

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний автомобільно-дорожній університет

До друку і в світ дозволяю
Заступник ректора

_____ І.П. Гладкий

Шевченко В.О.

КОНСПЕКТ ЛЕКЦІЙ

з дисципліни **"Інформатика та системологія"**
за напрямом підготовки 6.040106

Усі цитати, цифровий, фактичний
матеріал і бібліографічні відомості
перевірені, написання

одиниць відповідає стандартам

Затверджено
Радою Факультету МТЗ
протокол № ____ від

Укладач

В.О. Шевченко

Відповідальний за випуск

Харків 2016

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний автомобільно-дорожній університет

Шевченко В.О.

КОНСПЕКТ ЛЕКЦІЙ

з дисципліни **"Інформатика та системологія"**
за напрямом підготовки 6.040106

Харків 2016

Содержание

Лекция 1. Введение в дисциплину «Информатика и системология»	4
Лекция 2. Архитектура ПК. Назначение и технические характеристики устройств ПК	13
Лекция 3. Программное обеспечение ПК. Файловая система.....	26
Лекция 4. ОС Microsoft Windows. Типы окон	29
Лекция 5. Борьба с вирусами. Архивация файлов	35
Лекция 6. Табличный процессор Microsoft Excel	41
Лекция 7. Формулы и функции в Excel.....	47
Лекция 8. Построение графиков и диаграмм	55
Лекция 9. Алгоритмизация. Среда VB	59
Лекция 10. Основные компоненты языка Visual Basic. Линейные вычислительные процессы	65
Лекция 11. Разветвляющиеся вычислительные процессы	72
Лекция 12. Циклические вычислительные процессы.....	76
Лекция 13. Вычислительные процессы с массивами данных.....	80
Литература	85

Лекция 1.

ВВЕДЕНИЕ В ДИСЦИПЛИНУ «ИНФОРМАТИКА И СИСТЕМОЛОГИЯ»

Основные понятия и определения

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

Предмет информатики составляют следующие понятия:

- аппаратное обеспечение средств вычислительной техники;
- программное обеспечение вычