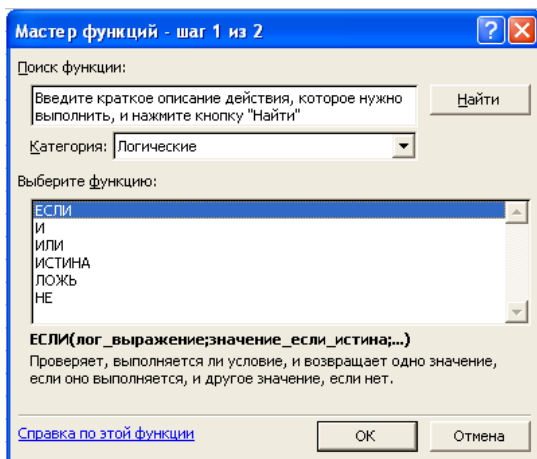
где в первую позицию записывается условие, как и в первое поле ввода, во вторую позицию записывается выражение, предназначенное для второго поля ввода, и, наконец, в третью позицию записываем оставшееся выражение.

Рассмотрим первый способ использования данной функции на примере 1.

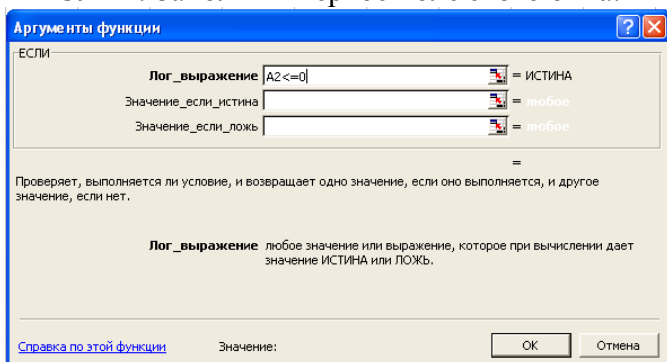
В ячейках **A1** и **B1** запишем заголовок таблицы: «x» и «f1(x)» соответственно. В ячейку **A2** введем -2, в **A3** -1,8, теперь выделим ячейки **A2** и **A3** и растянем до ячейки **A22**. Теперь установим курсор ячейки на **B2** и на строке формул нажмем кнопку , появится окно:



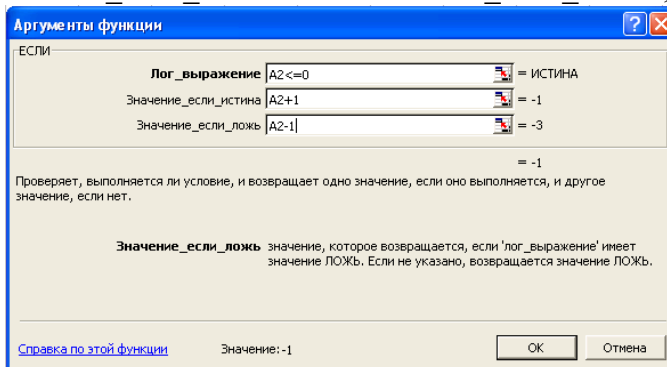
Развернем поле выбора «Категория» и выберем «Логические», получим:



В поле выбора «Выберите функцию:» выберем «ЕСЛИ» и нажмем «ОК». Получим окно функции «ЕСЛИ». Заполним первое поле этого окна:



Обращаем ваше внимание на то, что вместо x набрано **A2**, так как ячейке **B2**, содержащей значения функции, соответствует значение аргумента из ячейки **A2**. Далее, аналогично заполняя поля «Значение_если_истина» и «Значение_если_ложь», получим:



Теперь, нажав «ОК», мы получим следующий результат:

	B2		fx = ЕСЛИ(A2<=0;A2+1;A2-1)		
	A	B	C	D	E
1	x	f1(x)			
2	-2	-1			
3	-1,8				
4	-1,6				
5	-1,4				
6	-1,2				
7	-1				


Обратите внимание, как в строке формул заполнилась сама функция **ЕСЛИ()**.

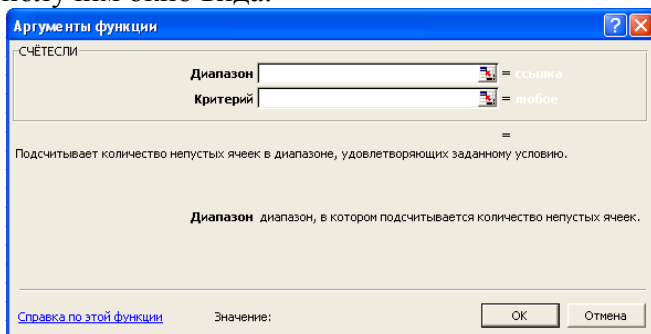
Осталось только растянуть формулу до ячейки **B22** и получить окончательный результат.

3. Нам осталось вычислить количество значений функции, попадающих в заданный интервал.

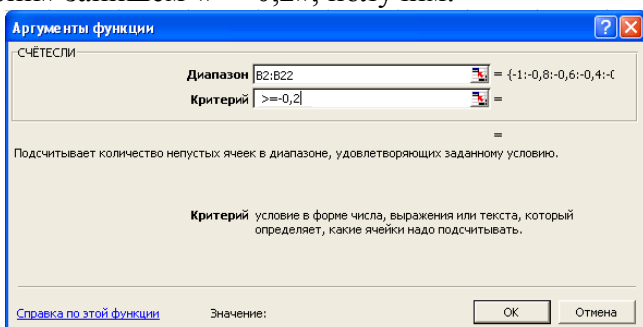
Для этого предназначена функция **СЧЁТЕСЛИ()**.

В ячейке **D2** произведем вычисления, для чего установим курсор ячейки на **D2** и нажмем

кнопку , выберем категорию «Статистические» и найдем функцию **СЧЁТ-ЕСЛИ()**, нажмем «ОК» и получим окно вида:



В поле «Диапазон» установим курсор и произведем выделение ячеек с **B2** до **B22**, а в поле «Критерий» запишем « $\geq -0,2$ », получим:



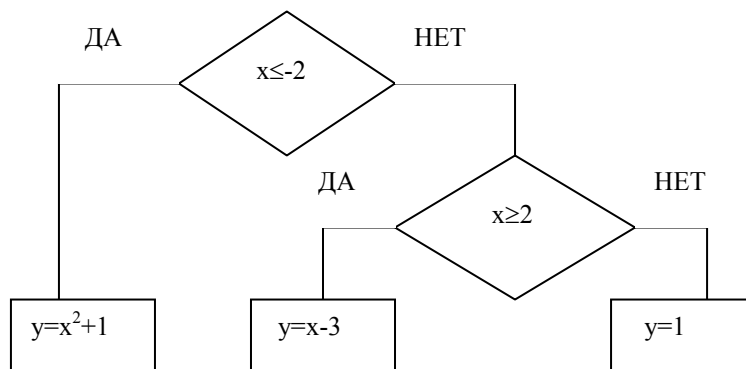
Теперь, нажав «ОК», мы получим количество значений функции, которые больше или равны $-0,2$, так как данная функция Excel позволяет находить количество не в интервале, а на заданной полуоси. Но нам необходимо найти количество значений функции, попадающих в заданный интервал. Как известно, вычисление количества в любом интервале можно представить как разность количеств на двух полуосях. Например, если нам необходимо вычислить количество элементов, попадающих в интервал $[a, b]$, представим его разностью между количеством на полуоси « $\geq a$ » и количеством с полуоси « $\geq b$ », в результате мы получим количество элементов, попадающих в заданный интервал. Применим это к нашей задаче. В ячейке **D2** мы уже вычислили количество элементов « $\geq -0,2$ », теперь в ячейку **D3** вычислим количество элементов « $\geq 0,2$ » и в ячейку **D4** запишем разность:

$$=D2-D3$$

Получим результат:

	D4		fx	=D2-D3
	A	B	C	D
1	x	f1(x)		
2	-2	-1		14
3	-1,8	-0,8		8
4	-1,6	-0,6		6
5	-1,4	-0,4		
6	-1,2	-0,2		
7	-1	0		
8	-0,8	0,2		

4. Теперь наберем более сложную функцию, но прежде нарисуем ее блок-схему:



Обратите внимание, что на одной оси условного оператора вложен еще один условный оператор. Теперь реализуем эту схему.

В ячейки **F1** и **G1** наберем заголовков для таблицы: **x** и **f2(x)** соответственно. В ячейки **F2–F18** занесем значения аргумента. В ячейку **G2** запишем:

=ЕСЛИ(F2<=-2;F2*F2+1;ЕСЛИ(F2>=2;F2-3;1))


Теперь, нажав кнопку  на строке формул, получим:

=ЕСЛИ(F2<=-2;F2*F2+1;ЕСЛИ(F2>=2;F2-3;1))

C	D	E	F	G
			x	f2(x)
	14		-4	17
	8		-3,5	
	6		-3	

Остается только растянуть формулу до ячейки **G18**.

5. Вычислим сумму тех значений функции, аргументы которых лежат в заданном интервале. Для этого есть функция **СУММЕСЛИ()**.

Нам необходимо вычислить сумму только тех элементов, аргументы которых лежат в заданном интервале. Для этого установим курсор ячейки на пустую ячейку **I2** и нажмем кнопку , в математических функций найдем нужную нам и нажмем «ОК». Получаем окно следующего вида:

Аргументы функции

СУММЕСЛИ

Диапазон: = ссылка

Критерий: = любое

Диапазон суммирования: = ссылка

Суммирует ячейки, заданные указанным условием.

Диапазон: диапазон проверяемых ячеек.

Справка по этой функции Значение:

В первое поле ввода – «Диапазон» – вводятся адреса тех ячеек, по которым будет происходить проверка условия, во второе поле – «Критерий» – вводится условие, накладываемое на заданный «Диапазон». В третье поле – «Диапазон суммирования» – вводятся адреса ячеек, данные в которых и будут суммироваться (заметим, что они зависят от ячеек, заданных в поле «Диапазон»). Напомним, что критерий суммирования ограничивает полупрямую, а не отрезок, поэтому поступим так же, как и в случае с вычислением количества. Для нашего интервала $[-1; 1]$, сначала вычислим сумму с критерием $\langle \geq -1 \rangle$, а затем $\langle > 1 \rangle$,

Для первой суммы получим формулу:



ИЗ	▼	$f_x = \text{СУММЕСЛИ}(F2:F18; ">1"; G2:G18)$
----	---	---

Теперь для ячейки **I4** запишем формулу:

нажмем **Enter** и получим окончательный результат.

Для этого вставим новый лист и назовем его *if(x,y)*. Ячейки **A2-A18** заполним значениями *if*, а ячейки **B1-R1** заполним значениями *y*. Теперь установим курсор в ячейку **B2** и запишем туда *y*:

Но растягивать такую формулу пока нельзя, так как будет происходить смещение ссылок. Для значений x и y . Закрепим для значений x имя столбца – **A**, а для значений y номер строки –

ЕСЛИ(\$A2*\$A2+B\$1*B\$1<=9;\$A2*\$A2+B\$1*B\$1+1;\$A2*\$A2+B\$1*B\$1-1)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		-2	-1.75	-1.5	-1.25	-1	-0.75	-0.5	-0.25	0	0.25	0.5	0.75	1	1.25	1.5	1.75	2
2	-4	19	18.063	17.25	16.563	16	15.563	15.25	15.0625	15	15.063	15.25	15.563	16	16.563	17.25	18.063	19
3	-3.5	15.25	14.313	13.5	12.813	12.25	11.813	11.5	11.3125	11.25	11.313	11.5	11.813	12.25	12.813	13.5	14.313	15.25
4	-3	12	11.063	10.25	9.5625	9	8.5625	8.25	8.0625	8	8.0625	8.25	8.5625	9	9.5625	10.25	11.063	12
5	-2.5	9.25	8.3125	7.5	6.8125	6.25	5.8125	5.5	5.3125	5.25	5.3125	5.5	5.8125	6.25	6.8125	7.5	8.3125	9.25
6	-2	9	8.0625	7.25	6.5625	6	5.5625	5.25	5.0625	5	5.0625	5.25	5.5625	6	6.5625	7.25	8.0625	9
7	-1.5	7.25	6.3125	5.5	4.8125	4.25	3.8125	3.5	3.3125	3.25	3.3125	3.5	3.8125	4.25	4.8125	5.5	6.3125	7.25
8	-1	6	5.0625	4.25	3.5625	3	2.5625	2.25	2.0625	2	2.0625	2.25	2.5625	3	3.5625	4.25	5.0625	6
9	-0.5	5.25	4.3125	3.5	2.8125	2.25	1.8125	1.5	1.3125	1.25	1.3125	1.5	1.8125	2.25	2.8125	3.5	4.3125	5.25
10	0	5	4.0625	3.25	2.5625	2	1.5625	1.25	1.0625	1	1.0625	1.25	1.5625	2	2.5625	3.25	4.0625	5
11	0.5	5.25	4.3125	3.5	2.8125	2.25	1.8125	1.5	1.3125	1.25	1.3125	1.5	1.8125	2.25	2.8125	3.5	4.3125	5.25
12	1	6	5.0625	4.25	3.5625	3	2.5625	2.25	2.0625	2	2.0625	2.25	2.5625	3	3.5625	4.25	5.0625	6
13	1.5	7.25	6.3125	5.5	4.8125	4.25	3.8125	3.5	3.3125	3.25	3.3125	3.5	3.8125	4.25	4.8125	5.5	6.3125	7.25
14	2	9	8.0625	7.25	6.5625	6	5.5625	5.25	5.0625	5	5.0625	5.25	5.5625	6	6.5625	7.25	8.0625	9
15	2.5	9.25	8.3125	7.5	6.8125	6.25	5.8125	5.5	5.3125	5.25	5.3125	5.5	5.8125	6.25	6.8125	7.5	8.3125	9.25
16	3	12	11.063	10.25	9.5625	9	8.5625	8.25	8.0625	8	8.0625	8.25	8.5625	9	9.5625	10.25	11.063	12
17	3.5	15.25	14.313	13.5	12.813	12.25	11.813	11.5	11.3125	11.25	11.313	11.5	11.813	12.25	12.813	13.5	14.313	15.25
18	4	19	18.063	17.25	16.563	16	15.563	15.25	15.0625	15	15.063	15.25	15.563	16	16.563	17.25	18.063	19

Для этого установим курсор ячейки на **B20** и в «Статистических» функциях выберем **МАКС()**, а в первое поле ввода запишем: «B2:R18». Аналогично в ячейку **B21** вычислим минимум при помощи **МИН()**. Получим:

	B21		f _x	=МИН(B2:R18)
	A	B	C	D
19				
20		19		
21		1		
22				

8. Теперь вычислим количество положительных и сумму отрицательных элементов функции.

Для этого установим курсор ячейки на **C20** и в строку формул запишем:

=СЧЁТЕСЛИ(B2:R18;">0")

а в ячейку **C21**:

=СУММЕСЛИ(B2:R18;"<0")

Как видим, в случае, когда суммируются и проверяются на критерий одни и те же данные, писать их дважды (до и после «критерия») не надо.

Получаем:

	C21								
	A	B	C	D	E	F	G		
19									
20		19	289						
21		1	0						
22									

9. Теперь вычислим произведение тех значений функции, которые меньше 2.

Для этого установим курсор в ячейку **E21** и в строке формул запишем:

=ПРОИЗВЕД(ЕСЛИ(B2:R18<2;B2:R18))

нажмем **Ctrl+Shift+Enter**, получим результат:

	E21								
	A	B	C	D	E	F	G	H	
19									
20		19	289						
21		1	0		1091				
22									

Обратите внимание на то, как выглядит запись в строке формул: в результате нажатия вышеуказанной комбинации клавиш Excel заключил всю формулу в фигурные скобки.

2.14 Лабораторная работа №16 (2 часа).

Тема: «Табличный процессор Excel: работа с массивами»

2.14.1 Цель работы: Научиться работать с массивами.

2.14.2 Задачи работы:

1. Находить скалярное произведение векторов.
2. Вычислять определители матриц, произведения матриц и матрицы на вектор.
3. Находить решение системы линейных уравнений

2.14.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.14.4 Описание (ход) работы:

Задание:

1. По заданным координатам точек A , B , C , D найти координаты векторов $a=AB$ и $b=CD$.
2. Вычислить скалярное произведение найденных векторов.
3. Найти следующие произведения векторов на заданную матрицу M : $a*M$ и $M*b$.
4. Вычислить определители матриц M и S .
5. Найти обратные матрицы S^{-1} и M^{-1} .
6. Вычислить произведение матрицы S на обратную к ней S^{-1} .

7. Найти решение системы линейных уравнений $Sx=b$ и $Mx=a$.
8. Выполнить проверку для найденных решений.
9. Сохранить документ.

Пример выполнения задания:

Задание:

$A=(2, -1, 0)$; $B=(-1, 4, -6)$; $C=(1, -3, -5)$; $D=(-2, -1, 0)$;

$$S = \begin{pmatrix} 1 & -1 & 8 \\ 3 & 3 & -5 \\ -2 & -7 & 11 \end{pmatrix}, \quad M = \begin{pmatrix} -1 & 4 & 13 \\ 0 & 7 & -3 \\ 7 & -6 & 17 \end{pmatrix}.$$

1. Для того чтобы найти координаты векторов, заданных координатами точек начала и конца вектора, занесем координаты этих точек в Excel.

Для этого создадим новый лист и назовем его «**массивы**». В ячейку **A2** запишем «A», в ячейки **B1:B3** заполним значения координат точки **A**. В ячейку **D2** запишем «B», в ячейки **E1:E3** заполним значения координат точки **B**. Аналогично для точки **C** заполняем ячейки: **A6** и **B5:B7**, для точки **D** – **D6**, **E5:E7**. Получаем:

	A	B	C	D	E
1		2			-1
2	A	-1		B	4
3		0			-6
4					
5		1			-2
6	C	-3		D	-1
7		-5			0
8					

Примечание. Напомним, что для вычисления координат вектора, заданного координатами начала и конца, необходимо из координат конца вектора вычесть координаты его начала.

В ячейку **A10** запишем «a», теперь выделим ячейки **B9:B11** и в строку формул запишем:
 $=E1:E3-B1:B3$

после чего нажмем **Ctrl+Shift+Enter**.

Примечание. Напомним, что при таких комбинациях клавиш следует нажать и удерживать клавиши **Ctrl+Shift** одной рукой, а второй рукой нажать клавишу **Enter**. **Настоятельно рекомендуем производить данную комбинацию двумя руками, а не одной.** Если не получилось, установите текстовый курсор в строку формул и попробуйте нажать вышеуказанную комбинацию клавиш еще раз.

Если вы правильно нажали клавиши, то получите результат, а в строке формул увидите следующую запись:

$\{=E1:E3-B1:B3\}$

Примечание. Отметим, что исправлять, удалять и переносить такие формулы можно только всем блоком. Для этого выделяем все (в нашем случае три) ячейки и удаляем. Если нужно формулу исправить, опять выделяем все ячейки и исправляем в строке формул, после чего нажимаем **Ctrl+Shift+Enter**.

Для вычисления вектора **b** в ячейку **D10** набираем «b», далее выделяем ячейки **E9:E11** и в строке формул набираем:

$=E5:E7-B5:B7$

нажимаем **Ctrl+Shift+Enter** и получаем:

	A	B	C	D	E
8					
9		-1			-1
10	a	-2		b	-5
11		-5			6

2. Вычислим скалярное произведение $a*b$ и $b*a$.

Для этого в ячейку **A15** наберем « $a*b=$ », а в ячейку **B15** наберем формулу:
=МУМНОЖ(ТРАНСП(B9:B11);E9:E11)
и нажмем **Ctrl+Sift+Enter**.

Отметим, что все массивы в Excel умножаются при помощи функции **МУМНОЖ()**, по принципу умножения матриц, т.е. строка на столбец. Поэтому если вы умножаете два вектора, то для правильного результата, необходимо умножать вектор-строку на вектор-столбец. Преобразовать же вектор-столбец в вектор-строку позволяет функция **ТРАНСП()**.

Для получения скалярного произведения $b*a$ в ячейку **D15** наберем « $b*a=$ », а в ячейку **E15**:

=МУМНОЖ(ТРАНСП(E9:E11);B9:B11)
нажмем **Ctrl+Sift+Enter**.

Получим результат:

	A	B	C	D	E	F
14						
15	$a*b=$	-19		$b*a=$	-19	
16						

3. Научимся вычислять произведение вектора на матрицу.

Для этого заполним заданные матрицы. В ячейку **G2** запишем «M», а в ячейки **H1:J3** заполним значения матрицы. Аналогично заполним значения матрицы **S** в ячейки **G6** и **H5:J7**. Получим:

	G	H	I	J
1		1	-1	8
2	M	3	3	-5
3		-2	-7	11
4				
5		-1	4	13
6	S	0	7	-3
7		7	-6	17

Теперь в ячейку **L2** заполним « $a*M=$ », выделим ячейки **M2:O2** и в строке формул запишем: =МУМНОЖ(ТРАНСП(B9:B11);H1:J3)
нажмем **Ctrl+Sift+Enter**.

Заметим, что для того чтобы умножить матрицу на вектор слева, необходимо, чтобы вектор был записан в виде вектора-строки, для этого мы и применяем функцию транспонирования. Отметим, что из алгебры вам должно быть известно, что результат представляет собой вектор-строку, что мы и получили.

Для умножения матрицы на вектор справа вектор должен иметь вид вектора-столбца; результат также имеет вид вектора-столбца.

В ячейку **L6** запишем « $M*b=$ », выделим ячейки **M5:M7** и в строку формул наберем:
=МУМНОЖ(H1:J3;E9:E11)
нажмем **Ctrl+Sift+Enter**.

В результате получим:

	L	M	N	O
1				
2	$a*M=$	3	30	-53
3				
4				
5		52		
6	$M*b=$	-48		
7		103		

4. Вычислим определители заданных матриц.

Для этого в ячейку **L10** наберем « $|M|=$ », а в ячейку **M10** формулу:
=МОПРЕД(H1:J3)

и нажмем **Ctrl+Sift+Enter**.

Теперь в ячейку **L14** наберем « $|S|$ =», а в ячейку **M14** формулу:
=МОПРЕД(H5:J7)

и нажмем **Ctrl+Sift+Enter**.

Получим:

	L	M	C
10	$ M $ =	-99	
11			
12			
13			
14	$ S $ =	-822	
15			

5. Так как определители наших матриц не равны нулю, то мы можем найти обратные матрицы M^{-1} и S^{-1} .

Для этого в ячейку **G10** запишем « $1/M$ », выделим ячейки **H9:J11** и в строку формул поместим:

=МОБР(H1:J3)

нажмем **Ctrl+Sift+Enter**.

В ячейку **G14** заполним « $1/S$ », выделим ячейки **H13:J15**, в строку формул запишем:

=МОБР(H5:J7)

и нажмем **Ctrl+Sift+Enter**.

Мы получим две обратные матрицы вида:

	G	H	I	J
8				
9		0,02020202	0,454545	0,191919192
10	$1/M$	0,232323232	-0,272727	-0,292929293
11		0,151515152	-0,090909	-0,060606061
12				
13		-0,122871046	0,177616	0,125304136
14	$1/S$	0,025547445	0,131387	0,003649635
15		0,059610706	-0,026764	0,008515815

6. Для проверки правильности вычисления обратной матрицы найдем произведение матрицы M на её обратную – если вычисления были произведены правильно, то мы получим единичную матрицу.

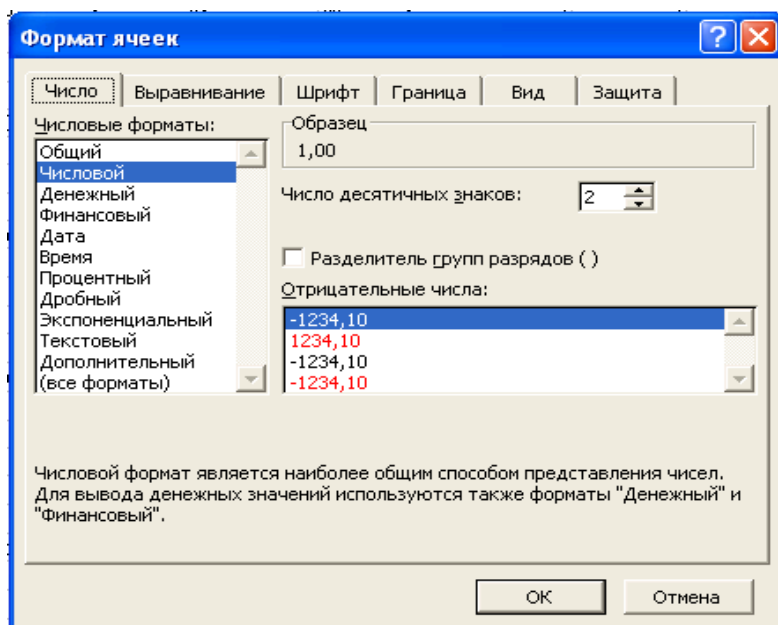
Для этого в ячейку **G18** заполним « $M*1/M$ », выделим ячейки **H17:J19** и в строке формул наберем: =МУМНОЖ(H1:J3;H9:J11)

нажмем **Ctrl+Sift+Enter**.

Аналогично для матрицы S заполним ячейки **G22** и **H21:J23**, получим:

	G	H	I	J
16				
17		1	0	-5,55112E-17
18	$M*1/M$ =	0	1	0
19		-2,22045E-16	2,22E-16	1
20				
21		1	5,55E-17	-4,16334E-17
22	$S*1/S$ =	0	1	0
23		0	1,11E-16	1

Мы видим, что в некоторых ячейках, где ожидали получить нули, мы получили некие значения с записью E-16 и E-17. Такая запись означает, что число умножается на 10^{-16} или 10^{-17} , что можно считать числом близким к нулю. Как правило, в прикладных задачах значимыми являются лишь 2–3 знака после запятой, поэтому с точностью до двух знаков мы получили нули. Чтобы запись приобрела нормальный вид, зададим для данных ячеек числовой формат с двумя позициями после запятой. Для чего выделим ячейки **H8:J14** и выберем **\Формат\Ячейки**, получим окно:



На закладке «Число» в поле выбора «Числовые форматы» выберем «Числовой», в поле «Число десятичных знаков» установим «2» и нажмем «ОК». В результате получим:

	G	H	I	J
16				
17		1,00	0,00	0,00
18	M*1/M=	0,00	1,00	0,00
19		0,00	0,00	1,00
20				
21		1,00	0,00	0,00
22	S*1/S=	0,00	1,00	0,00
23		0,00	0,00	1,00

Как мы убедились, обратные матрицы вычислены верно, теперь можно их использовать для дальнейших вычислений.

7. Найдем решение систем линейных уравнений $M \cdot x_1 = a$ и $S \cdot x_2 = b$.

Сначала решим первую систему, для этого в ячейку **A21** заполним «x1=», выделим ячейки **B20:B22** и в строку формул запишем:

=МУМНОЖ(Н9:J11;B9:B11)

нажмем **Ctrl+Sift+Enter**.

Решение второй системы получим в ячейки **E20:E22**, в результате имеем:

	A	B	C	D	E
19					
20		-1,8889			-0,01338
21	x1=	1,77778		x2=	-0,66058
22		0,33333			0,125304

8. Выполним проверку полученных решений.

Для этого вычислим следующие значения: $|M \cdot x_1 - a|$ и $|S \cdot x_2 - b|$. В ячейку **A26** заполним « $|M \cdot x_1 - a|$ =», выделим ячейки **B25:B27** и в строку формул запишем:

=ABS(МУМНОЖ(Н1:J3;B20:B22)-B9:B11)

нажмем **Ctrl+Sift+Enter**.

Аналогично, для проверки решения второй системы выделим ячейки **E25:E27** и в строку формул запишем:

=ABS(МУМНОЖ(Н5:J7;E20:E22)-E9:E11)

нажмем **Ctrl+Sift+Enter**.

В результате получим:

	A	B	C	D	E	F	G	H	I	J	K
1	Объем продукции за 3 квартал 2005г.										
2											
3	Код предприятия	0251	0253	0234	0245	0243	0236	0238	0254	0261	0263
4	Итого(в тыс.р.)	43678	45657	12323	45776	23435	65788	34522	78534	35567	12334
5											
Лист1 / Лист2 / Лист3 / Лист4 / Лист5											
	A	B	C	D	E	F	G	H	I	J	K
1	Объем продукции за 4 квартал 2005г.										
2											
3	Код предприятия	0251	0253	0234	0245	0243	0236	0238	0254	0261	0263
4	Итого(в тыс.р.)	52312	45320	65341	67345	87234	23786	45987	37856	76448	75648
5											
Лист1 / Лист2 / Лист3 / Лист4 / Лист5											
	A	B	C	D	E	F	G	H	I	J	K
1	Объем продукции за 2005 год										
2											
3	Код предприятия	0251	0253	0234	0245	0243	0236	0238	0254	0261	0263
4	За 2005 год										
5											
Лист1 / Лист2 / Лист3 / Лист4 / Лист5											

Как вы уже поняли, на пятый лист будет производиться консолидация данных. Установите курсор в ячейку **B4 Лист5** и наберите туда
=СУММ()

Установите текстовый курсор между круглыми скобками и откройте **Лист1**, на нем выберите ячейку **B4**, после этого нажмите клавишу **Shift** и, удерживая ее, щелкните левой кнопкой мышки по наименованию **Лист4**. Получим:

B4	=СУММ(Лист1:Лист4!B4)										
	A	B	C	D	E	F	G	H	I	J	K
1	Объем продукции за 2005 год										
2											
3	Код предприятия	0251	0253	0234	0245	0243	0236	0238	0254	0261	0263
4	За 2005 год	235999									
5											
Лист1 / Лист2 / Лист3 / Лист4 / Лист5											

Теперь растянем формулу:

	A	B	C	D	E	F	G	H	I	J	K
2											
3	Код предприятия	0251	0253	0234	0245	0243	0236	0238	0254	0261	0263
4	За 2005 год	235999	200971	198852	212854	221665	233660	244686	190190	209791	169106
5											
Лист1 / Лист2 / Лист3 / Лист4 / Лист5											

2. Рассмотрим создание именованных диапазонов.

Задав имя диапазона ячеек, вы можете далее использовать его название вместо стандартного адреса диапазона.

Чтобы присвоить диапазону имя, нужно выделить ячейки, объединенные в диапазон, и выполнить: **Вставка\Имя\Присвоить...**

получим диалоговое окно вида:

	A	B	C	D	E	F	G	H	I
1									
2		2001							
3	ПГС	3,2							
4	ГСХ	3,1							
5	ПЗ	2,4							
6	ЗУН	3,4							
7	МАС	4,2							
8	ПСМИК	4,5							
9	ГТС	3,5							
10	ТГив	3,6							
11	ВВ	4,4							
12									
13									
14									
15									

Присвоение имени

Имя: АСФ

Формула: =Лист1!\$B\$3:\$B\$6

ОК

Заккрыть

Добавить

Удалить

В поле «Имя» введите имя диапазона. Теперь посчитаем средний конкурс на факультете, для этого установим курсор в ячейку **B13** и наберем формулу:

=СРЗНАЧА(АСФ)

Получим:

B13		fx = СРЗНАЧА(АСФ)			
	A	B	C	D	E
1					Конкурс
2		2001	2002	2003	2004
3	ПГС	3,2	3,4	3,2	
4	ГСХ	3,1	3,1	3,4	
5	ПЗ	2,4	2,7	2,9	
6	ЭУН	3,4	3,5	3,4	
7	МАС	4,2	4,1	3,9	
8	ПСМИК	4,5	4,3	4,2	
9	ГТС	3,5	3,6	3,4	
10	ТГиВ	3,6	3,5	3,3	
11	ВВ	4,4	4,5	4,3	
12					
13		3,025			
14					

3. Поиск решения с помощью подбора параметра.

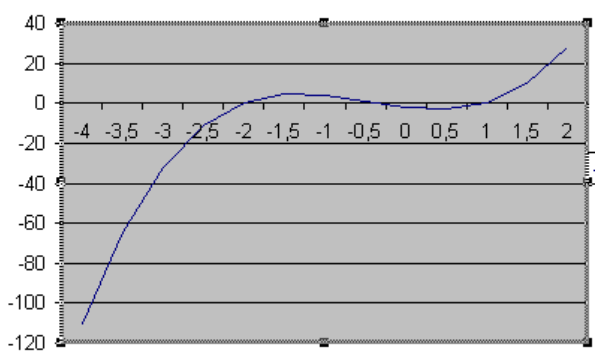
Данная задача возникает в тех случаях, когда результат некоего действия, выраженного математической формулой, зависит от какого-либо параметра. Необходимо бывает произвести подбор этого параметра так, чтобы результат действия был искомым.

Рассмотрим данную задачу на примере поиска корней уравнения третьей степени: $3x^3 + 4x^2 - 5x - 2$. Пусть параметром подбора будет x , а искомым результатом 0 .

Для начала построим таблицу функции:

	A	B
1	x	P(x)
2	-4	-110
3	-3,5	-64,125
4	-3	-32
5	-2,5	-11,375
6	-2	0
7	-1,5	4,375
8	-1	4
9	-0,5	1,125
10	0	-2
11	0,5	-3,125
12	1	0
13	1,5	9,625
14	2	28

и ее график:



Из графика видно, что корни лежат в интервалах: $[-2,5; -1,5]$; $[-0,5; 0]$; $[0,5; 1,5]$.

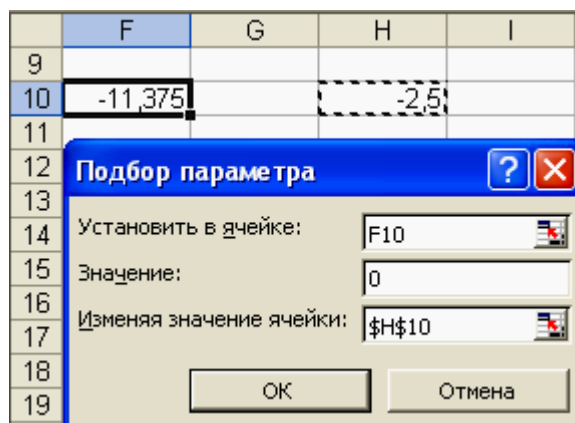
Теперь в ячейку **H10** запишем приблизительное значение параметра из первого интервала, а именно «-2,5», а в ячейку **F10** запишем нашу формулу относительно заданного параметра. Получим:

	F	G	H
9			
10	-11,375		-2,5
11			

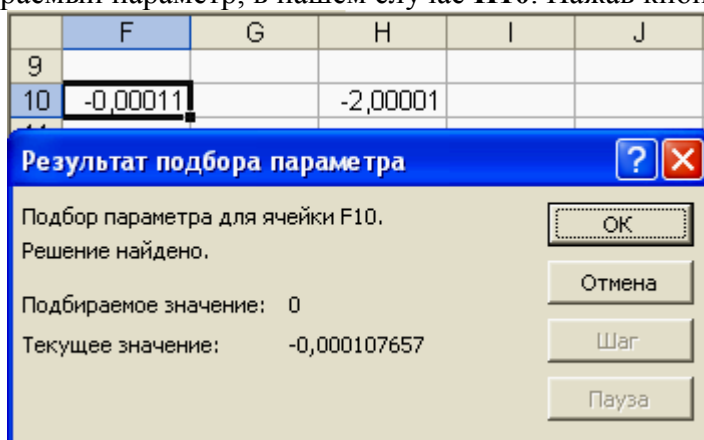
Установив курсор на ячейку **F10**, выполним:

\Сервис\Подбор параметра...

Получим диалоговое окно вида:

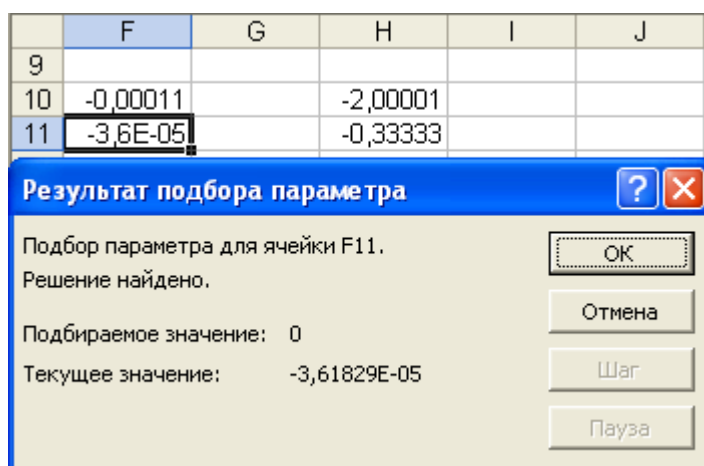


В первом поле ввода «Установить в ячейке:» значение уже заполнено. Во второе поле ввода «Значение:» устанавливается значение которое необходимо получить в результате подбора параметра, в нашем случае **0**. В третьем поле «Изменяя значение ячейки:» записывается адрес ячейки, где хранится подбираемый параметр, в нашем случае **H10**. Нажав кнопку «ОК», получим:

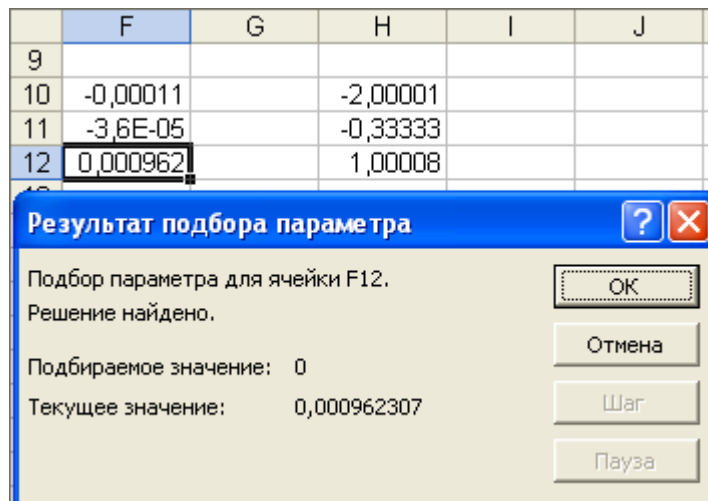


Как вы можете видеть, корни вычисляются с некоторой точностью. В ячейке **F10** получается результат нашего выражения при подстановке подобранного параметра, а в ячейке **H10** – подобранный параметр. В нашем случае мы подбирали корень из первого интервала и он равен -2.

Аналогично в ячейках **F11** и **H11** вычислим второй корень, при начальном значении параметра 0 получим:



Для третьего корня начальное значение параметра равно 1,5, имеем:



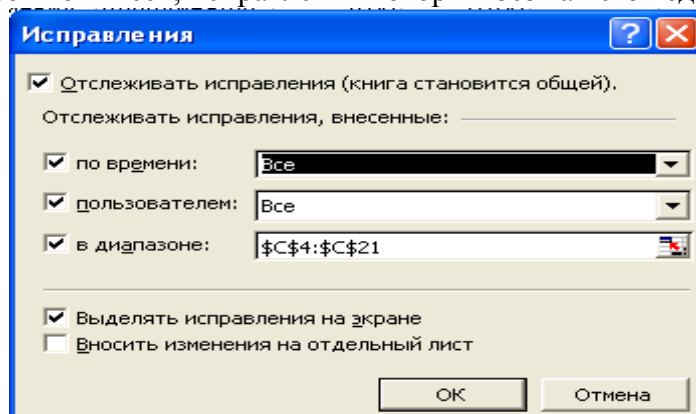
Как можно увидеть, наше уравнение имеет три корня: -2, -1/3, 1.

4. Выделение изменений, внесенных в книгу.

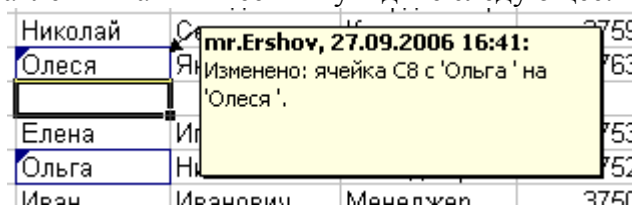
С помощью выделения изменений вы можете отслеживать все последующие исправления вносимые в файл. Для того чтобы задать данную команду, выполните:

\Сервис\Исправления\Выделить исправления...

В диалоговом окне «Исправления» включите нужные опции, а в поле ввода «в диапазоне:» укажите диапазон тех ячеек, исправления которых без вашего ведома нежелательны:



В результате при исправлении таких ячеек вы увидите следующее:



Как вы можете видеть, ячейки, в которых были произведены изменения, выделяются синей рамкой с закрашенным уголком в верхней левой ее части.

Если исправления вас не устраивают, то вы можете от них отказаться при помощи команды:

\Сервис\Исправления\Принять/отклонить исправления...

5. Вставка примечаний.

Примечания (комментарии) отображаются на экране при наведении указателя мыши на ячейку.

Для установки примечания выделите ячейку, к которой это примечание пишется, и выполните:

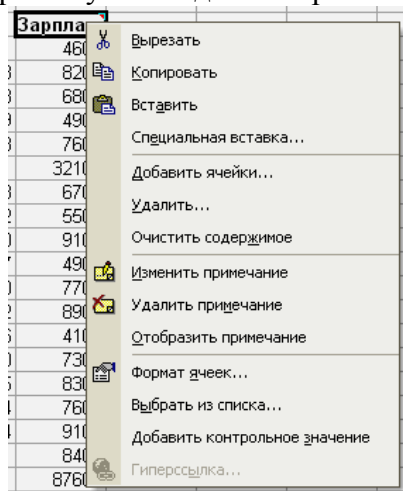
\Вставка\Примечание

Вы увидите следующий результат:

		mr.Ershov:
Зарплата		
4600		
8200		
6800		
4900		

Вам остается заполнить примечание. Как вы можете заметить, у ячеек с примечанием закрашен красным цветом верхний правый угол. Примечания появляются при наведении курсора мыши на такую ячейку.

Если необходимо снять примечание, то вызовите контекстное меню на данной ячейке (щелчок правой кнопки мыши) и выберите пункт «Удалить примечание»:



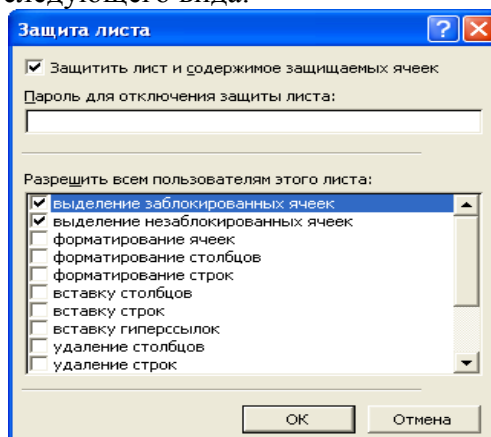
Там же вы можете найти пункт меню, позволяющий исправлять примечания, а также их отображать.

6. Бывает необходимо запретить исправлять информацию в документе Excel или в некоторых его ячейках.

Для ограничения доступа к книге выполните команду:

\Сервис\Защита\Защитить лист...

Появится диалоговое окно следующего вида:



Здесь вы можете указать разрешенные изменения для других пользователей. Для наибольшей эффективности защиты рекомендуем вам задать пароль для отключения защиты.

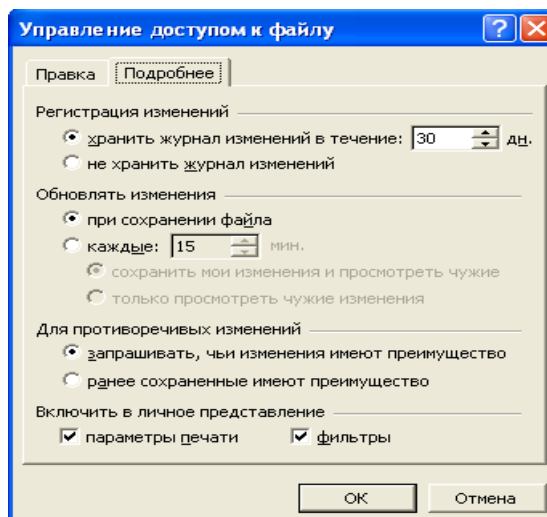
Если вы хотите запретить другим пользователям работать с книгой, то вы полните команду:

\Сервис\Доступ к книге...

и на закладке «Правка» уберите флажок с пункта «Разрешить совместный доступ».

Если же вы собираетесь разрешить другим пользователям вносить исправления в свою книгу, то флажок на пункте «Разрешить совместный доступ» следует оставить, а также выполнить

настройки на закладке «Подробнее»:

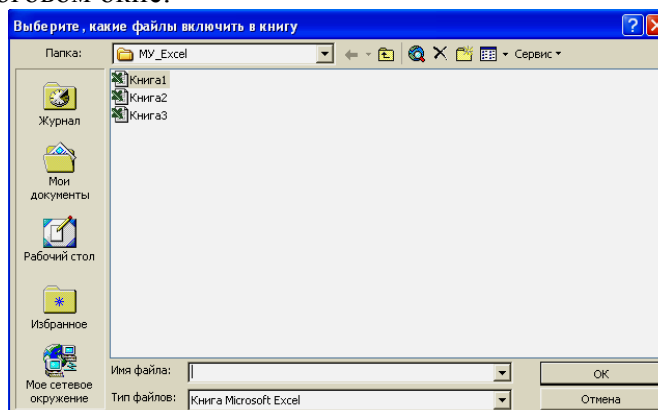


7. Иногда целесообразно создать несколько копий книги, к которой разрешен общий доступ, и затем регулярно отслеживать исправления в каждой из них. Проанализировав внесенные изменения, вы можете объединить эти копии.

Для слияния документов выполните команду:

\\Сервис\Сравнить и объединить книги

в появившемся диалоговом окне:



выберите файлы, которые будут объединены с текущим.

2.16 Лабораторная работа №18(2 часа).

Тема: «Проектирование базы данных в СУБД MS Access»

2.16.1 Цель работы: Изучение информационной технологии создания пустой базы данных ручным способом и с помощью шаблонов средствами мастера в системе управления базами данных (СУБД) Microsoft Access. Изучение объектов учебной базы «Борей».

2.16.2 Задачи работы:

1. Создание пустой базы данных.
2. Создание пустой базы данных с помощью шаблонов средствами мастера.

2.16.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.16.4 Описание (ход) работы:

Задание 1. Создание пустой базы данных.

Порядок работы

1. Запустите программу СУБД Microsoft Access. Для этого при стандартной установке MS Office выполните: *Пуск/Программы/ Microsoft Access*. В открывшемся окне *Microsoft Access*, предназначенном для открытия или выбора базы данных (БД), нажмите кнопку *Отмена*.

2. Изучите интерфейс программы, подводя мышь к различным элементам экрана.

3. Выберите команду *Файл/Создать*. На экране откроется окно диалога *Создание*, содержащее две вкладки – *Общие* и *Базы данных* (рис. 22.1):

вкладка *Общие* предназначена для создания новой пустой базы данных;

вкладка *Базы данных* позволяет создать базу данных с помощью мастера и выбрать образец, содержащий большинство необходимых по определенной тематике объектов базы данных.

4. Перейдите на вкладку *Общие* и нажмите кнопку *ОК* в нижней части окна диалога. На экране откроется окно диалога *Файл новой базы данных* (рис. 22.2).

5. Из раскрывающегося списка «Папка» выберите папку «Мои документы», в которой вы будете сохранять базу данных, а в поле *Имя файла* введите имя базы данных «Моя пустая база данных» (в имя базы можно ввести свою фамилию). Расширение для имени файла (mdb) можно не указывать, поскольку

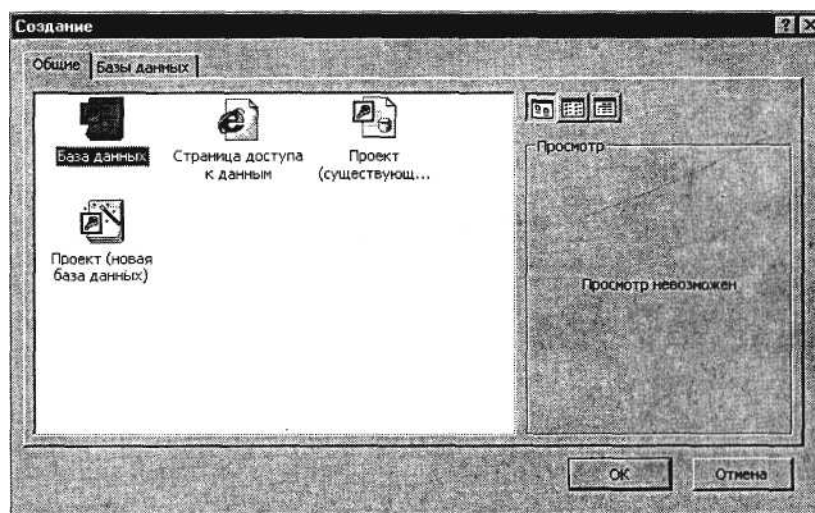


Рис. 1. Окно диалога *Создание*

по умолчанию в поле ввода *Тип файла* установлен тип «База данных Microsoft Access».

6. После ввода имени создаваемой базы данных нажмите кнопку *Создать*. На экране откроется окно *Базы данных* (рис. 22.3). Изучите интерфейс окна базы данных.

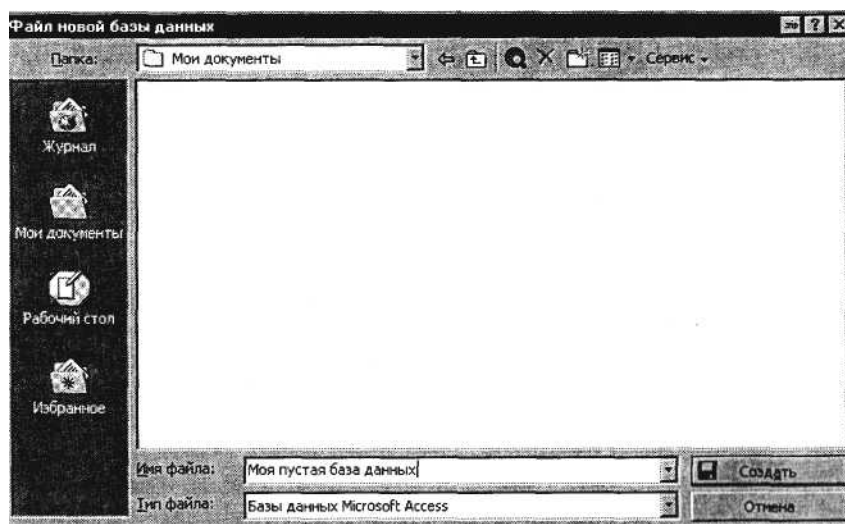


Рис. 2. Окно диалога *Файл новой базы данных*



Рис. 3. Окно новой базы данных

К р а т к а я с п р а в к а . Основу базы данных составляют хранящиеся в ней данные. Однако в базе данных есть и другие важные компоненты, которые принято называть объектами. Каждому объекту соответствует своя вкладка. Ярлыки объектов расположены в левой части окна базы данных. В окне базы данных создаются все объекты базы, перечень которых приведен ниже:

Таблицы – содержат данные;

Запросы – позволяют задавать условия для отбора данных и вносить изменения в данные;

Формы – служат для ввода, просмотра и редактирования информации;

Страницы – файлы в формате HTML, позволяющие просматривать данные с помощью браузера Internet Explorer;

Отчеты – позволяют обобщать и распечатывать информацию;

Макросы – выполняют одну или несколько операций автоматически;

Модули – программа автоматизации и настройки функций базы данных, написанных на языке VB (Visual Basic).

7. Познакомьтесь со свойствами вашей базы данных командой *Файл/Свойства базы данных* (рис. 22.4). Определите размер созданной БД.
8. Закройте созданную вами пустую базу данных.

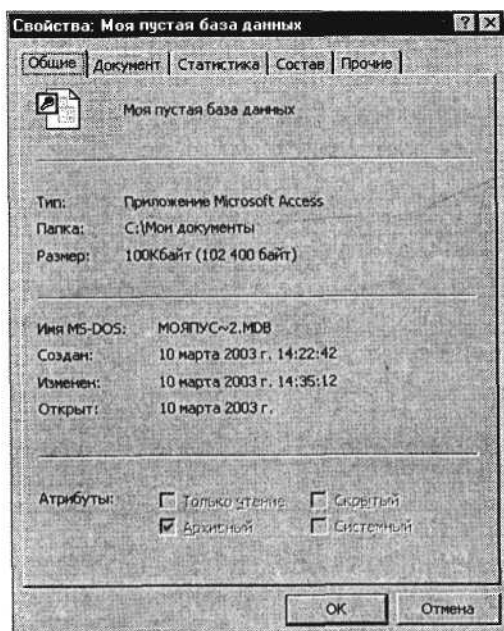


Рис.4. Окно свойств базы данных

Задание 2. Создание пустой базы данных с помощью шаблонов средствами мастера.

Порядок работы

1. Выберите команду *Файл/Создать* или нажмите клавиши [Ctrl]-[N]. На экране откроется окно диалога *Создание*, содержащее две вкладки.
2. Перейдите на вкладку *Базы данных*. На экране появится список баз данных (шаблонов), предлагаемых мастером (рис. 22.5).
3. Выберите из списка образец базы данных «Контакты» и запустите на выполнение мастера создания базы данных нажатием кнопки *ОК*.
4. Выберите из раскрывающегося списка «Папка» папку «Мои документы», в которой вы будете сохранять базу данных, а в поле *Имя файла* введите имя базы данных «Мои контакты», затем нажмите кнопку *Создать*.
5. В следующем окне диалога мастер сообщает, какую информацию будет содержать создаваемая база данных (рис. 22.6). В нижней части окна находятся следующие кнопки:
Отмена – прекращает работу мастера; *Назад* – позволяет вернуться к предыдущему шагу в работе мастера;

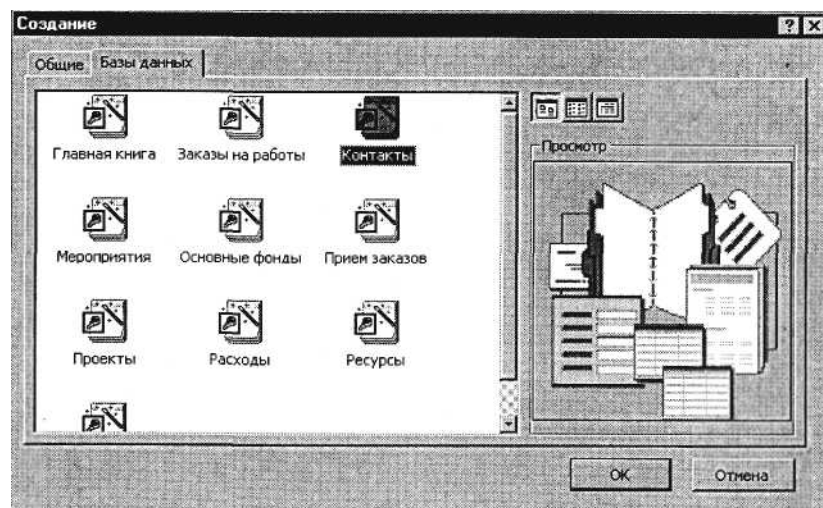


Рис.5. Вкладка *Базы данных*

Далее – позволяет перейти к следующему шагу в работе мастера;

Готово – запускает мастера на создание базы с установленными параметрами.

Для продолжения работы нажмите кнопку *Далее*. 6. Открывшееся окно диалога (рис. 22.7) содержит два списка. Первый из них – список таблиц базы данных, второй – список полей выбранной таблицы. Обычно отмечены в списке

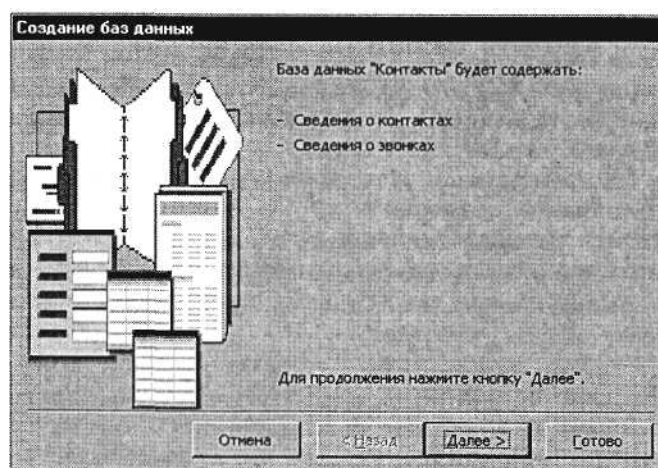


Рис. 6. Окно диалога с информацией о создаваемой базе

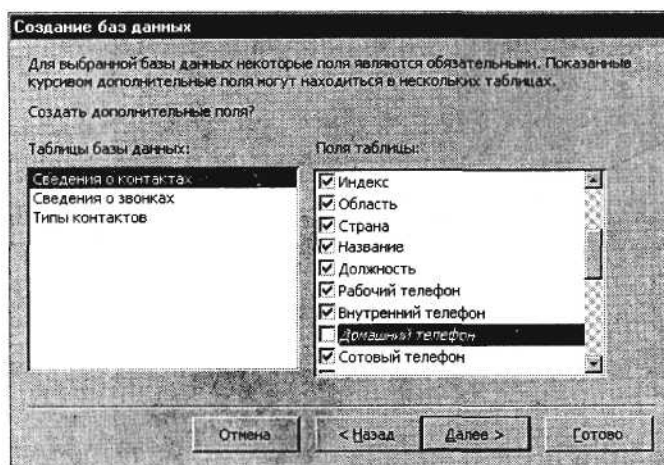


Рис. 7. Выбор списка полей таблиц базы данных

поля, которые будут включены в таблицу, но можно включить и дополнительные поля, отмечая их в списке. Для перехода к следующему окну мастера нажмите кнопку *Далее*.

7. В следующих окнах выберите вид оформления экрана, вид создаваемых отчетов, заголовков и рисунок, которые будут появляться во всех отчетах.

8. После нажатия в последнем окне кнопки *Готово* мастер переходит к созданию базы данных, состоящей из таблиц с заданными вами полями, форм ввода и просмотра информации, а также отчетов. После завершения процесса создания базы данных вы сразу же можете пользоваться готовой базой данных: вводить данные в таблицы, просматривать и распечатывать их.

9. Закройте созданную базу данных «Контакты» и СУБД Microsoft Access.

Задание 3. Знакомство с учебной базой данных «Борей».

Порядок работы

1. Запустите программу СУБД Microsoft Access. Для этого при стандартной установке MS Office выполните: *Пуск/Программы/ Microsoft Access*. В открывшемся окне (рис. 22.8) выберите позицию «Открыть базу данных», а в нижнем окне выберите «База данных Борей».

Примечание. В более ранних версиях MS Office-97 найдите {Пуск/Найти} файл Nwind.mdb, соответствующий базе «Борей».

2. После открытия базы данных «Борей» на экране появится окно с краткой характеристикой базы (рис. 22.9). Нажмите кнопку *OK*.



Рис. 22.8. Открытие базы данных «Борей»

3. Установите табличный вид экрана (*Вид/Таблица*) для вывода краткого описания объектов базы (рис. 22.10). Изучите структуру базы «Борей», переключая вкладки объектов базы – *Таблицы, Запросы, Формы, Отчеты*.

На вкладке *Таблицы* подсчитайте количество таблиц в базе «Борей».

Изучите связи между таблицами. Для этого вызовите схему данных командой *Сервис/Схема данных* или кнопкой *Схема данных* (рис. 22.11). Определите, с какими таблицами связана таблица «Товары».



Рис. 22.9. Окно *Характеристика базы данных*

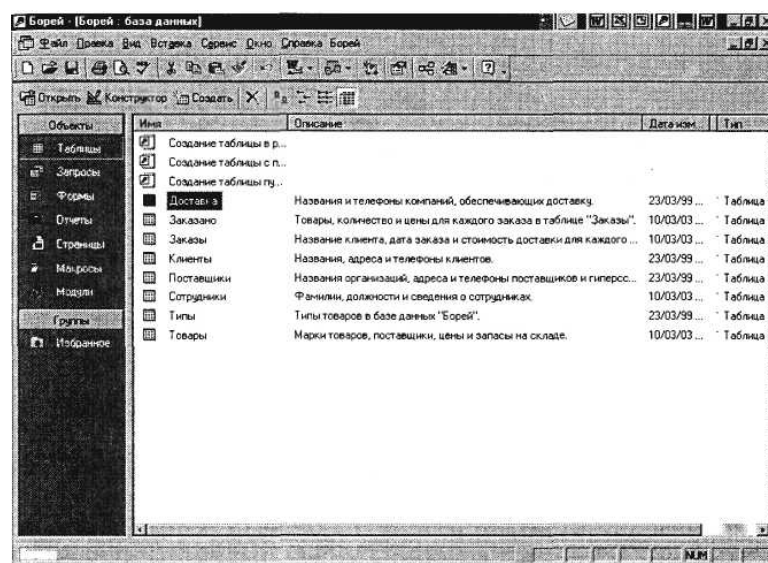


Рис. 10. Таблицы базы «Борей» с описанием

К р а т к а я с п р а в к а . *Таблица* – это объект базы данных, предназначенный для хранения данных в виде записей (строк) и полей (столбцов). Обычно каждая таблица используется для хранения однотипных данных по конкретному вопросу.

5. Выберите объект базы – *Таблицы*. Откройте таблицу «Заказы» двойным щелчком мыши или кнопкой *Открыть*. Определите, сколько в ней записей и полей. Число записей отображается в нижней части окна таблицы справа от кнопок управления записями.

6. Произведите сортировку по клиентам в таблице «Заказы». Для сортировки установите курсор в поле *Клиент* и выполните команду *Записи/ Сортировка/ Сортировка по возрастанию*. Подсчитайте количество заказов у первого клиента в списке.

7. Проведите фильтрацию данных таблицы «Заказы» по дате размещения заказа, расположенной в верхней записи (строке). Для фильтрации выделите дату в верхней строке таблицы и выполните команду *Записи/ Фильтр/ Фильтр по выделенному*. Обратите внимание, как изменился вид таблицы – видны данные, относящиеся только к одной дате. Снимите фильтр {*Записи/Удалить фильтр*}. Закройте таблицу «Заказы».

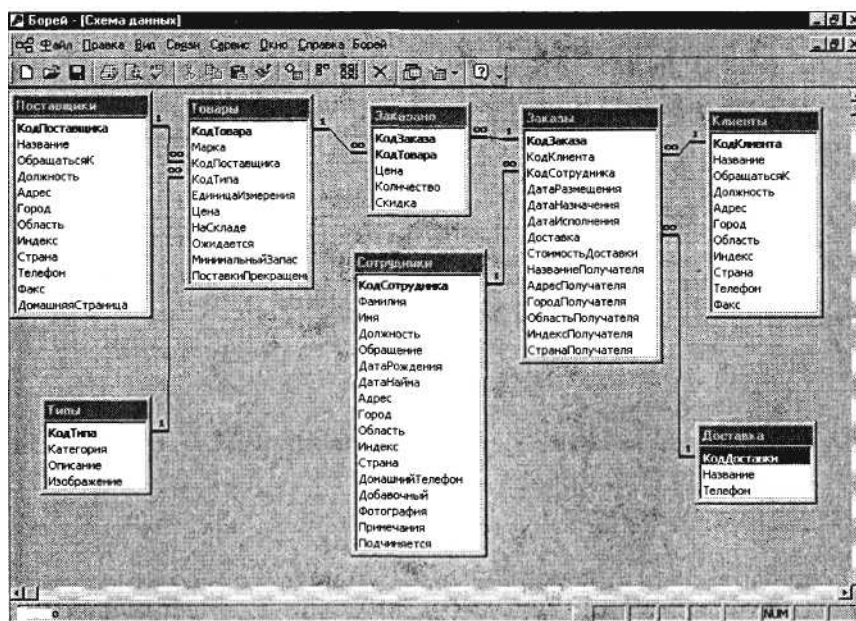


Рис. 11. Схема данных базы «Борей»

8. Откройте таблицу «Клиенты». Определите общее количество клиентов (в нижней части окна таблицы «Клиенты» справа от кнопок управления записями) (рис. 22.12). На рисунке видно, что клиентов – 91.

9. Найдите в поле *Город* Лондон. Для этого установите курсор в поле *Город* и выполните команду *Правка/Найти*. В открывшемся окне *Поиск и замена* (рис. 22.13) на вкладке *Поиск* введите в качестве образца слово «Лондон» и нажмите кнопку *Найти далее*. Произойдет поиск, и курсор будет установлен на названии города – Лондон. Закройте окно *Поиск и замена*.

Борей

Файл

Правка

Вид

Вставка

Формат

Записки

Сервис

Окно

Справка

Борей

</

Рис. 12. Таблица «Клиенты» базы «Борей»

The 'Поиск и замена' dialog box is shown with the 'Поиск' tab selected. The search criteria are:

- Образец:** Лондон
- Поиск в:** Город
- Совпадение:** Поля целиком

Buttons: **Найти далее**, **Отмена**, **Больше >>**

Рис. 13. Поиск по образцу в поле таблицы

10. Выберите фильтрацией клиентов из Лондона (в поле *Город* выделите слово «Лондон» и выполните команды *Записи/Фильтр/Фильтр по выделенному*). Подсчитайте количество клиентов из Лондона.

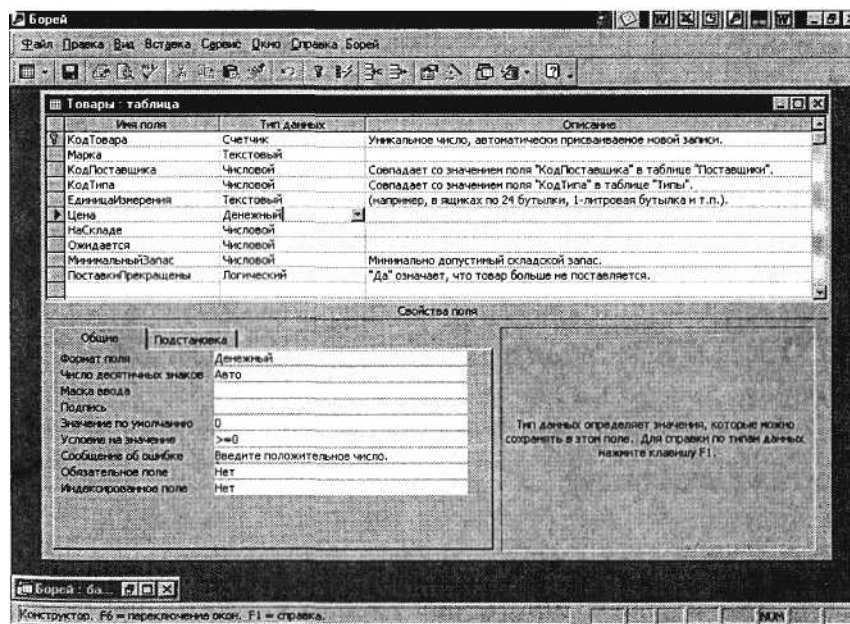


Рис. 14. Окно *Конструктор* таблицы «Товары»

Снимите фильтр (*Записи/Удалить фильтр*). Проведите сортировку по названию клиента (по убыванию).

11. Откройте таблицу «Товары» в *Конструкторе*, для этого установите курсор на таблицу «Товары» и нажмите кнопку *Конструктор* (рис. 22.14). Внимательно рассмотрите внешний вид *Конструктора таблиц*. В верхней части таблицы находится таблица с наименованием полей, их типом данных и описанием. Определите, какое поле – ключевое. В нижней части отображаются свойства поля.

12. Закройте базу данных «Борей» и СУБД MS Access.

Дополнительное задание

Задание 4. Создать базу данных «Заказы на работы» с помощью шаблона средствами мастера.

Изучите связи между таблицами базы данных (*Сервис/Схема данных*).

2.17 Лабораторная работа №19 (2 часа).

Тема: «Создание таблиц и пользовательских форм для ввода данных в СУБД MS ACCESS»

2.17.1 Цель работы: Изучение информационной технологии создания таблиц и пользовательских форм для ввода данных в СУБД Access.

2.17.2 Задачи работы:

1. С помощью мастера создания таблиц по образцу создать таблицу
2. В той же БД создать таблицу в режиме таблицы.

3. В той же БД создать автоформу в столбец по таблице
4. В той же БД создать форму с помощью мастера форм на основе таблицы

2.17.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.17.4 Описание (ход) работы:

Задание 1. С помощью мастера создания таблиц по образцу создать таблицу «Студенты». В качестве образца использовать таблицу «Студенты».

Порядок работы

1. Запустите программу СУБД Microsoft Access и откройте созданную вами БД «Моя пустая база данных».
2. В окне базы данных выберите в качестве объекта «Таблицы». Создайте таблицу с помощью мастера. Для этого выберите команду *Создание таблицы с помощью мастера* (рис. 23.1) или нажмите кнопку *Создать/Мастер таблиц/ОК*.
3. В открывшемся диалоговом окне *Создание таблиц* (рис. 23.2) в качестве образца таблицы выберите «Студенты». Из образцов полей выберите поля (используйте кнопки со стрелками диалогового окна *Выбор одного/Всех полей*) в указанной последовательности:

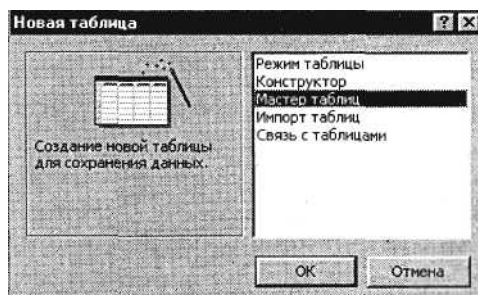


Рис.1. Выбор мастера таблиц при создании новой таблицы

Имя, Отчество, Фамилия, Должность, Адрес, Номер телефона, Специализация.

Нажмите кнопку *Далее*.

4. Задайте имя таблицы – «Студенты». Переключатель установите в положение «Автоматическое определение ключа в Microsoft Access». Нажмите кнопку *Далее*. В следующем окне *Мастер* в «Дальнейших действиях после создания таблицы» выберите *Непосредственный ввод данных в таблицу*. Нажмите кнопку *Готово*.

5. Мастер автоматически создаст ключевое поле, при этом будет создано новое поле *Код студента* с типом данных «Счетчик». Откройте таблицу «Студенты» в *Конструкторе (Вид/Конструктор)* и убедитесь, что слева от имени поля «Код» появился значок ключа – отметка ключевого поля.

6. Перейдите в режим таблицы (*Вид/Режим таблицы*). Перенесите поле *Фамилия* левее поля *Имя*. Для перемещения поля

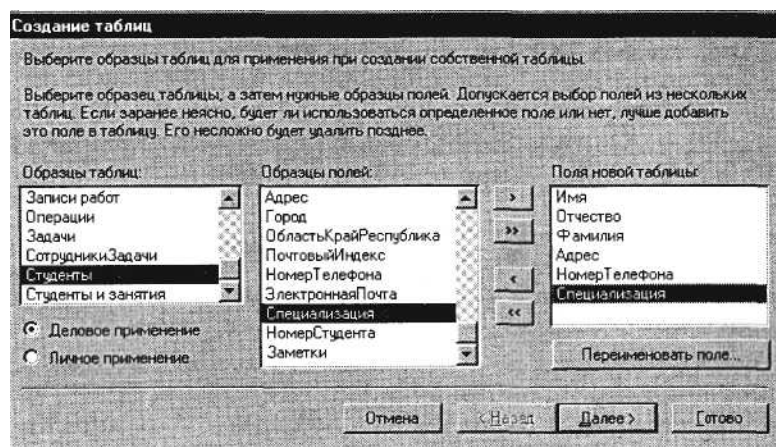


Рис. 23.2. Выбор полей из образца таблицы «Студенты»

КодСтудент	Фамилия	Имя	Отчество	Адрес	Телефон	Специализация
1	Сергеев	Андрей	Львович	г.Долгопрудный	457896	технолог
2	Проскурин	Андрей	Петрович	г.Москва	7458962	технолог
3	Смирнова	Ольга	Ивановна	г.Москва	3698521	бухгалтер
4	Орлова	Инна	Олеговна	г.Долгопрудный	852967	бухгалтер
5	Анплевая	Вера	Петровна	г.Москва	8625471	технолог
6	Березкина	Анна	Романовна	г.Люберцы	748536	технолог
7	Говорова	Дина	Евгеньевна	г.Люберцы	919597	технолог
8	Семенова	Ольга	Сергеевна	г.Москва	9191954	бухгалтер
(Счетчик)						

Рис. 3. Таблица «Студенты»

выделите его щелчком мыши по названию и мышью за название перетащите поле на новое место.

7. Введите в таблицу «Студенты» восемь записей (строк) по образцу (рис. 23.3).
8. Сохраните таблицу.
9. В случае необходимости создайте резервную копию БД на дискете.

Задание 2. В той же БД создать таблицу «Студенты и задания» в режиме таблицы.

Порядок работы

1. Выберите команду *Создание таблицы* путем ввода данных или нажмите кнопку *Создать/Режим таблицы* – рис. 23.4.
2. Переименуйте поля таблицы, присвоив им имена: *Фамилия*, *Описание задания*, *Начальная дата*, *Конечная дата*, *Замечания*.
- Краткая справка. Для изменения имени поля сделайте двойной щелчок мыши по названию поля и введите новое имя.
3. Сохраните таблицу с именем «Студенты и задания».
4. При сохранении программа спросит вас, надо ли создавать ключевое поле? Нажмите кнопку *Да* для создания ключевого поля, при этом будет создано новое поле *Код* с типом данных «Счетчик». Откройте таблицу в *Конструкторе* и убедитесь, что слева от имени поля «Код» появился значок ключа – отметка ключевого поля.

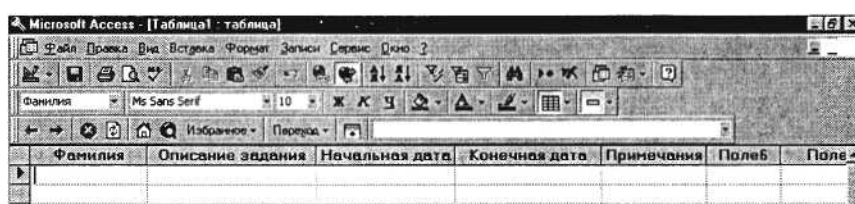


Рис. 4. Вид таблицы, созданной в режиме *Таблица*

5. Скопируйте фамилии студентов из таблицы «Студенты» в таблицу «Студенты и задания». Для копирования перейдите в таблицу «Студенты», выделите поле *Фамилия* и выполните команду *Правка/Копировать*, при этом фамилии будут записаны в буфер памяти. После этого откройте таблицу «Студенты и задания», выделите поле *Фамилия* и выполните команду *Правка/ Вставить*. Убедитесь, что фамилии появились в поле таблицы «Студенты и задания».

6. Перейдите в режим *Конструктор {Вид/Конструктор}* – (рис. 23.5). Установите для полей *Начальная дата* и *Конечная дата* тип данных – «Дата/Время», формат поля – *Краткий формат даты*, маску ввода – *Краткий формат даты* (рис. 23.6).

7. Введите данные в таблицу «Студенты и задания» по образцу, представленному на рис. 23.7.

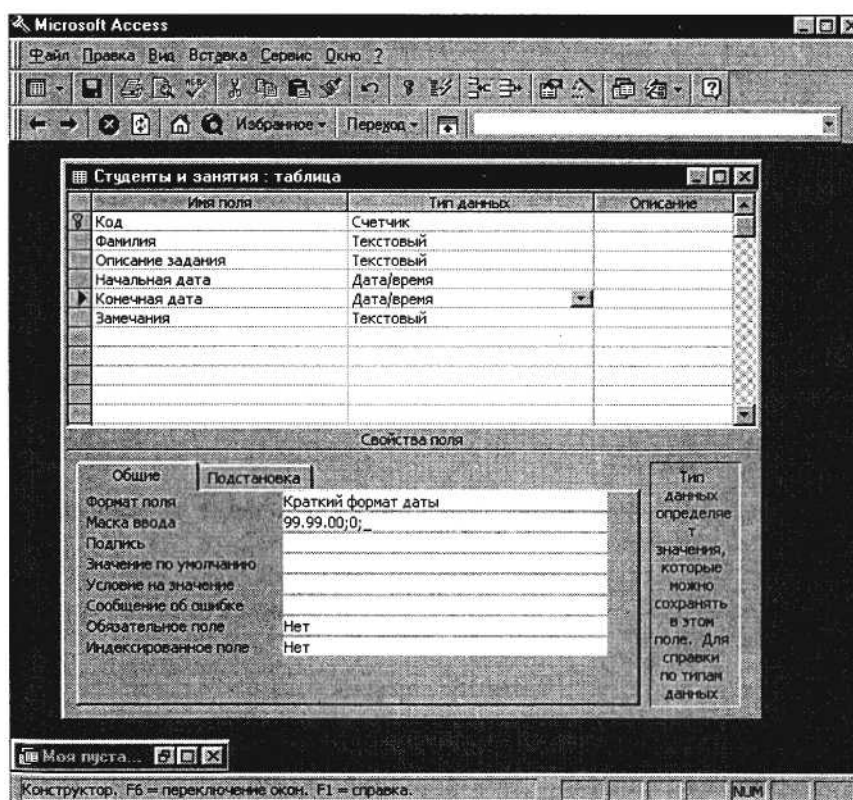


Рис. 5. Задание типа данных – *Дата/Время*

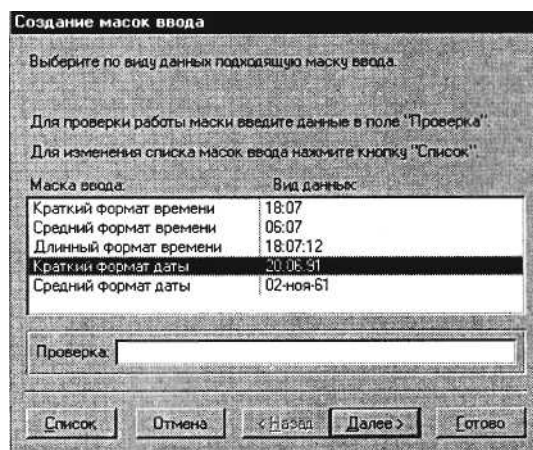


Рис..6. Создание маски ввода даты

Код	Фамилия	Описание задания	Начальная дата	Конечная дата	Занятия
1	Сергеев	Электронная почта	12.03.04	15.05.04	
2	Проскурин	Телеконференции	10.02.04	20.05.04	
3	Смирнов	Браузер	20.01.04	15.04.04	
4	Орлова	Служба FTP	15.01.04	25.04.04	
5	Анпеева	Поисковые системы Интернет	30.01.04	10.05.04	
6	Березкина	Интернет 2	25.02.04	30.05.04	
7	Говорова	IP-телефония	25.02.04	12.05.04	
8	Семенов	Подключение к Интернету	10.03.04	30.05.04	

Рис. 7. Конечный вид таблицы «Студенты и задания»

8. Выполните текущее сохранение таблицы «Студенты и занятия» и закройте таблицу. В случае необходимости создайте резервную копию БД на дискете.

Задание 3. В той же БД создать автоформу в столбец по таблице «Студенты».

Краткая справка. *Форма* – это объект базы данных, отображающий данные из таблиц или запросов. Форма предназначена в основном для ввода данных.

Порядок работы

1. Выберите объект базы – *Формы*. Нажмите кнопку *Создать*, в открывшемся окне *Новая форма* выберите вид формы: «Автоформа: в столбец»; в качестве источника данных укажите таблицу «Студенты» (рис. 23.8). Сохраните созданную форму с именем – «Студенты».

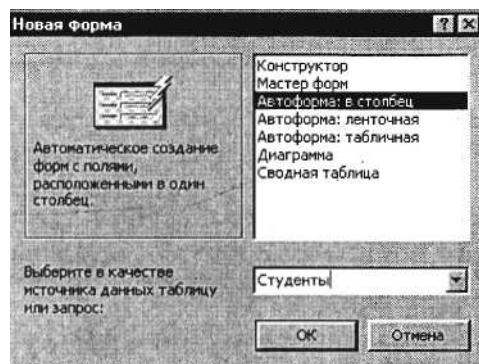


Рис.8. Создание автоформы таблицы «Студенты»

2. Используя кнопки работы с записями в нижней части окна, перейдите на последнюю запись, затем на первую запись.
3. Введите две новых записи с использованием формы «Студенты» (рис. 23.9). Для ввода новой записи используйте кнопки работы с записями в нижней части окна (правую кнопку).

Рис. 9. Автоформа «Студенты»

4. Сохраните созданную форму с именем «Студенты». В случае необходимости создайте резервную копию БД на диске.

Задание 4. В той же БД создать форму с помощью мастера форм на основе таблицы «Студенты и задания».

Порядок работы

1. Для создания формы мастером выберите объект базы – *Формы*. Нажмите кнопку *Создать*, в открывшемся окне *Новая форма* выберите вид формы – «Мастер форм»; в качестве источника данных укажите таблицу «Студенты и задания».

Рис.10. Выбор полей при создании формы мастером форм

2. Выберите поля – *Фамилия, Описание задания, Конечная дата* (рис. 23.10) (для выбора полей используйте кнопки *Выбор одного/всех полей* между окнами выбора); внешний вид формы – в один столбец;

стиль – официальный;

имя формы – «Студенты и задания».

3. В режиме формы (*Вид/Режим формы*) добавьте несколько записей. Для перехода по записям и создания новой записи используйте кнопки в нижней части окна.

4. Сохраните созданную форму с именем «Студенты и задания».

5. Мастером форм на основе всех полей таблицы «Студенты и задания» создайте форму «Студенты и задания 1». Сравните внешний вид созданной формы с формой «Студенты и задания». Введите три новых записи, пользуясь формой «Студенты и задания».

Дополнительные задания

Задание 5. В той же БД создать таблицу «Итоги сессии» с помощью мастера создания таблицы со следующими полями: «Фамилия», «Группа», «Экономика», «Философия», «Математика», «Примечания».

Порядок работы

1. Поля выбирать самостоятельно из разных образцов, применяя возможность переименования полей.

2. Выполнить автоматическое создание ключевого поля при сохранении таблицы. В режиме *Конструктор* проверить тип созданных полей.

3. Скопировать фамилии студентов из таблицы «Студенты». Ввести в режиме таблицы пять записей в созданную таблицу «Итоги сессии». Просмотреть таблицу «Итоги сессии» в режиме Предварительный просмотр и разместить ее на одном листе. Вероятно, вам придется задать альбомную ориентацию листа и уменьшить размеры полей. Сохраните таблицу. В случае необходимости создайте резервную копию БД на дискете.

Задание 6. Создать ленточную и табличную автоформы по таблице «Итоги сессии».

Ввести несколько записей, используя созданные автоформы.

Задание 7. В БД «Контакты» ввести пять произвольных записей в таблицу «Контакты», используя форму «Контакты».

Задание 8. В БД «Заказы на работы» ввести пять произвольных записей в таблицу «Сотрудники», используя форму «Сотрудники».

2.18 Лабораторная работа №20(2 часа).

Тема: «Модификация таблиц и работа с данными с использованием запросов в СУБД MS ACCESS»

2.18.1 Цель работы: Изучение информационной технологии модификации таблиц БД и создания запросов и отчетов в СУБД Access.

2.18.2 Задачи работы:

1. Модификация таблицы
2. Произвести расчеты значений поля
3. Поиск повторяющихся записей по полю
4. Запросы на выборку по условию

2.18.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.18.4 Описание (ход) работы:

Задание 1. Модификация таблицы «Студенты».

Порядок работы

1. Запустите программу СУБД Microsoft Access и откройте созданную вами БД «Моя пустая база данных».
2. Откройте таблицу «Студенты» и проведите ее редактирование: во второй или третьей записях (в зависимости от вашего пола) измените фамилию на свою; скопируйте запись с фамилией «Орлова» на девятую; введите новую запись в режиме *Ввод данных (Записи/Ввод данных)*. Обратите внимание, что произошла фильтрация данных, и все записи стали невидимыми; верните обычный вид таблицы; для этого снимите фильтр (*Записи/Удалить фильтр*); выберите всех студентов с именем «Андрей» (фильтром по выделенному); выберите всех студентов из города «Люберцы»; выберите всех студентов специализации «Технолог».
3. Добавьте в таблицу «Студенты» перед полем *Специализация* новые поля: *Стипендия*, *Надбавка*. Для этого сделайте текущим или выделите поле *Специализация* и выполните команду *Вставка/Столбец*. Присвойте созданным полям соответствующие имена – «Стипендия» и «Надбавка».
4. Перейдите в режим *Конструктор (Вид/Конструктор)* и проверьте, а при необходимости измените типы данных созданных полей (созданные поля должны иметь числовой или денежный тип данных). Вернитесь в режим таблицы (*Вид/Режим таблицы*).
5. Заполните поле *Стипендия* числовыми данными в размере 450 р.
6. Закройте таблицу «Студенты».

Задание 2. Произвести расчеты значений поля «Надбавка» в таблице «Студенты» созданием запроса на обновление. Надбавка составляет 35 % от стипендии.

К р а т к а я с п р а в к а . *Запрос* – это объект базы данных, позволяющий получить нужные данные из таблиц. Запрос представляет собой выборку данных, хранящихся в таблицах, или инструкцию на отбор записей, подлежащих изменению. Наиболее распространенный тип запросов – запрос на выборку. Запрос на выборку отбирает данные из одной или более таблиц по заданным условиям, а затем отображает их в нужном порядке. Запрос можно создать с помощью мастера или самостоятельно. Во втором случае следует в режиме *Конструктор* выбрать таблицы или запросы, содержащие нужные данные, и заполнить бланк запроса.

Порядок работы

1. Для заполнения поля *Надбавка* выберите объект – *Запросы*, вызовите бланк запроса командой *Создать/Конструктор*.
Краткая справка. Бланк запроса – это бланк, предназначенный для определения запроса или фильтра в режиме *Конструктор* или в окне *Расширенный фильтр*. В предыдущих версиях Access использовался термин «бланк запроса по образцу» (QBE).
В открывшемся диалоговом окне *Добавление таблицы* выберите таблицу «Студенты», нажмите кнопку *Добавить* и закройте это окно (рис. 24.1), при этом к бланку запроса добавится *Список полей* таблицы «Студенты» (рис. 24.2). По умолчанию откроется бланк запроса на выборку.

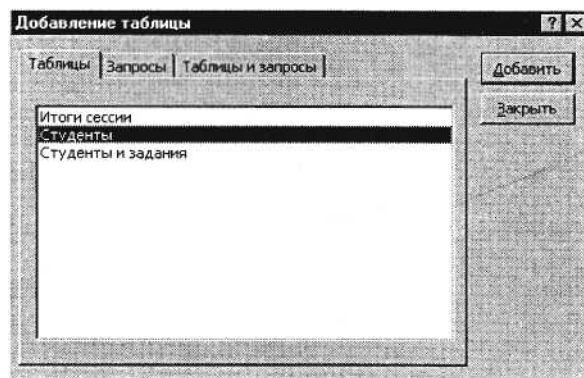


Рис. 1. Добавление списка полей таблицы «Студенты»

Краткая справка. Список полей (в форме и отчете) – окно небольшого размера, содержащее список всех полей в базовом источнике записей. В базе данных Microsoft Access имеется возможность отобразить список полей в режиме *Конструктор форм, отчетов и запросов*, а также в окне *Схемы данных*.

2. В меню *Запрос* выберите команду *Обновление*. Обратите внимание на изменения в бланке вида запроса (*Сортировка* изменилась на *Обновление*).

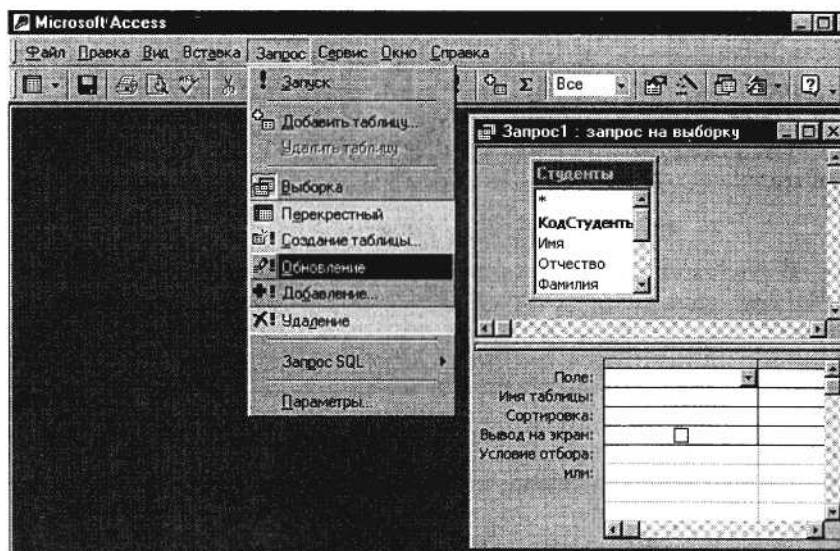


Рис.2. Бланк запроса на выборку

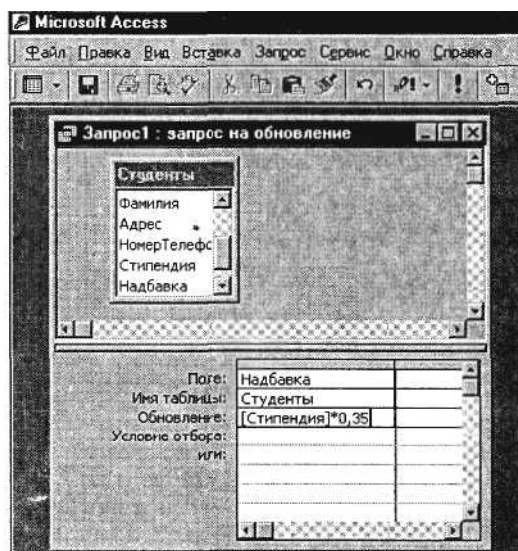


Рис. 3. Бланк запроса для расчета поля *Надбавка*

3. Из списка полей в бланк запроса перетащите поле, которое нужно обновить – *Надбавка*; в строке «Обновление» введите расчетную формулу для заполнения поля *Надбавка* (рис. 24.3).

Поскольку *Надбавка* составляет 35 % от Стипендии, в строке «Обновление» для расчета поля *Надбавка* наберите:

[Стипендия] * 0,35.

К р а т к а я с п р а в к а . Названия полей при наборе формулы в строке «Обновление» заключаются в квадратные скобки.

4. Проведите *Обновление по запросу*, для чего запустите запрос на исполнение командой *Запрос/Запуск* или кнопкой *Запуск* в панели инструментов (в виде восклицательного знака). При этом подтвердите выполнение запроса кнопкой *Да* в открывающемся диалоговом окне.

5. Сохраните запрос под именем «Надбавка» (рис. 4).

6. Откройте таблицу «Студенты» и проверьте правильность расчетов. Если все сделано правильно, то поле *Надбавка* будет заполнено значениями 157,50 р.

7. Измените последовательность полей: поле *Специализация* поместите перед *Стипендией*. Правила перемещения такие же, как во всех приложениях Windows (выделить поле *Примечание*, мышью перетащить на новое место).



Рис. 4. Задание имени запроса при сохранении

8. Сохраните изменения в таблице. В случае необходимости создайте резервную копию БД на диске.

Задание 3. Поиск повторяющихся записей по полю «Имя» таблицы «Студенты».

Порядок работы

1. Выберите объект базы – *Запросы*. Нажмите кнопку *Создать*, в открывшемся окне *Новый запрос* выберите вид запроса – «Повторяющиеся записи» (рис..5).

2. В качестве источника данных укажите таблицу «Студенты» (рис. 6).

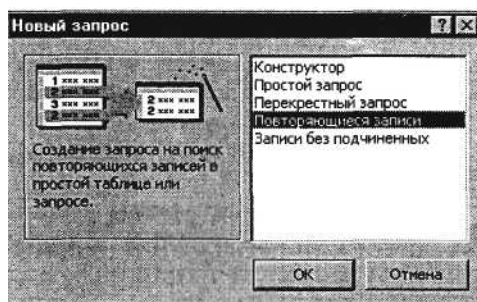


Рис. 5. Создание запроса поиска повторяющихся записей

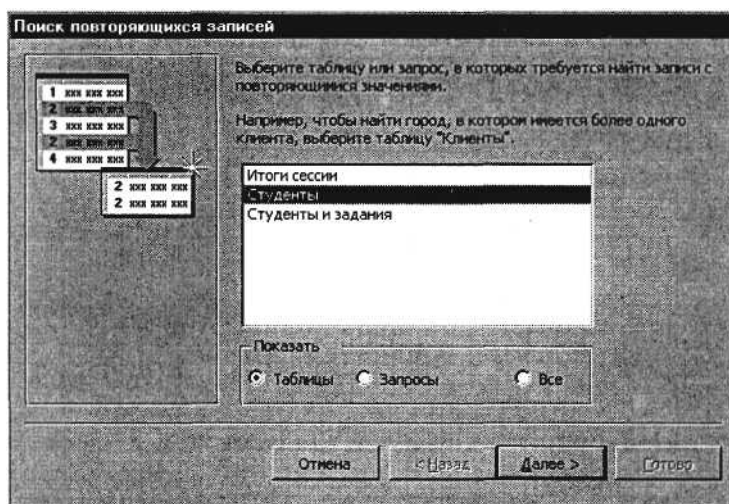


Рис. 6. Выбор таблицы «Студенты» в качестве источника повторяющихся записей

3. В следующих диалоговых окнах выберите поле, по которому будет происходить поиск повторяющихся записей – *Имя*, в качестве дополнительных полей выберите поля *Фамилия* и *Специализация*. В результате работы будут отобраны записи повторяющихся имен студентов, а к ним добавлены сведения о фамилиях и специализации студентов. Сохраните запрос под именем «Повторяющиеся записи».

Задание 4. Запросы на выборку по условию. *Порядок работы*

1. Выберите из таблицы «Студенты» фамилии, имена и телефоны всех студентов, у которых фамилия начинается на букву «С».

Для этого выберите объект базы – *Запросы*. В режиме *Конструктор* создайте запрос на выборку (*Создать/Конструктор*). Добавьте таблицу «Студенты».

Выберите из списка полей таблицы поля *Фамилия*, *Имя*, *Телефон*. В строке «Условие отбора» поля *Фамилия* бланка запроса наберите условие – «С*» (символ * свидетельствует о наличии произвольных символов за буквой «С») (рис. 7).

Задайте сортировку по полю *Имя*. Проверьте, чтобы в строке «Вывод на экран», отвечающей за вывод записей в динамическом наборе на экран компьютера, стояли галочки.

После запуска запроса на исполнение командой *Запрос/Запуск* или кнопкой *Запуск* панели инструментов («!» – восклицательный знак) произойдет отбор по условию. Сохраните запрос под именем «Фамилия С».

3. Выберите всех сотрудников со специализацией «технолог».

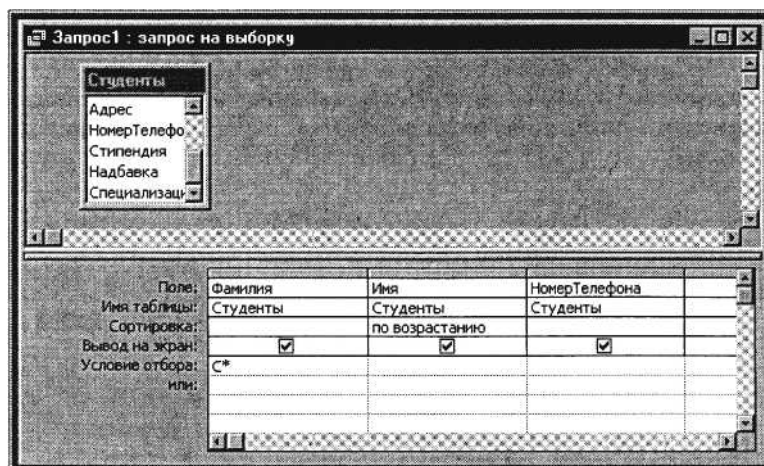


Рис. 7. Отбор фамилий, начинающихся на букву «С»

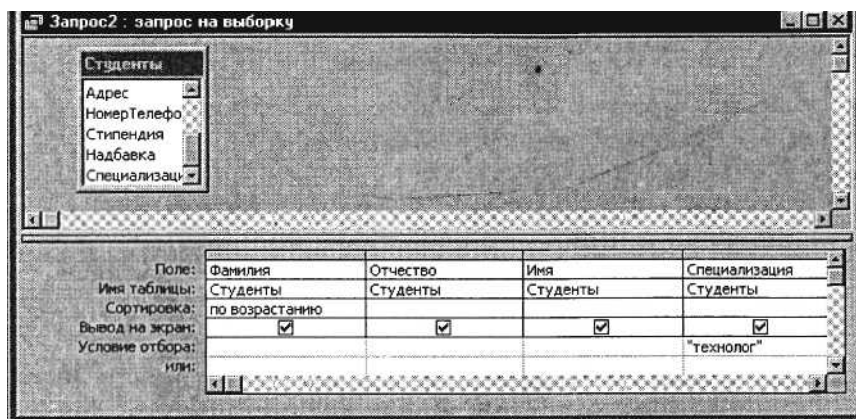


Рис. 8. Отбор студентов по специальности «технолог»

Для этого создайте запрос (*Создать/Конструктор*). Добавьте таблицу «Студенты». Выберите выводимые поля *Фамилия, Имя, Отчество, Специализация*. В строке «Условие отбора» поля *Специализация* бланка запроса наберите условие – «технолог». Задайте сортировку по возрастанию по полю *Фамилия*.

Для запуска запроса выберите команду *Запрос/Запуск*. Сохраните запрос под именем «Запрос – Технолог» (рис. 8).

Дополнительные задания

Задание 5. В той же БД создать запрос на выборку по таблице «Студенты и задания» всех студентов, которые получили задания позже 20.03.04 (в поле «Начальная дата» задайте условие отбора > 20.03.04).

Задание 6. В той же БД по таблице «Студенты и задания» создать запрос на поиск повторяющихся записей по полю «Конечная дата».

2.19. Лабораторная работа №21 (2 часа).

Тема: «Работа с данными и создание отчетов в СУБД MS Access. Комплексная работа с объектами СУБД MS ACCESS»

2.19.1 Цель работы: Изучение информационной технологии создания запросов и отчетов в СУБД Access. Проверка приобретенных навыков работы по созданию и модификации таблиц, пользовательских форм, разных видов запросов и отчетов в СУБД Access.

2.19.2 Задачи работы:

1. Расчет суммарного значения поля
2. Запрос на выборку в интервале дат
3. Запрос на выборку по нескольким полям.
4. Создание автоотчета.

2.19.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер

2. Операционная система Microsoft Windows.

3. Microsoft Office

2.19.4 Описание (ход) работы:

Задание 1. Расчет суммарного значения поля.

Порядок работы

1. В таблице «Студенты» с помощью запроса подсчитайте суммарное значение по полям *Стипендия* и *Надбавка*.

2. Для расчета суммарного значения полей создайте запрос в *Конструкторе* и в бланке запроса выберите поля *Стипендия* и *Надбавка*.

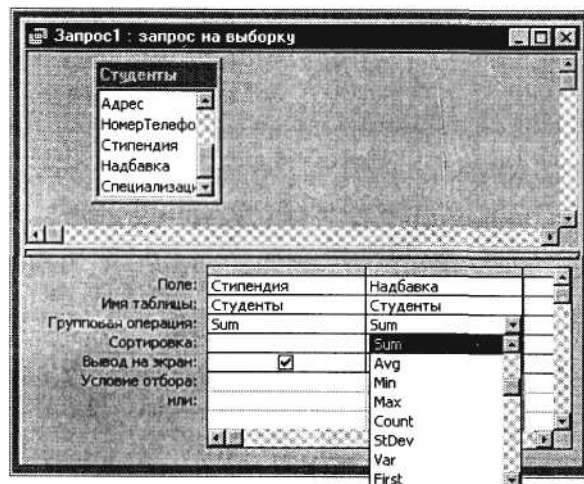


Рис. 1. Расчет суммарного значения по полям *Стипендия* и *Надбавка*

3. Нажмите кнопку *Групповые операции* (X) на панели инструментов. В появившейся строке «Групповые операции» бланка запроса из раскрывающегося списка выберите функцию Sum (рис1). Запустите запрос на исполнение. Сохраните запрос под именем «Запрос –Сумма».

Задание 2. Запрос на выборку в интервале дат.

Создайте по таблице «Студенты и задания» запрос на выборку всех студентов, которым надо представить курсовые работы (конечная дата) с 01.05.04 по 25.05.04 (рис. 2). Задайте сортировку по *Начальной дате* по возрастанию. Сохраните запрос под именем «Запрос – Итог».

К р а т к а я с п р а в к а . При наборе условия используется логический оператор AND. Условие данного запроса имеет вид

$\geq 01.05.04$ AND $\leq 25.05.04$.

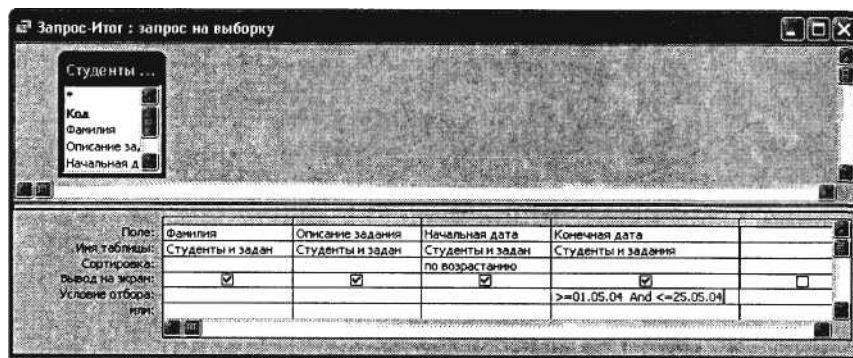


Рис. 2. Запрос с логическим оператором AND на выборку по условию

Задание 3. Запрос на выборку по нескольким полям.

Выведите в запросе всех студентов с сортировкой по фамилиям, обучающихся по специальности «бухгалтер» и проживающих в Москве (рис. 25.3). Сохраните запрос под именем «Бухгалтер – Москва».

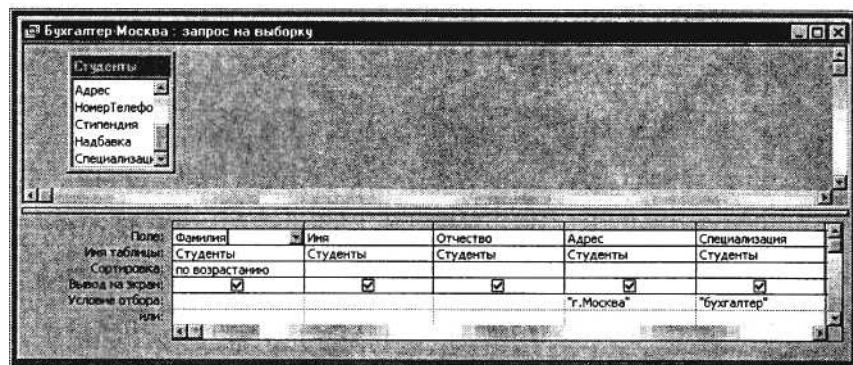


Рис. 3. Выбор по Адресу и Специализации с сортировкой по Фамилии

Краткая справка. *Отчет* – это объект базы данных, предназначенный для вывода (на экран, принтер или в файл) информации из БД.

Задание 4. Создание автоотчета.

Создайте автоотчет в столбец по таблице «Студенты».

Краткая справка. После выбора источника записей и макета (в столбец, ленточный) автоотчет создает отчет, который использует все поля источника записей и применяет последний использованный автоформат.

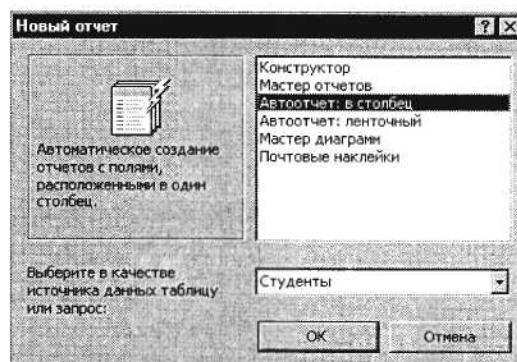


Рис.4. Создание автоотчета в столбец

Выберите объект базы – *Отчеты*. Нажмите кнопку *Создать*, в открывшемся окне *Новый отчет* выберите вид отчета – «Автоотчет: в столбец» (рис. 25.4).

Краткая справка. При выборе вида автоотчета помните, что в ленточном отчете названия полей расположены в строку, как в таблице. На каждой странице размещаются несколько записей, что удобно при просмотре и сравнении данных. Однако не все поля могут поместиться в одной строке, поэтому ленточный автоотчет неудобно использовать при большом числе полей.

В качестве источника данных выберите таблицу «Студенты». Нажмите кнопку *ОК* и дождитесь окончания работы мастера создания автоотчетов.

Просмотрите отчет в режиме *Предварительный просмотр* (Файл/Предварительный просмотр).

Перейдите в режим *Конструктор* и посмотрите, как выглядит отчет в этом режиме.

Сохраните отчет под именем «Студенты».

Задание 5. Создание отчета по таблице «Студенты и задания» с помощью мастера создания отчетов.

Краткая справка. Мастер задает подробные вопросы об источниках записей, полях, макете, требуемых форматах и создает отчет на основании полученных ответов.

Выберите объект базы – *Отчеты*. Нажмите кнопку *Создать*, в открывшемся окне *Новый отчет* выберите вид отчета «Мастер отчетов». В качестве источника данных выберите таблицу «Студенты и задания», выберите все поля, задайте сортировку по полю *Описание задания*, вид макета – в столбец.

Примерный вид отчета приведен на рис. 25.5. Сохраните отчет под именем «Студенты и задания».

The screenshot shows a report window titled "Студенты и задания". The report content is as follows:

Описание задания	Код	Фамилия	Начальная дата	Конечная дата	Замечания
IP-телефония	7	Говорова	25.02.2004	12.05.2004	
Браузер	3	Смирнова	20.01.2004	15.04.2004	
Интернет 2	6				

At the bottom of the window, there is a status bar that reads "Страница: 14" and a navigation bar with buttons for navigating between pages.

Рис. 5. Вид отчета в столбец

Дополнительные задания

Задание 6. В той же БД в таблице «Студенты» создать новое поле «Студент работает» с логическим типом поля.

Создайте запрос на выборку работающих студентов. При создании запроса в строке отбора поля *Студент работает* введите «Да».

Краткая справка. Для создания поля с логическим типом откройте таблицу «Студенты» в режиме *Конструктор* (рис. 25.6). После этого введите имя поля и задайте логический тип поля.

Затем перейдите в обычный вид таблицы и заполните данными созданное поле таблицы, отметив мышью примерно половину студентов как работающих (вы увидите в поле галочку).

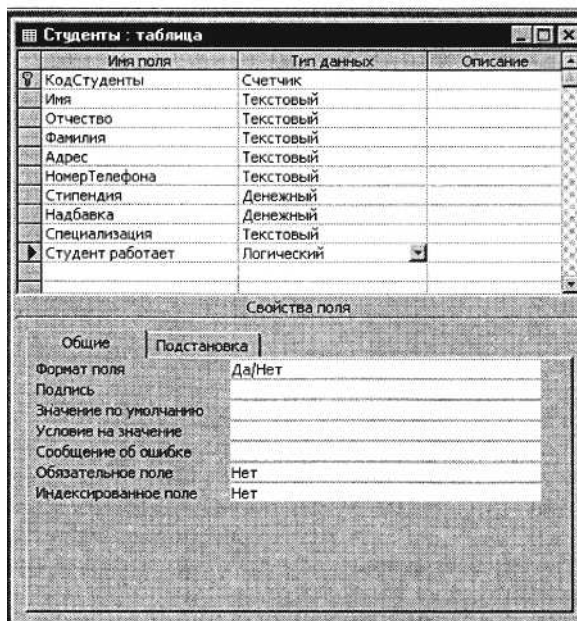


Рис. 6. Задание логического типа поля

Задание 7. По данным таблицы «Студенты» создать запрос на выборку неработающих студентов, обучающихся по специальности «технолог».

Самостоятельная работа

Запустите программу СУБД Microsoft Access и откройте созданную вами БД «Моя пустая база данных».

1. Создайте табличную автоформу на основе всех полей таблицы «Студенты». Заполните таблицу таким образом, чтобы в ней стало 15 записей.

2. В таблицу «Студенты» введите новое поле *Дата рождения*. Установите в созданном поле тип данных – «Дата/Время», формат поля – «Краткий формат даты», маску ввода – «Краткий формат даты».

3. Заполните поле *Дата рождения* данными. Наберите даты в интервале 01.01.1988...31.12.2000.

4. С помощью запроса составьте список студентов, которые родились позже 15.03.1989. Задайте сортировку по фамилиям.

5. В таблицу «Студенты» введите новое поле *Доплаты*. Установите денежный тип данных в созданном поле.

6. С помощью запроса на обновление в таблице «Студенты» рассчитайте поле *Доплаты* из расчета 44 % от Стипендии. Запросу присвойте имя «Доплата 44».

7. Создайте запрос на выборку под именем «Отчества П» для таблицы «Студенты», осуществив выборку всех студентов, у которых отчество начинается на букву «П».

8. Создайте отчет с помощью мастера отчетов по данным таблицы «Студенты», включающий поля: *Фамилия, Имя, Дата рождения, Доплаты*. Отчету присвойте имя «Отчет Студенты».

Ваши навыки и умения оцениваются «Удовлетворительно».

9. В таблицу «Студенты» введите новое поле *Общий доход*. Установите денежный тип данных в созданном поле.

10. Рассчитайте поле *Общий доход* как сумму полей *Стипендия, Надбавка и Доплата*. Запросу присвойте имя «Запрос–Доход».

11. Создайте запрос на повторяющиеся записи по полю *Имя* таблицы «Студенты». Запросу присвойте имя «Повтор имени».

12. В таблице «Студенты и задания» переименуйте поле *Замечания*, присвоив ему новое имя «Защита проекта». Задайте полю *Защита проекта* логический тип поля. Заполните данными созданное поле, отметив мышью примерно половину студентов как защитивших курсовой проект.

Ваши навыки и умения оцениваются «Хорошо».

13. Создайте запрос на выборку из таблицы «Студенты и задания» студентов, защитивших курсовой проект.

В запросе должны быть представлены два поля: поле *Фамилия* с сортировкой по возрастанию и поле – *Защита проекта*. Сохраните запрос под именем «Защита».

14. В таблице «Студенты» с помощью запроса подсчитайте суммарное значение поля *Общий доход*.

15. Создайте автоотчет по запросу «Защита».

Ваши навыки и умения оцениваются «Отлично».

2.20 Лабораторная работа №22(2 часа).

Тема: «Алгоритмизация математических задач»

2.20.1 Цель работы: Усвоить понятия: алгоритм как фундаментальное понятие информатики, способы описания, основные типы алгоритмов, освоить принципы решения задач с использованием основных алгоритмических конструкций.

2.20.2 Задачи работы:

1. знать назначение алгоритма и его определение;
2. знать формы представления алгоритма;
3. знать формы представления алгоритма;
4. уметь представлять алгоритм в виде блок-схемы;
5. уметь приводить примеры алгоритмов и применять их для построения блок-схем;
6. уметь составлять и записывать алгоритм одним из способов.

2.20.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.20.4 Описание (ход) работы:

Общие теоретические сведения

Решение любой задачи на ЭВМ можно разбить на следующие этапы: разработка алгоритма решения задачи, составление программы решения задачи на алгоритмическом языке, ввод программы в ЭВМ, отладка программы (исправление ошибок), выполнение программы на ПК, анализ полученных результатов.

Первый этап решения задачи состоит в разработке алгоритма.

Алгоритм – это точная конечная система правил, определяющая содержание и порядок действий исполнителя над некоторыми объектами (исходными и промежуточными данными) для получения после конечного числа шагов искомого результата.

Алгоритм может быть описан одним из трех способов:

- словесным (пример в начале раздела);
- графическим (виде специальной блок-схемы);
- с помощью специальных языков программирования.

Блок-схема – распространенный тип схем, описывающий алгоритмы или процессы, изображая шаги в виде блоков различной формы, соединенных между собой стрелками.

1. **Линейный алгоритм** – это такой алгоритм, в котором все операции выполняются последовательно одна за другой.

2. **Алгоритмы разветвленной структуры** применяются, когда в зависимости от некоторого условия необходимо выполнить либо одно, либо другое действие.

3. **Алгоритмы циклической структуры.**

Циклом называют повторение одних и тех же действий (шагов). Последовательность действий, которые повторяются в цикле, называют **телом цикла**.

Циклические алгоритмы подразделяют на алгоритмы с предусловием, постусловием и алгоритмы с конечным числом повторов. В алгоритмах с предусловием сначала выполняется проверка условия окончания цикла и затем, в зависимости от результата проверки, выполняется (или не выполняется) так называемое тело цикла.

Задание 1. Определить площадь трапеции по введенным значениям оснований (a и b) и высоты (h).

Запись решения задачи на алгоритмическом языке:

алг трапеция

вещ a,b,h,s

нач

ввод f,b,h

$s := ((a+b)/2) * h$

вывод s

кон

Запись алгоритма в виде блок-схемы (рис. 1):

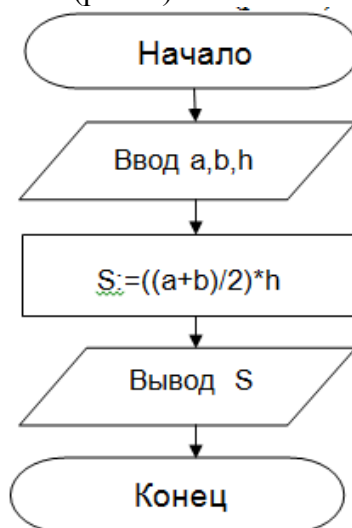


Рисунок 1. Блок-схема линейного алгоритма

Задание 2. Определить среднее арифметическое двух чисел, если a положительное и частное (a/b) в противном случае.

Запись решения задачи на алгоритмическом языке:

алг числа

вещ a, b, c

нач

ввод a, b

если $a > 0$

то $c := (a+b)/2$

иначе $c := a/b$

все

вывод c

кон

Запись алгоритма в виде блок-схемы (рис. 2):

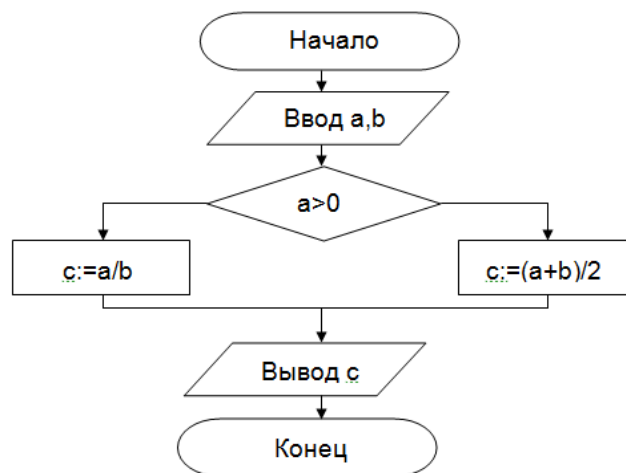


Рисунок 2. Блок-схема алгоритма с ветвлением

Задание 3. Составить алгоритм нахождения суммы целых чисел в диапазоне от 1 до 10.

Запись решения задачи на алгоритмическом языке:

алг сумма

вещ a, s

нач

$S := 0;$

$A := 1;$

нц

пока $a \leq 10$

$S := S + a;$

$A := a + 1;$

кц

вывод S

кон

Запись алгоритма в виде блок-схемы (рис. 3):

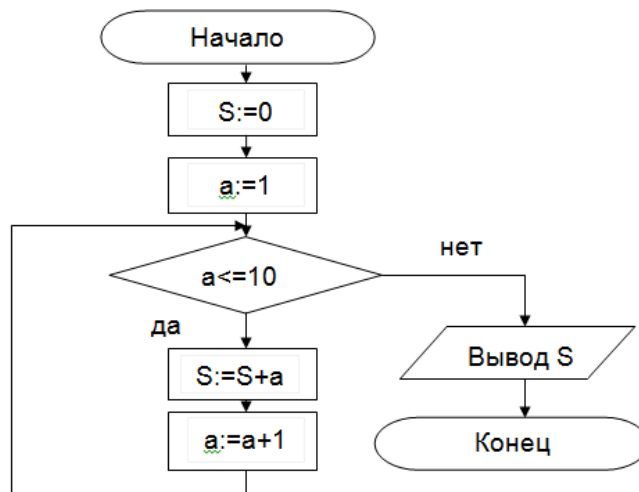


Рисунок 3. Циклический алгоритм с предусловием

В алгоритме с постусловием сначала выполняется тело цикла, а затем проверяется условие окончания цикла. Решение задачи нахождения суммы первых десяти целых чисел в данном случае будет выглядеть следующим образом:

```

алг сумма
  вещ a,s
нач
  S:=0;
  A:=1;
нц
  S:=S+a;
  A:=a+1;
пока a<=10
кц
  вывод S
кон
  
```

Запись алгоритма в виде блок-схемы (рис. 4):

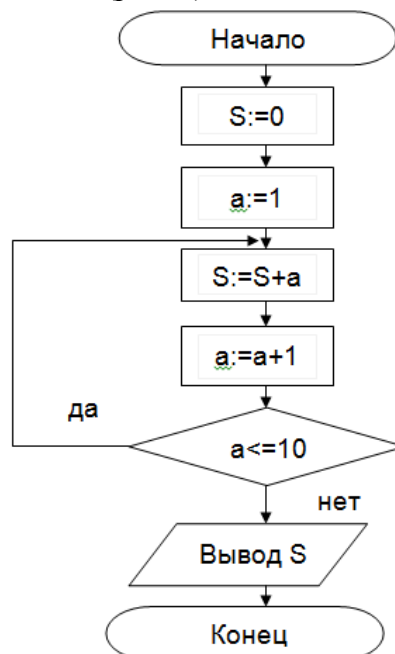


Рисунок 4. Циклический алгоритм с постусловием

2.21 Лабораторная работа №23(2 часа).

Тема: «Элементы программирования на языке высокого уровня»

2.21.1 Цель работы: изучение основных принципов программирования, обучение основам языков высокого уровня Паскаль, а также выработки у студентов современного стиля программирования.

2.21.2 Задачи работы:

1. Создать простую программу, выполняющую арифметические действия, осуществляющую проверку некоторых условий и выполняющую обмен информацией с пользователем.
2. Создать простую программу, выполняющую в цикле определенные действия с массивами, и выводящую результаты в удобной для восприятия форме.
3. Получить практические навыки работы с основными конструкциями языков.
4. Развитие у студентов навыков использования алгоритмических языков высокого уровня при решении широкого круга практических задач.

2.21.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.21.4 Описание (ход) работы:

Простейшие программы

Программа 1. Напишите программу вычисления функции $z=\ln(x^2+y^2+1)$ по введенным X и

Y.

```
PROGRAM Primer1; {Программа вычисления функции  $z=\ln(x^2+y^2+1)$ }  
USES Crt;      {Раздел модулей}  
VAR      X, Y, Z :Real; {Раздел переменных}  
BEGIN      {Раздел операторов}  
  ClrScr;  
  Write('Вычисление функции  $z=\ln(x*x+y*y+1)$ );  
  Write('Введите X: ');  
  Readln(X);  
  Write('Введите Y: ');  
  Readln(Y);  
  Writeln;  
  Z:=LN(X*X+Y*Y+1);  
  Writeln('Z=',Z:5:1)  
END.          {Конец раздела операторов}
```

В фигурных скобках размещены комментарии к программе. Они могут быть опущены, так как при работе компилятора пропускаются. ClrScr – процедура очистки экрана в Турбо Паскале. Процедура Writeln с пустым списком вывода переводит строку. Процедура ClrScr описана в модуле Crt, а процедуры Write, Read, Writeln, Readln в модуле System. В последнем операторе Writeln для переменной z указан формат ее вывода на экран: всего пять позиций

(включая десятичную точку и знак) и одна цифра после десятичной точки. При форматированном выводе происходит автоматическое округление результата.

Программа 2. Напишите программу, вычисляющую площадь трапеции по заданным длинам оснований и высоты.

```
PROGRAM Trap;  
USES Crt;  
VAR A, B, H, S: Real;  
BEGIN  
  ClrScr;  
  Writeln ('Вычисление площади трапеции');  
  Write('Введите длину первого основания:');  
  Readln(A);  
  Write('Введите длину второго основания:');  
  Readln(B);  
  Write('Введите высоту:');  
  Readln(H);  
  S:=((A+B)/2)*H;  
  Writeln;  
  Writeln('Площадь трапеции равна:',S:6:2)  
END.
```

Задания к лабораторной работе

Написать программы для решения предложенных ниже задач. Во всех вариантах аргументы функций, заданные коэффициенты и другие данные вводятся с клавиатуры, а полученные результаты выводятся на экран.

Задание 1. Вычислить значение функции:

- 1) $y = (1 + \sin x)^2$;
- 2) $y = \sqrt{1 + x^2}$;
- 3) $y = |5 - 6x|$;
- 4) $y = \log_2 \frac{x^2 + 1}{3}$;
- 5) $y = \sqrt[4]{1 + x + x^2}$
- 6) $y = \sin(\cos x)$;
- 7) $y = \operatorname{ctg}(x - 4)$;
- 8) $y = 2^{(x+1)}$

Задание 2.

1. Вычислить длину окружности, площадь круга и объем шара одного и того же заданного радиуса.
2. По координатам трех вершин некоторого треугольника найти его периметр и площадь.
3. Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов.
4. Вычислить площадь полной поверхности и объем прямого кругового цилиндра, заданного высотой и диаметром основания.
5. Вычислить площадь полной поверхности и объем прямого конуса, заданного высотой и радиусом основания.
6. Вычислить периметр и площадь правильного 15-угольника, вписанного в окружность заданного радиуса.
7. Вычислить площадь полной поверхности и объем правильного тетраэдра по длине

его ребра.

8. Вычислить площадь параллелограмма, у которого даны длины сторон и величина угла между ними (в градусах).

Операторы языка программирования Паскаль

1. Написать программу вычисления значения функции

$$y = \begin{cases} |x|, & x \leq 0 \\ e^x, & 0 < x \leq 1 \\ 1/(2x+1), & x > 1 \end{cases}$$

Значение x вводится оператором **READ**.

Программа 1.

```
PROGRAM PR31;  
VAR X, Y: REAL;  
BEGIN  
  WRITE ('X=');  
  READLN (X);  
  6. IF X<=0 THEN Y:=ABS(X)  
  ELSE  
  7. IF X<=1 THEN Y:=EXP(X)  
  ELSE Y:=1/(2*X+1);  
  WRITELN (' Y=',Y);  
END.
```

Пример 3.2.

8. Написать программу, подсчитывающую сумму только положительных из трех данных целых чисел.

Программа 3.2.

```
Program pr32;  
Uses CRT;  
Var a, b, c, s: integer;  
Begin  
  ClrScr;  
  Writeln ('Введите три целых числа:');  
  Readln (a, b, c);  
  s:=0;  
  If a>0 then s:=s+a;  
  If b>0 then s:=s+b;  
  If c>0 then s:=s+c;  
  Writeln ('s=',s);  
End.
```

Задания к работе №2

Задание 1. Написать программу вычисления функции. В случае ввода недопустимого аргумента на экране должен печататься текст: "Недопустимый аргумент".

$$1) y = \begin{cases} x^2, & 0 < x < 1 \\ x^3, & 1 \leq x \leq 10 \end{cases}$$

$$2) y = \begin{cases} 3x - 4, & 0 < x \leq 2 \\ x^2 - 2, & 2 < x \leq 8 \end{cases}$$

$$3) y = \begin{cases} x^3 - 7, & 0 < x \leq 2 \\ 7 - 3x, & 2 < x \leq 9 \end{cases}$$

$$5) y = \begin{cases} x - x^2, & 0 < x \leq 1 \\ 4 - 2x, & 1 < x \leq 8 \end{cases}$$

$$6) y = \begin{cases} 3x - 4, & 0 < x \leq 1 \\ -x^2, & 1 < x \leq 10 \end{cases}$$

$$7) y = \begin{cases} 10x - 3, & 0 < x \leq 3 \\ x^2 - 8, & 3 < x \leq 5 \end{cases}$$

$$4) y = \begin{cases} 5 - 2x, & 0 < x \leq 2.5 \\ 4x - 10, & 2.5 < x \leq 5 \end{cases}$$

$$8) y = \begin{cases} 8 - 2x, & 0 < x \leq 2 \\ x + 2, & 2 < x \leq 10 \end{cases}$$

Задание 2. Написать программу для решения следующей задачи.

1. Найти максимум из трех заданных целых чисел a, b, c.
2. Даны произвольные целые числа a, b, c. Определить, можно ли построить треугольник с длинами сторон, равными этим числам.
3. Найти действительные корни квадратного уравнения: $ax^2 + bx + c = 0$. Если таких корней нет, то вывести сообщение об этом.
4. Решить систему уравнений
$$y = \begin{cases} ax + by = c \\ dx + ey = f \end{cases}$$
5. Дана точка A(x,y). Определить, какой из координатных четвертей она принадлежит.
6. Определить, какая из трех точек A(x1,y1), B(x2,y2), C(x3,y3) ближе к началу координат.
7. Если треугольник со сторонами a,b,c является прямоугольным, то напечатать длину медианы, выходящей из прямого угла, в противном случае напечатать его периметр.

Если треугольник со сторонами a,b,c является равнобедренным, найти его площадь, в противном случае найти его периметр.

Операторы цикла в языке программирования Паскаль

Программа 1. Цикл с параметром (шаг +1)

```
PROGRAM FACTORIAL1;
VAR I, N:INTEGER;
F: REAL;
BEGIN
WRITE ('Введите N: ');
READLN (N); F:=1;
FOR I:=1 TO N DO F:=F*I;
Writeln ('Факториал от',N 'равен' ,F);
END.
```

Программа 2. Цикл с параметром (шаг -1)

```
PROGRAM FACTORIAL2;
VAR I, N:INTEGER;
F: REAL;
BEGIN
WRITE ('Введите N: ');
READLN (N); F:=1;
FOR I:=N DOWNT0 1 DO F:=F*I;
Writeln ('Факториал от',N 'равен' ,F);
END.
```

Программа 3. Цикл с предусловием

```
PROGRAM FACTORIAL3;
VAR I, N:INTEGER;
F: REAL;
BEGIN
WRITE ('Введите N: ');
READLN (N); F:=1; I:=1;
While I<=N DO
BEGIN
```

```

F:=F*I;
I:=Succ(I);
END;
WRITELN ('Факториал от',N 'равен' ,F);
END.

```

Программа 4. Цикл с постусловием

```

PROGRAM FACTORIAL4;
VAR I, N:INTEGER;
F: REAL;
BEGIN
WRITE ('Введите N: ');
READLN (N); F:=1; I:=1;
REPEAT
F:=F*I;
I:=SUCC(I)
UNTIL I>N;
WRITELN ('Факториал от',N 'равен' ,F);
END.

```

Задания к лабораторной работе

Задание 1

Написать программу, печатающую таблицу значений функции на отрезке [0;1] с шагом

$h=0.1$:

- | | |
|------------------|----------------------|
| 1) $y = 2^x$ | 5) $y = e^{2x+1}$ |
| 2) $y = \sin 2x$ | 6) $y = \arctg(1+x)$ |
| 3) $y = \cos 2x$ | 7) $y = \ln(x+1)$ |
| 4) $y = (x+1)^2$ | 8) $y = x^3+1$ |

Задание 2

По заданной формуле члена последовательности с номером k составить программу вычисления суммы всех членов последовательности, не меньших заданного числа ϵ :

- | | |
|------------------------------|-----------------------------------|
| 1) $\frac{1}{(2k-1)(2k+1)}$ | 5) $\frac{k+1}{k^3 + \sqrt{k+2}}$ |
| 2) $\frac{k}{(k+1)^2 + 3}$ | 6) $\frac{k+4}{(k^2+2)(k+8)}$ |
| 3) $\frac{2k}{(k^2+1)(k+2)}$ | 7) $\frac{3(k+1)}{7k^2+9}$ |
| 4) $\frac{2k+1}{(2k^2+1)k}$ | 8) $\frac{1}{k^2+3k+4}$ |

2.22 Лабораторная работа №24 (2 часа).

Тема: «Локальные и глобальные сети ЭВМ»

2.22.1 Цель работы: научиться работать в локальной и глобальной компьютерной сети

2.22.2 Задачи работы:

1. Поиск данных в Internet и копирование их на свой винчестер
2. Почта в Internet – создание почтового ящика, отправка и получение корреспонденции.

2.22.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.22.4 Описание (ход) работы:

Основными практическими задачами работы в Internet являются:

- 1 Поиск необходимой информации (статей, книг, новостей, почты, объявлений и др.).
- 2 Копирование найденной информации на свой компьютер или его распечатка непосредственно из Internet.
- 3 Создание почтового ящика, отправка и получение корреспонденции.
- 4 Создание сайта в Internet

Для выхода в сеть Internet нужно дважды щелкнуть мышью по значку Internet на Рабочем столе, появится окно Microsoft Internet Explorer с окном Удалённое соединение (см. рисунок 16.1).

После нажатия клавиши Подключиться компьютер соединяется сетью Internet.

Задание 1

Поиск и копирование данных выполняется следующим образом:

1 Нажать кнопку Поиск на панели Стандартная. Рабочая зона разделится на две части: левая – для ввода ключевых слов для поиска, правая – для вывода найденной информации. Поиск выполняется с помощью поисковиков Aport, Rambler и др. На примере поискового сервера Апорт осуществите поиск информации.

2 В поле Адрес введите адрес сервера: www.aport.ru. Нажмите клавишу Enter на клавиатуре или **Обновить** на панели инструментов.

3 Ввести ключевые слова в левую часть, нажать клавишу Начать поиск. В результате в левой части появится количество найденных документов и список первых 15-ти из них. Перейти к следующим 15 документам можно внизу списка. Содержание выделенного документа выводится в правой части.

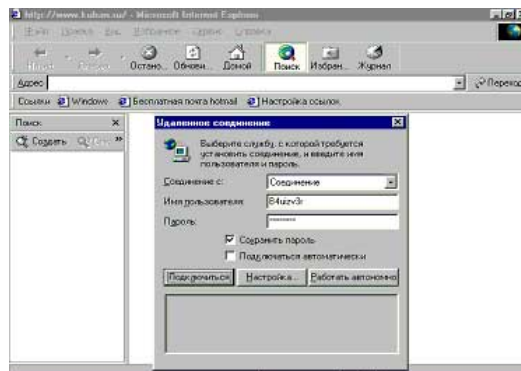


Рисунок 1 Диалоговое окно «Удаленное соединение»

4 Выделить текст выбранного документа командой **Правка\Выделить всё**, затем выполнить команду **Правка\Копировать** выйти из Internet.

5 Открыть на своём компьютере новый документ, выполнить команду **Правка\Вставить**. Можно вместо выполнения п. 4, не выходя из Internet, выполнить команду **Файл\Сохранить как...** и указать в окне сохранения имя файла и папку, в которой документ должен быть сохранён., например, **Диск 3,5(A)**.

Задание 2

Электронный почтовый ящик создаётся следующим образом:

1 После открывания окна Microsoft Internet Explorer в поле Адрес набрать WWW.mail.ru – это сайт (страница, в переводе – местоположение, местонахождение) компании ``Mail.ru`` которая бесплатно предоставляет в Internet почтовые услуги, аналогично можно выбрать Rambler или Yandex, которые помимо поиска также предоставляют возможность создания бесплатного почтового ящика.

2 Адрес почтового ящика состоит из Логина и Доменного имени, разделённых символом ``собака`` @ (В Китае его зовут ``улитка``, в Венгрии - ``червяк``, в Норвегии – ``котёнок``). Например, в адресе komkov@mail.ru логином является komkov, а доменным именем – mail.ru. Login – это название почтового ящика, а Доменное имя – название сервера, т.е. компьютера (Web-сервера), на котором хранятся Web-страницы.

В левой части окна нажать Регистрация, в правой части появится текст Договора по пользованию электронной почтой. В конце текста Договора нажать **Я согласен**, появится регистрационная страница, на которой нужно выбрать регистрационное имя, в строке Login ввести название почтового ящика совпадающее с вашей фамилией транслитерируемой латинскими буквами, например, Ivanov, для человека с фамилией Иванов, в строке Доменное имя выбрать из списка mail.ru, list.ru или любое другое, в строке Укажите пароль ввести свой пароль, повторить его в строке Повторите пароль.

3 Если Вы забыли пароль, запишите в окнах Вопрос и Уникальный ответ, те данные, которые помогут вспомнить пароль. Например, для пароля Мурка вводим вопрос ``Любимые домашние животные?`` и ответ ``Кошка``.

4 Нажать надпись Продолжить регистрацию, компьютер выполняет сверку с базой данных, если такого логина нет, то регистрация продолжается, если есть, то база данных просит сменить его. Появляется надпись ``Регистрация успешно завершена``.

5 После регистрации заполняется поле необязательных данных (имя, отчество, дата рождения, организация и др.), вводится имя другого, ранее сделанного почтового ящика, на который будут переданы логин и пароль только что зарегистрированного почтового ящика.

Почтовый ящик создан, им можно пользоваться:

6 В окне **Mail.ru** щелкнуть по тексту Внести изменения, ввести логин и пароль, в пункте Почта выбрать Написать письмо, заполнить графы От кого, Кому (указать почтовый ящик адресата), ввести текст письма, выбрать кодировку (для России принят код кой-8), нажать Отправить. Если адрес есть и верен, т.е. графа Кому заполнена правильно, появится текст ``Сообщение успешно отправлено``, если нет – ``Не могу послать сообщение``.

7 К письму можно присоединить с помощью браузера до 10 файлов общим размером около 10 Мб, нажав на кнопку **Обзор** ниже текста письма. После выбора файлов нужно щелкнуть мышью по значку **Скрепка** или **Прикрепить**, файлы присоединятся к письму, затем нажать **Отправить**.

8 Для чтения письма получатель должен в своём ящике нажать на строку **Новое письмо** или **Чтение письма**. Количество полученных корреспонденций сообщается в том же окне ящика.

9 Для выхода из электронной почты нужно нажать **Завершить сеанс** или **Выход** иначе любой пользователь может с этого компьютера войти в Ваш ящик, не вводя логина и пароля, т.к. они уже введены, а сеанс не завершён.

2.23 Лабораторная работа №25 (2 часа).

Тема: « Работа в глобальной сети Internet»

2.23.1 Цель работы: Использование собственной учетной записи, изучение настроек, работа с сообщениями..

2.23.2 Задачи работы:

1. Настройка окна области Почта Microsoft Outlook.
2. Создание сообщения.
3. Работа с поступившей почтой.

2.23.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.23.4 Описание (ход) работы:

1 Настройка окна области Почта Microsoft Outlook

Для работы с сообщениями электронной почты в Microsoft Outlook предназначена область **Почта**, которая содержит почтовые папки: **Входящие**, **Исходящие**, **Отправленные** и др. В зависимости от настроек окно этой области может выглядеть по-разному. Вид окна, установленный по умолчанию, приведен на рисунок 16.2.

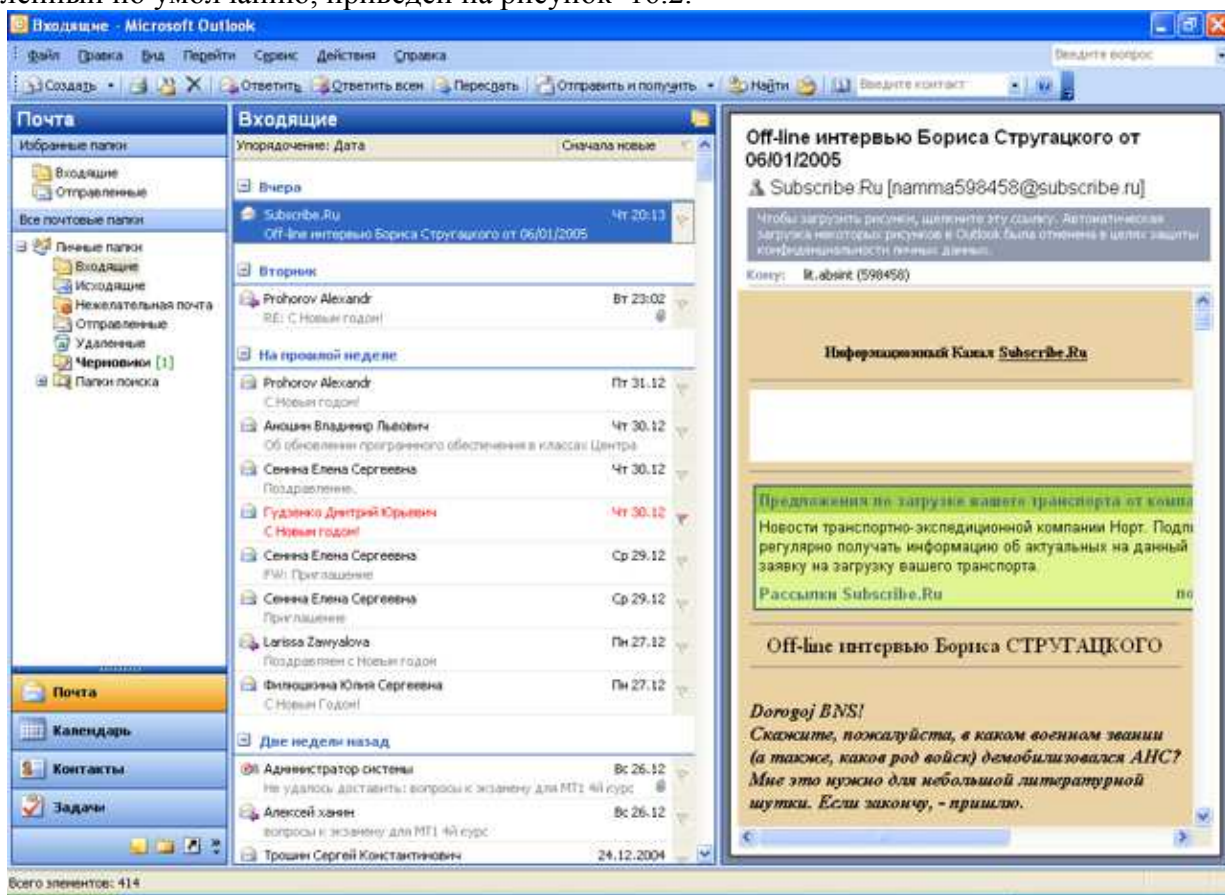


Рисунок 2 - Вид окна области "Почта"

Левая часть окна называется **Область переходов**. В нижней части она содержит значки других областей Microsoft Outlook (**Календарь**, **Контакты**, **Задачи** и др.). В верхней части Область переходов содержит ярлыки почтовых папок.

В центральной части окна отображается список сообщений почтовой папки, выделенной в **Области переходов**.

Правая часть окна называется **Область чтения**. В области чтения отображается содержание сообщения, выделенного в центральной части.

Настройка отображения областей и их расположения производится с использованием команд меню **Вид** (рисунок 3).

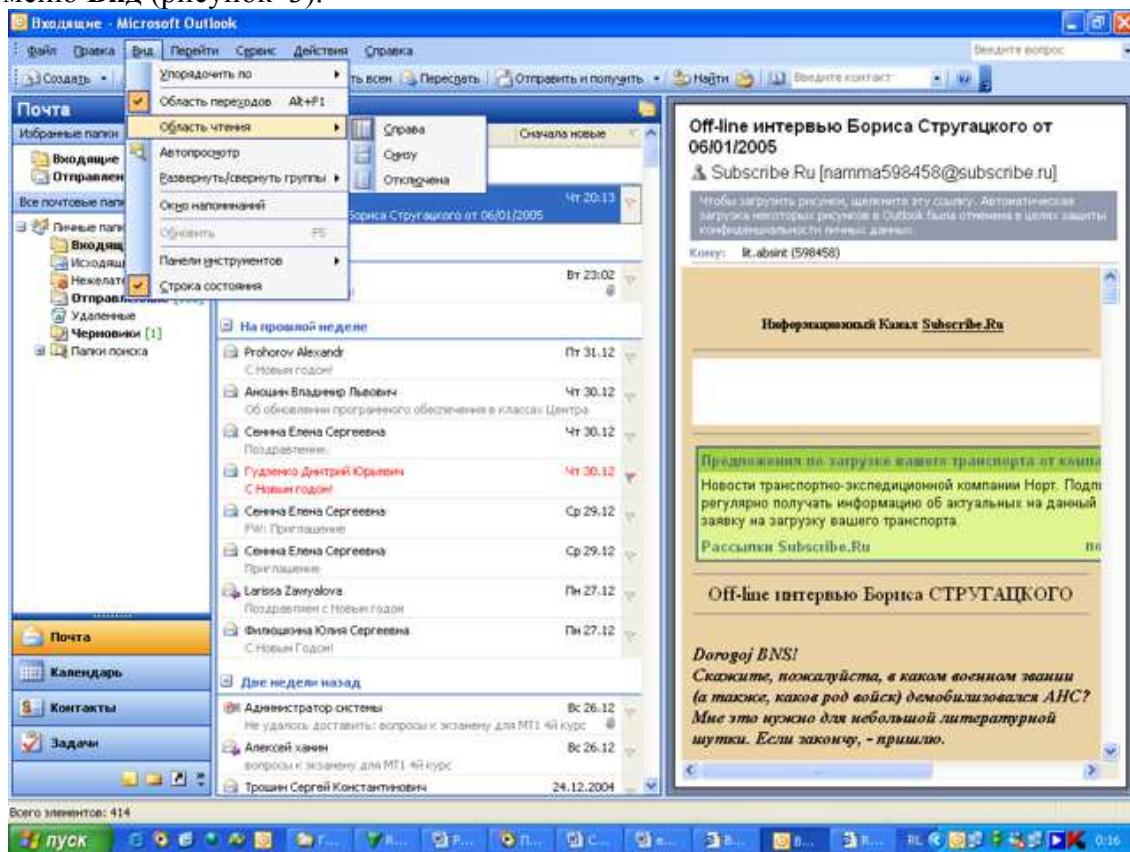


Рисунок 3 - Настройка отображения областей в окне Microsoft Outlook

Границы между областями окна можно перетаскивать при нажатой левой кнопке мыши. Более традиционным является расположение **Области чтения** в нижней части окна (рисунок 4).

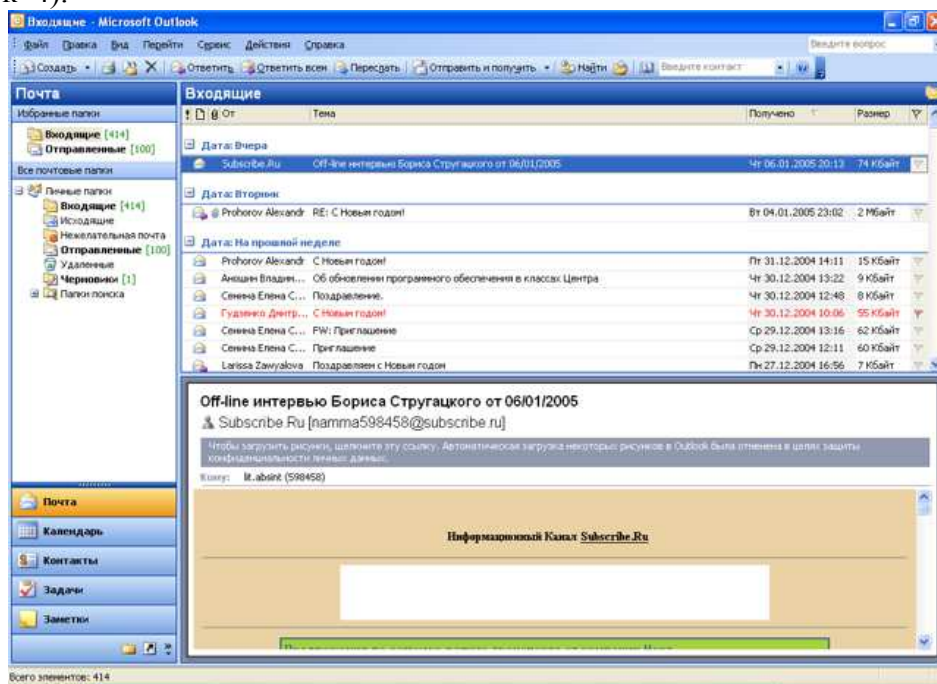


Рисунок 4 - Окно области "Почта" Microsoft Outlook

2 Создание сообщения

2.1 Создание простого сообщения

Для создания сообщения следует в области переходов окна Microsoft Outlook перейти к области (модулю) **Почта** и нажать кнопку **Создать** панели инструментов **Стандартная** (рисунок 4 предыдущий пункт).

К созданию сообщения можно приступить и находясь в любой другой области Microsoft Outlook. Для этого нужно щелкнуть по стрелке в правой части кнопки **Создать** и выбрать команду **Сообщение** (рисунок 5).

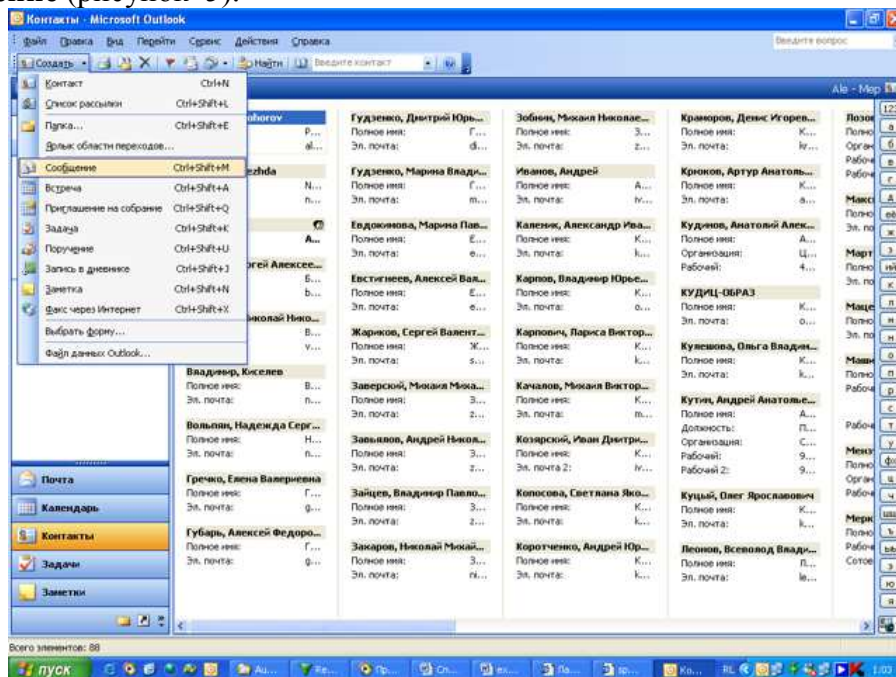


Рисунок 5 - Создание сообщения

В окне **Сообщение без заголовка** (рисунок 6) в поле **Кому** введите электронный адрес получателя.

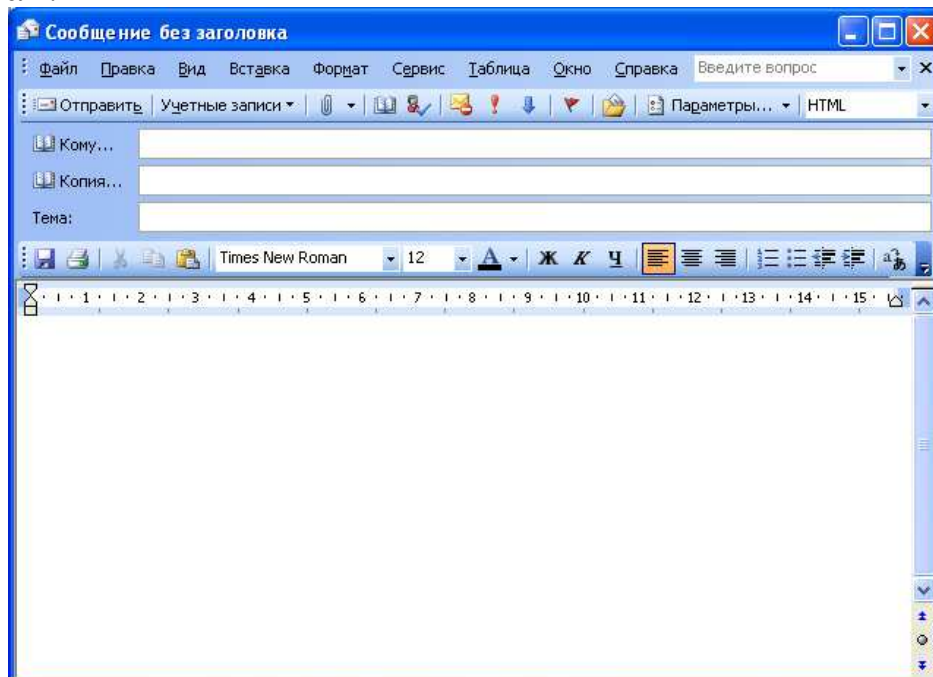


Рисунок 6 - Окно Сообщение без заголовка

Для выбора адреса из адресной книги в окне сообщения (см. рисунок 6) можно нажать

кнопку **Кому**, после чего появится окно **Выбор имен** (рисунок 7).

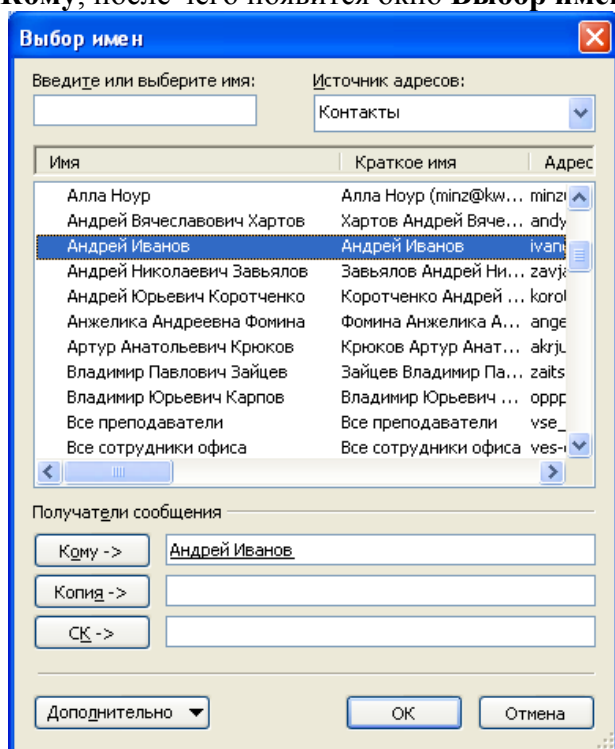


Рисунок 7 - Выбор адреса в окне "Выбор имен"

В окне (см. рисунок 7) необходимо выбрать имя адреса в списке, нажать кнопку **Кому** и нажать кнопку **ОК**.

В окне сообщения (см. рисунок 6) можно также ввести адрес в поле **Копия**. В это же поле можно поместить адрес, выбрав его в списке адресов в окне **Выбор имен** (см. рисунок 7) и нажав кнопку **Копия**.

Одно письмо можно направить сразу нескольким адресатам. Для этого надо в поле **Кому** окна сообщения (см. рисунок 6) последовательно ввести несколько адресов, разделяя их точкой с запятой. Можно также последовательно выбрать несколько имен в окне **Выбор имен** (см. рисунок 7).

При регулярной рассылке сообщений стабильному составу адресатов можно создать список рассылки. Список рассылки создается при работе с областью **Контакты** Microsoft Outlook.

Ошибочно введенные адреса можно удалить непосредственно в полях **Кому** и **Копия**. Редактировать адреса в этих полях нельзя.

После ввода адресов получателей следует указать тему сообщения. Тема указывается в поле **Тема** окна сообщения (см. рисунок 6). Тема должна отражать основное содержания отправляемого письма. Строго говоря, количество слов темы практически не ограничено, но следует ограничиваться четырьмя-пятью словами. После ввода темы сообщения эта же тема указывается в заголовке окна сообщения (рисунок 8).

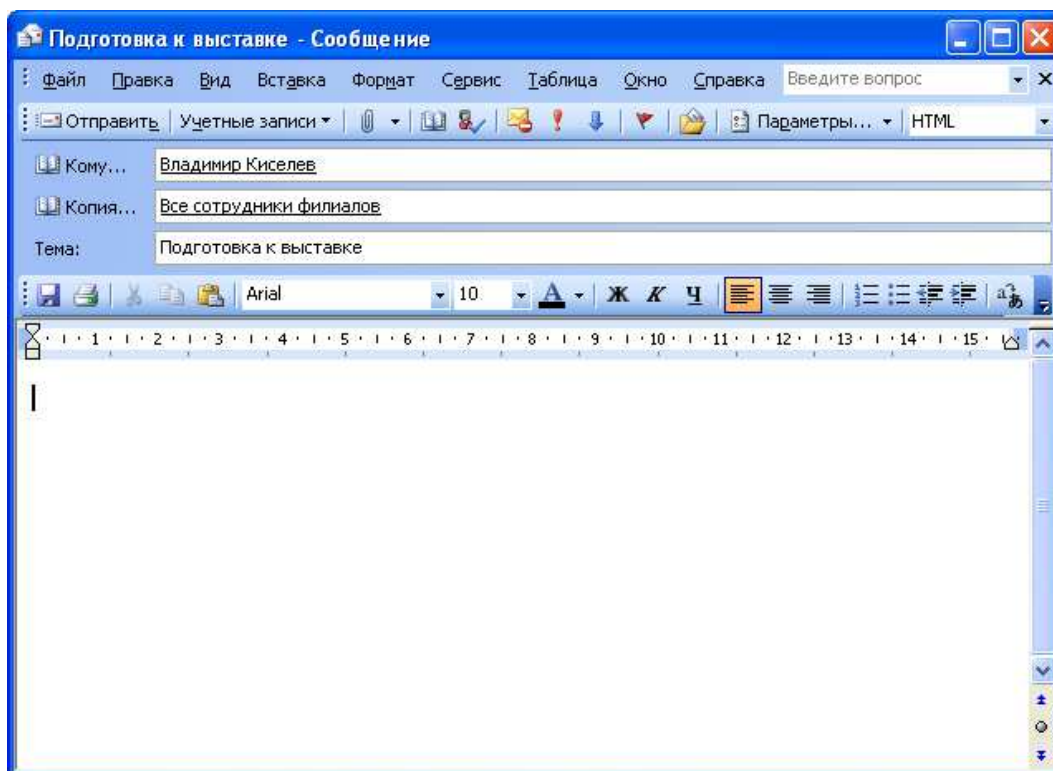


Рисунок 8 - Окно сообщения с выбранными адресами и темой сообщения

В основной части окна сообщения следует ввести текст письма. При желании и необходимости текст можно оформлять как обычный документ Microsoft Word. После ввода текста для отправки сообщения необходимо нажать кнопку **Отправить**.

2.2 Настройка параметров сообщения

В любой момент до отправки сообщения можно настроить его параметры. Для этого в окне сообщения следует нажать кнопку **Параметры**.

Настройка параметров производится в окне **Параметры сообщения** (рисунок 9).

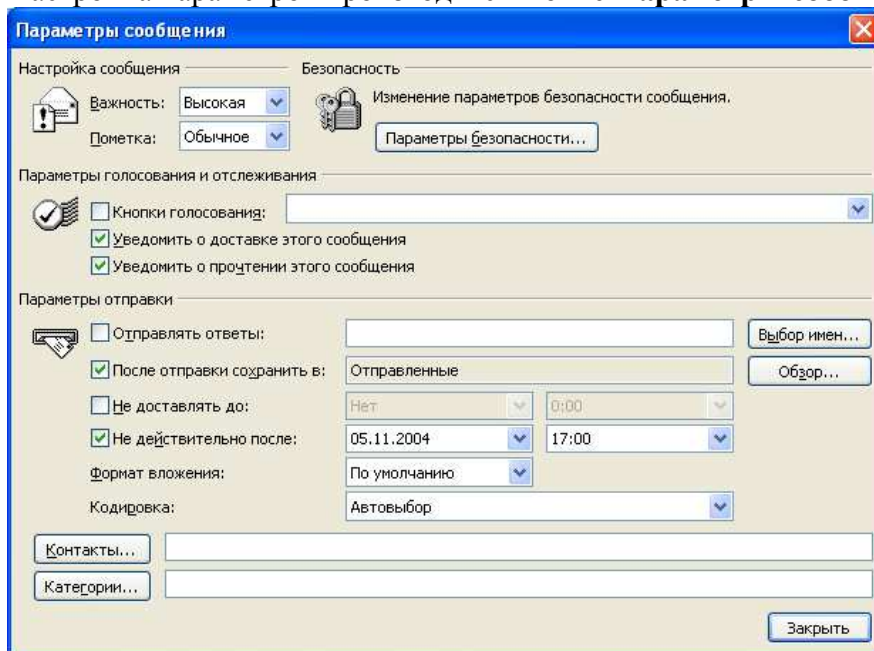


Рисунок 9 - Настройка параметров сообщения

В раскрывающемся списке **Важность** можно выбрать уровень важности письма. При выборе высокого уровня рядом с сообщением будет отображен восклицательный знак, что может привлечь более пристальное внимание получателя письма. Для того чтобы знать, получено и прочитано ли сообщение, можно установить флажки **Уведомить о доставке этого сообщения** и **Уведомить о прочтении этого сообщения**. При этом при доставке сообщения и его открытии получателем Вам будут направлены соответствующие уведомления. Можно ограничить срок отправки письма. Если установить флажок **Не действительно после** и указать соответствующую дату, то после этой даты сообщение у получателя в папке **Входящие** будет зачеркнуто. После установки всех параметров следует нажать кнопку **Заккрыть**.

2.3 Отправка документа Microsoft Word по электронной почте

Отправить документ Microsoft Word по электронной почте можно в виде текста сообщения электронной почты или в виде вложения к сообщению электронной почты.

В первом случае необходимо нажать кнопку **Конверт** панели инструментов **Стандартная**. После этого под панелями инструментов появится панель сообщения (рисунок 10).

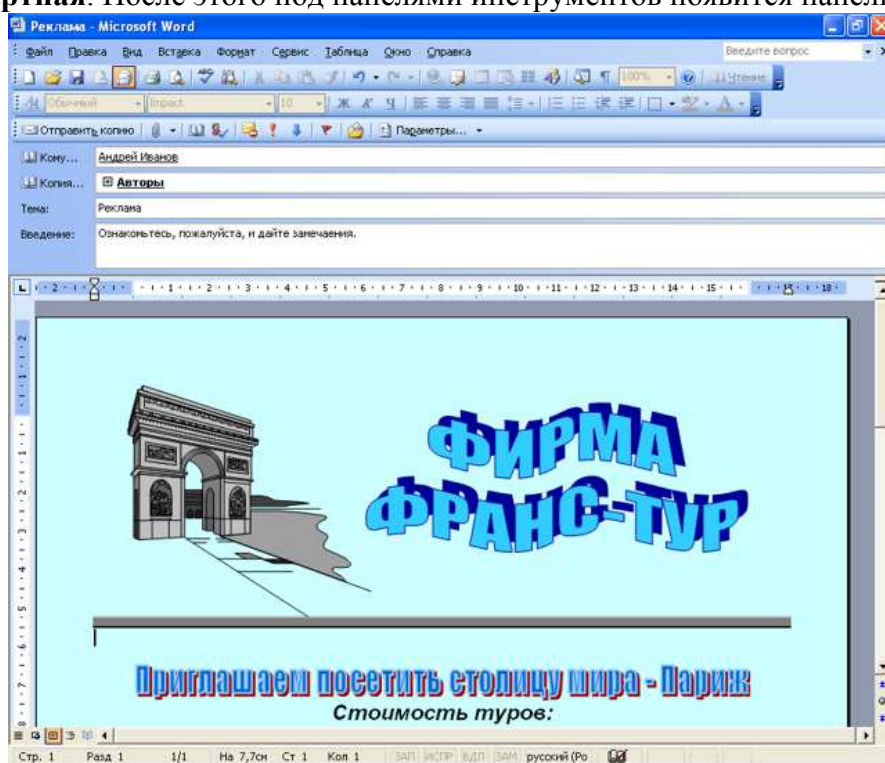


Рисунок 10 - Отправка документа Microsoft Word по электронной почте

В поле **Кому** следует ввести адрес получателя письма или несколько адресов через запятую. При необходимости можно ввести адрес в поле **Копия**. Если адреса получателей сообщения имеются в адресной книге, то для выбора адресов можно воспользоваться кнопками **Кому** и **Копия**. По умолчанию в поле **Тема** отображается имя документа. При желании можно ввести другую тему. Для отправки сообщения нажмите кнопку **Отправить копию**. При отправке документа в виде вложения к сообщению электронной почты следует выполнить команду **Файл/Отправить/Сообщение (как вложение)**. После этого запустится Microsoft Outlook (рисунок 11) или другое приложение, настроенное для работы с электронной почтой.

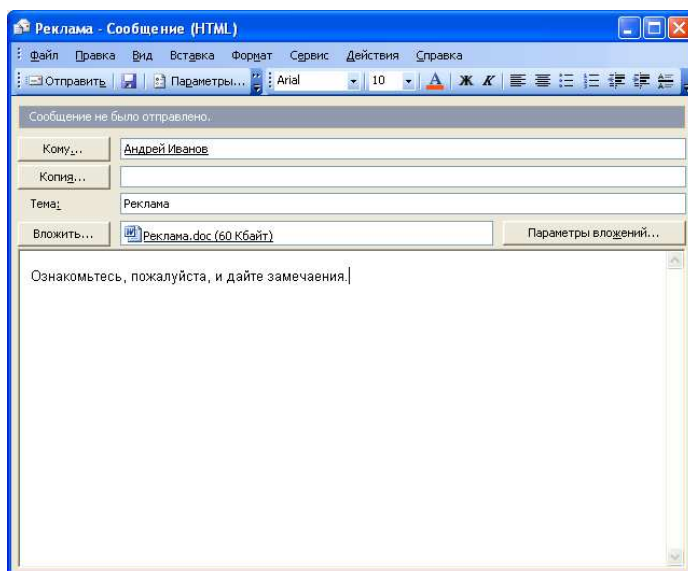


Рисунок 11 - Окно сообщения Microsoft Outlook

В поле **Кому** следует ввести адрес получателя письма или несколько адресов через запятую. При необходимости можно ввести адрес в поле **Копия**. Если адрес получателей сообщения имеются в адресной книге, то для выбора адресов можно воспользоваться кнопками **Кому** и **Копия**. По умолчанию в поле **Тема** отображается имя документа. При желании можно ввести другую тему. При необходимости в основном поле сообщения можно ввести сопроводительный текст. Для отправки сообщения нажмите кнопку **Отправить**.

3 Работа с поступившей почтой

3.1 Просмотр сообщения

Для просмотра сообщения можно воспользоваться областью чтения. Обычно она находится в нижней части окна, но может находиться и справа от списка сообщений (рисунок 12).

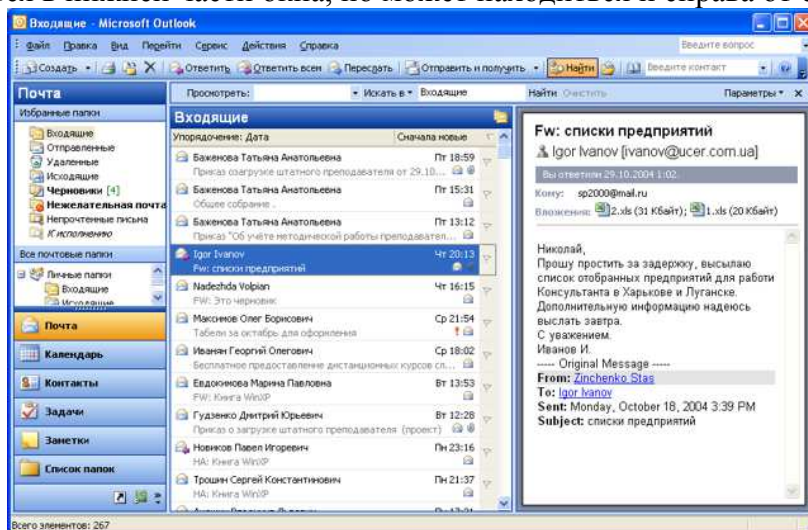


Рисунок 12 - Окно Microsoft Outlook, область "Почта". Область чтения расположена справа

Если область чтения не отображена, в меню **Вид** следует выбрать команду **Область чтения/Внизу** или **Область чтения/Справа**. В области просмотра сообщения отображается тема и имя отправителя сообщения. В виде значков отображены вложенные файлы. Длинное сообщение, как правило, не полностью отображается в области чтения. Для открытия поступившего сообщения в отдельном окне (рисунок 13) следует дважды щелкнуть мышью по

сообщению списке папки **Входящие**.

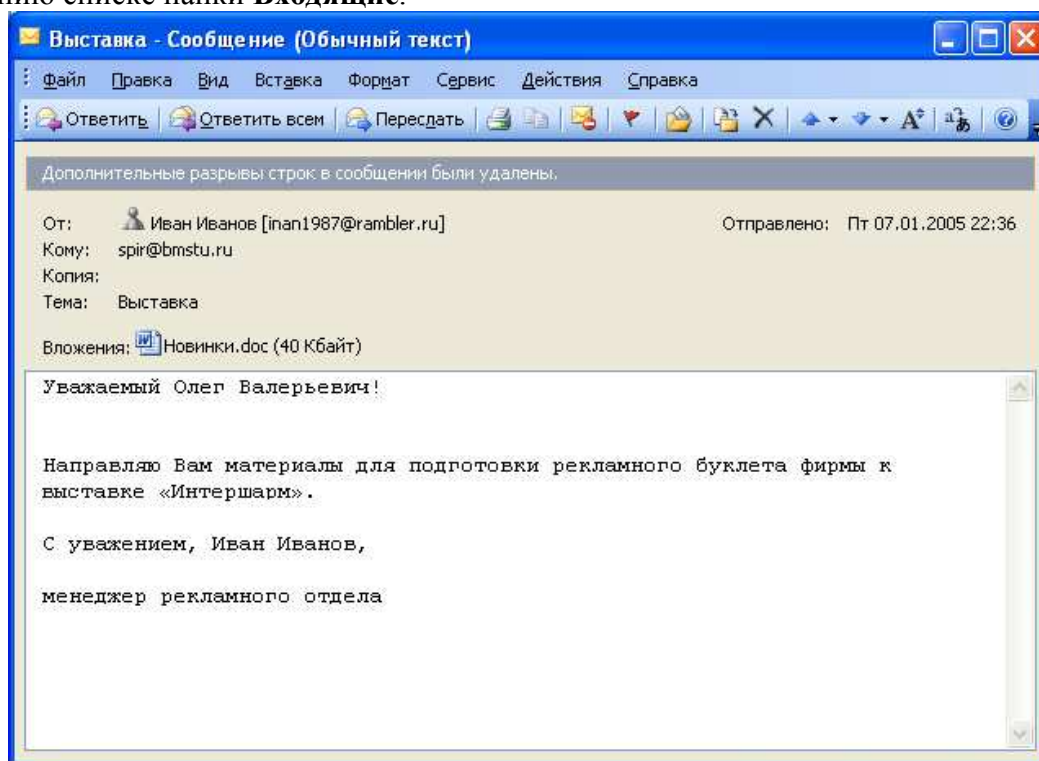


Рисунок 13 - Поступившее сообщение, открытое в отдельном окне

3.2 Создание ответа на полученное сообщение

Чтобы ответить на полученное сообщение, в области (модуле) **Почта** в папке **Входящие** следует выделить сообщение и нажать кнопку **Ответить** (рисунок 14).

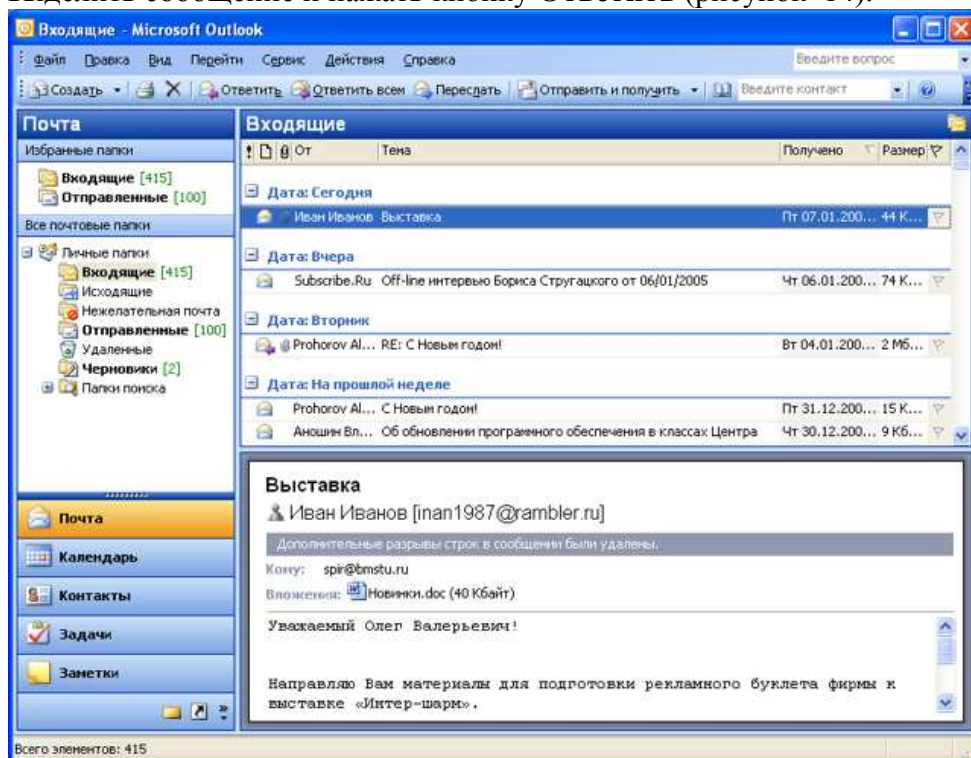


Рисунок 14 - Окно папки "Входящие" модуля "Почта" Microsoft Outlook

Кроме того, если сообщение открыто в отдельном окне (рисунок 15), также можно нажать кнопку **Ответить**.

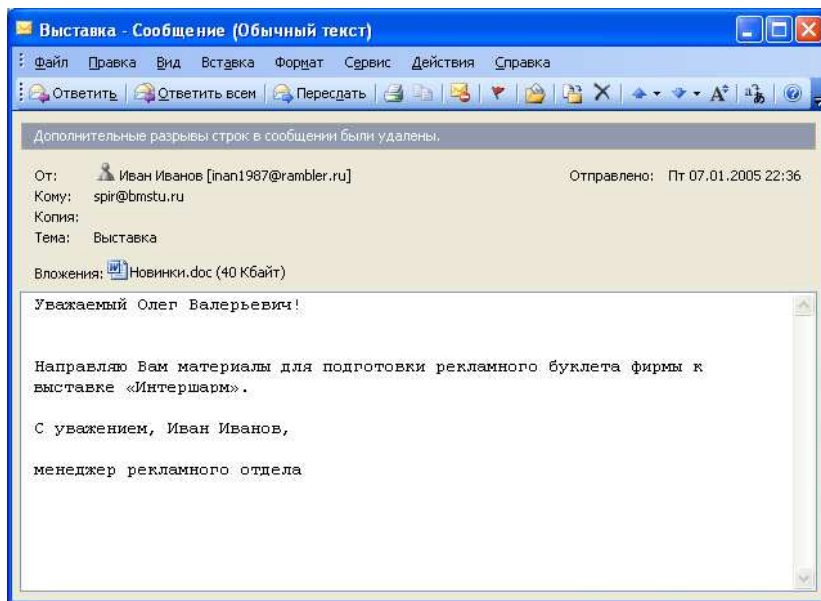


Рисунок 15 - Окно полученного сообщения

После этого будет открыто окно нового сообщения, в его поле **Кому** уже автоматически введен адрес отправителя письма, на которое создается ответ. Также указана и тема сообщения с добавлением в начале **RE:**. Если письмо, на которое создается ответ, было направлено нескольким адресатам, можно воспользоваться кнопкой **Ответить всем**. В этом случае в поле **Кому** будут автоматически введены имена всех получателей этого письма, а в поле **Копия** - адреса всех получателей копии этого письма. Текст ответа обычно вводится над чертой в верхней части окна сообщения. Текст вводится и оформляется обычным образом. При желании и необходимости в поле **Кому** можно добавить адреса других получателей письма. Адреса можно ввести с клавиатуры или нажать кнопку **Кому** и выбрать в окне **Выбор имен** (см. рисунок 16). Можно также внести изменения в тему сообщения, но обычно делать это не рекомендуется.

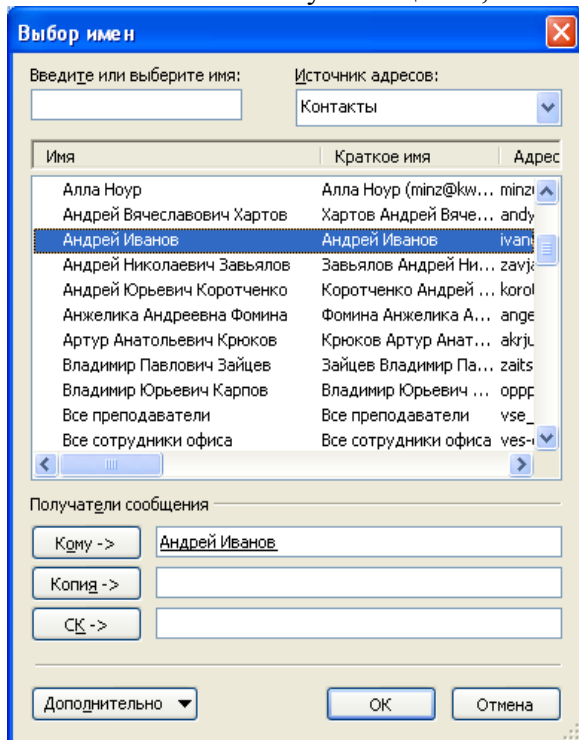


Рисунок 16 - Выбор адреса в окне "Выбор имен"

Подготовленное сообщение отправляют обычным порядком. После отправки ответа в папке **Входящие** у сообщения появляется соответствующий значок. А в области просмотра и в окне открытого сообщения соответствующая отметка об ответе на это сообщение.

3.3 Пересылка полученных сообщений

Полученное сообщение можно переслать любому адресату. Для этого в области (модуле) **Почта** в папке **Входящие** следует выделить сообщение и нажать кнопку **Переслать**. Кроме того, если сообщение открыто в отдельном окне, также можно нажать кнопку **Переслать**.

После этого будет открыто окно нового сообщения, в котором указана тема сообщения с добавлением в начале **FW:**. В отличие от процедуры создания ответа, поле **Кому** в данном случае остается пустым. К пересылаемому письму можно добавить собственный текст. Текст обычно вводится над чертой в верхней части окна сообщения. В поле **Кому** необходимо ввести адреса получателя или получателей письма. Адреса можно ввести с клавиатуры или нажать кнопку **Кому** и выбрать в окне **Выбор имен**. При желании и необходимости можно внести изменения в тему сообщения, но обычно делать это не рекомендуется.

3.4 Работа с прикрепленными файлами

Файлы, прикрепленные к поступившему сообщению, отображаются в виде значков в области чтения папки **Входящие** или в отдельном окне сообщения.

Для выполнения какого-либо действия с прикрепленным файлом можно щелкнуть по его значку правой кнопкой мыши и выбрать необходимую команду в контекстном меню.

Вложенные файлы могут содержать компьютерные вирусы, поэтому обычно прикрепленные файлы сначала сохраняют на компьютере. Для сохранения файла следует выбрать команду **Сохранить как**. В окне **Сохранение вложения** следует выбрать папку, в которую сохраняется вложенный файл. При желании и необходимости можно изменить имя файла. Расширение менять не рекомендуется. После сохранения файлы проверяют антивирусными программами и только после этого открывают. Если сообщение с прикрепленными файлами получено от надежного адресата, можно выбрать команду **Открыть**. При этом появится предупреждение (рисунок 17), в котором надо нажать кнопку **Открыть**.

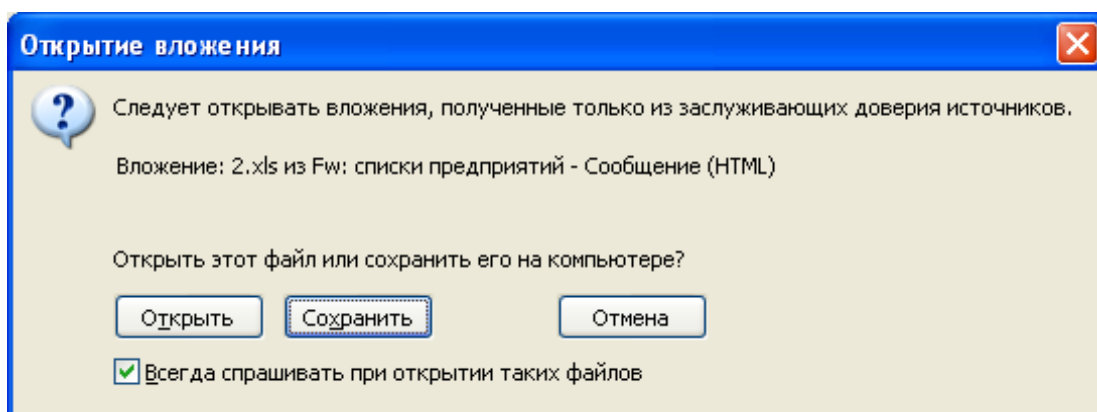


Рисунок 17 - Открытие вложенного файла

Но и в этом случае рекомендуется сначала сохранить файл. Вложенный файл можно удалить. Для этого следует в контекстном меню выбрать команду **Удалить**.

5.3 Варианты заданий по теме Интернет

- 1 Создайте учетную запись на свою фамилию (Пример: Ivanov@mail.ru).
- 2 Создайте сообщение, в которое поместите в заархивированном виде результаты выполнения лабораторной работы по разделу Access.

- 3 Отправьте сообщение своему соседу слева, а копию отправьте соседу справа.
- 4 Проверьте почту и убедитесь, что получили 2 сообщения.
- 5 Перешлите сообщение от своего соседа слева – соседу справа.

2.24 Лабораторная работа №26(2 часа).

Тема: «Защита информации в компьютерах и сетях»

2.24.1 Цель работы: Научиться работать с программами-архиваторами. Уметь использовать их для эффективного и безопасного хранения информации. Знакомство с антивирусными программами

2.24.2 Задачи работы:

1. Знакомство с архивированием данных
2. Знакомство с Антивирусом Касперского 7.0 (Kaspersky Antivirus Security 7.0)
3. Знакомство с антивирусным программным обеспечением avast!

2.24.3 Перечень приборов, материалов, используемых в лабораторной работе:

1. Персональный компьютер
2. Операционная система Microsoft Windows.
3. Microsoft Office

2.24.4 Описание (ход) работы:

1) Архивирование данных - это процесс сжатия файлов, с целью освобождения места на диске.

Очень часто случается так, что данные не помещаются на дискету или на компакт-диск, а после того как вы выполните архивирование данных, все прекрасно поместится.

Особенно хорошо сжимаются тестовые файлы, если повторов очень много, то сжатия можно добиться до 10 раз. Хуже сжимаются цветные графические файлы. Можно сказать, что в среднем архиваторы дают выигрыш в 2-3 раза.

Программа, которая сжимает текстовый файл, называется упаковщиком или архиватором. Программы-упаковщики архивируют не только текстовые файлы, а также программы, звуковые, графические, видеофайлы и другие.

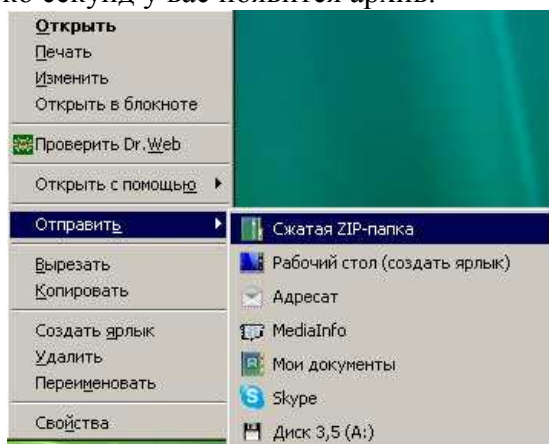
В процессе архивирования данных создается архивный файл, который меньше по объему сжимаемых файлов. После создания архива, сжимаемые файлы можно удалить, тем самым освобождая место на диске.

Если же вам снова понадобилось вернуть архивные файлы в первоначальное состояние, то можно распаковать архив, вернув тем самым файлы на прежнее место. Архив при этом можно удалить, чтобы просто не занимал лишнего места на диске.

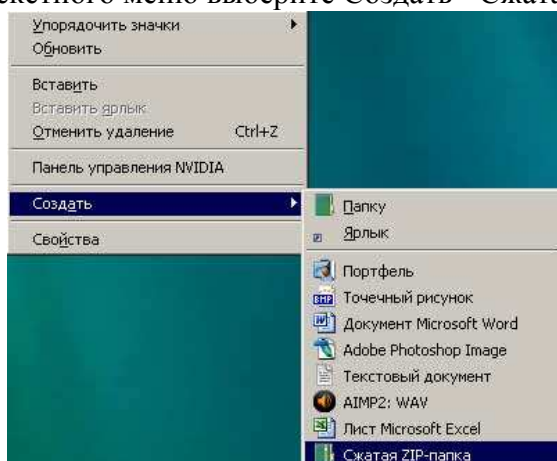
Существует достаточное количество архиваторов и столько же типов архивных файлов. Среди них самыми распространенными являются ZIP и RAR. (Здесь можно скачать winzip и winrar).

Если у вас нет на компьютере никакого архиватора, то можно воспользоваться встроенным архиватором Windows, который отвечает за работу с zip-архивами.

Попробуем создать архив. Для этого выделите файлы и папки, которые вы хотите заархивировать, щелкните правой кнопкой мыши по выделенному и из контекстного меню выберите команду Отправить, а там - Сжатая ZIP-папка, как показано на рисунке ниже. Через несколько секунд у вас появится архив.



Еще один способ архивирования данных - это создание пустого архива. Можете попрактиковаться: щелкните правой кнопкой мыши на пустое место в папке или на рабочем столе, из контекстного меню выберите Создать - Сжатая ZIP-папка.



Можете дать название новому архиву или оставить название по умолчанию, то есть то, что предлагает сама программа, имя такого архива будет кончаться на .zip. В пустой архив вы можете бросать файлы как в обычную папку.



Встроенный архиватор Windows не может защитить архив паролем, не может создавать самораспаковывающийся архив, не сможет большой архив порезать на части, чтобы разместить на дискетах или компакт-дисках.

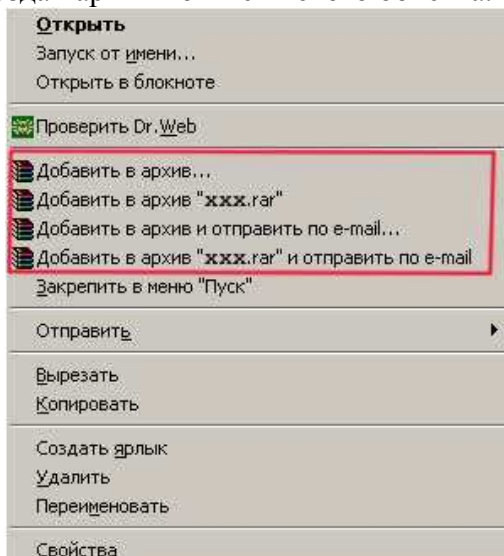
Поэтому, если вы хотите избавиться себя от этих ограничений, то вам нужно установить на вашем компьютере отдельную программу-архиватор.

7-ZIP - отличный архиватор - автор Игорь Павлов. Может создавать архивы, используемые в других операционных системах и извлекать файлы из всех архивных форматов.

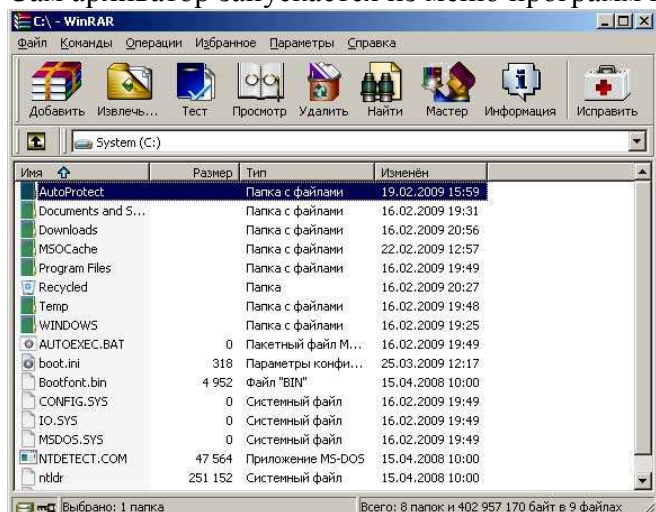
WinRAR - один из лучших в мире архиваторов - автор Евгений Рошал. Умеет создавать архивы как RAR та и ZIP.

После установки архиватора вы заметите, что в контекстном меню файлов появятся новые

пункты (в красной рамочке), вместо XXX будет стоять имя текущего архивируемого объекта и будет создан архив по имени этого объекта.



Сам архиватор запускается из меню программ и окно программы будет выглядеть так.

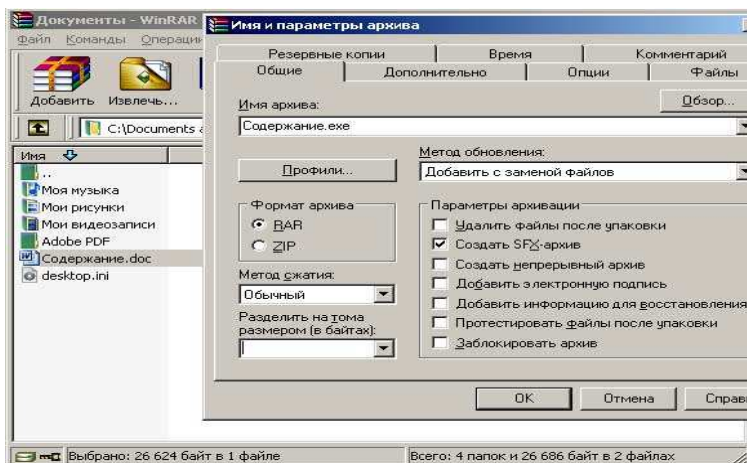


Самораспаковывающийся архив

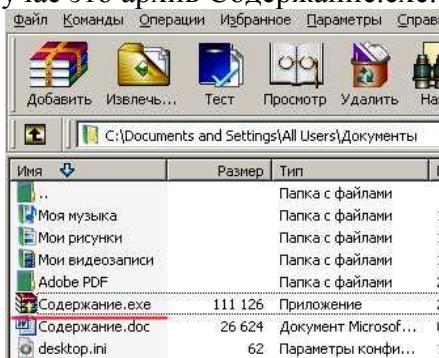
Самораспаковывающийся архив - это архив, который распаковывается без всякого архиватора, то есть самостоятельно и его имя кончается на .exe.

Такой архив можно создать, воспользовавшись окном программы WinRAR, либо используя контекстное меню архивируемого объекта.

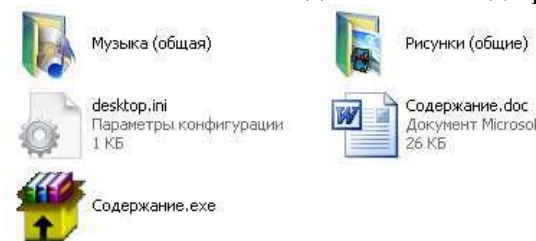
Итак, запустите программу WinRAR, выберите архивируемый объект, который вы хотите запаковать самораспаковывающимся архивом, щелкните на кнопку Добавить (стопка перевязанных книг) - откроется новое окно Имя и параметры архива и в Параметрах архивации поставьте галочку Создать SFX-архив, нажмите кнопку ОК.



У вас появится самораспаковывающийся архив xxx.exe, где xxx - имя вашего объекта. В моем случае это архив Содержание.exe.

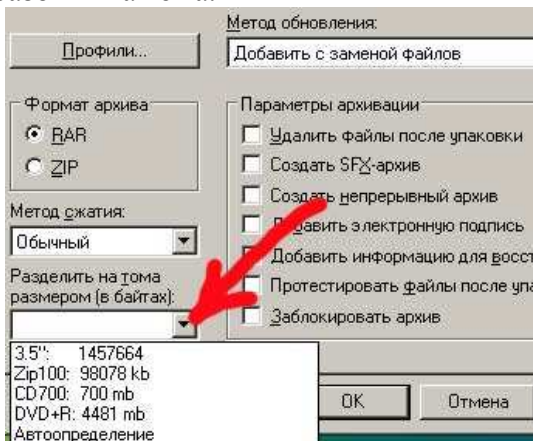


Если зайти в документы, то мы можем увидеть его в виде коробки, упакованной книгами, это зависит от того в каком виде в меню Вид представлены документы для обозрения.



Для распаковки такого архива достаточно запустить его двойным щелчком мыши и в появившемся окне нажать кнопку Извлечь.

Если у вас большой архив, который не умещается на дискету или на компакт-диск, то его можно разбить на тома.



Кроме того архиватор может сделать еще много полезных вещей, например, вы можете в RAR создать ZIP-архив, хотя RAR-архивы получаются меньшего размера, закрыть доступ к данным вашего архива, защитив его паролем, и многое другое.

2) Компьютерный вирус – это небольшая вредоносная программа, которая самостоятельно может создавать свои копии и внедрять их в программы (исполняемые файлы), документы, загрузочные сектора носителей данных. В зависимости от среды обитания основными типами компьютерных вирусов являются: программные, загрузочные, макровирусы, сетевые вирусы. Для выполнения этой лабораторной работы необходимо ознакомиться с [антивирусными программами и методами защиты от компьютерных вирусов](#), изложенными в лекционном курсе или других источниках информации.

Цель работы

Знакомство с антивирусными программами и приобретение навыков работы с [Антивирусом Касперского 7.0](#) или Kaspersky Antivirus Security 7.0 и [avast!](#) antivirus версия 4.8 Home Edition (проверка настроек антивирусов, сканирование файлов, папок и дисков, обновления антивирусной базы).

Постановка задачи

1 Знакомство с Антивирусом Касперского 7.0 (Kaspersky Antivirus Security 7.0) и антивирусным программным обеспечением avast!

2 Проверьте дату и способы обновления антивирусных баз.

3 Выполните сканирование дисков.

4 Выполните сканирование папок и файлов.

5 Сравните работу двух антивирусных программ.

Пошаговое выполнение работы

Ознакомьтесь с Антивирусом Касперского 7.0 (Kaspersky Antivirus Security 7.0)

1 Запустите Антивирус Касперского, щелкнув на кнопке Пуск и выбрав команду Программы/Антивирус Касперского 7.0 (Kaspersky Antivirus Security 7.0). В результате откроется главное окно Антивирус Касперского в режиме Защита (на вкладке Защита).

2 В окне Антивирус Касперского ознакомьтесь с компонентами, основными функциями программы и ее настройками.

3 Проверьте наличие обновлений (дату и способы обновления) антивирусной базы. Антивирус Касперского имеет встроенный планировщик автоматического обновления антивирусных баз, поэтому программа самостоятельно периодически проверяет наличие новых антивирусных баз на сервере Лаборатории Касперского.

Но если запустить автоматическое обновление антивирусных баз из интернета невозможно, то можно использовать антивирусные базы, выпускаемые в ZIP-архивах. Архивы с антивирусными базами регулярно выкладываются на сайты Лаборатории Касперского в каталог zips.

После скачивания распаковка архива *cumul.zip (полный набор антивирусных баз имеет имя av-i386-cumul.zip - это все обновления, вышедшие до последнего воскресенья) производится в отдельную папку. При настройке обновления в качестве источника обновлений указывается папка, с распакованным архивом *cumul.zip.

4 Обновите антивирусную базу. Для этого в главном окне щелкните на вкладке (кнопке) Обновление, в открывшемся справа фрейме щелкните на строке "Обновить базы". Антивирусная база будет обновлена.

4 Установите флоппи-диск с файлами и папками в накопитель для гибких магнитных дисков или USB flash drive (флешку) в USB-порт.

5 В главном окне Антивируса Касперского 7.0 на вкладке **Поиск вирусов** выполните сканирование сменных дисков и диска C: на наличие вирусов. Проверьте результаты сканирования.

6 Для сканирования папок или файлов щелкните правой кнопкой мыши на требуемую папку или файл и в контекстном меню выберите команду "Проверить на вирусы". Проверьте результаты сканирования.

Ознакомьтесь с антивирусным программным обеспечением avast! antivirus

1 Запустите антивирусную программу avast! antivirus из главного меню? откроется окно avast! antivirus.

2 В окне ознакомьтесь с основными элементами: меню (проверьте настройки антивируса); пиктограммами: выбор папки, сменные носители, локальные диски; кнопки вирусное хранилище, iavs (обновить антивирусную базу данных), Резидентный сканер.

3 Обновите антивирусную базу данных, щелкнув на пиктограмме iavs.

4 Установите один из сменных носителей информации.

5 Выполните сканирование сменного носителя информации, используя элементы управления окна avast! antivirus.

6 Выполните сканирование локальных дисков, используя элементы управления окна avast! antivirus.

7 Выполните сканирование папок, используя элементы управления окна avast! antivirus.

8 Для сканирования папок или файлов щелкните правой кнопкой мыши на требуемую папку или файл и в контекстном меню выберите команду "Сканировать_имя папки или файла".

9 Осуществите настройку резидентных провайдеров, щелкнув левой кнопкой мыши на кнопке А в панели индикации. Откроется Сканер доступа avast!, в котором осуществляется настройка 7 резидентных провайдеров или модулей.

Сравните работу двух антивирусных программ

На основании результатов выполненной работы с антивирусными программами дайте оценку этим программам.

Завершение работы

Сообщите преподавателю о выполненной работе. После разрешения на завершение работы можете приступить к сдаче тестов по выполненной работе.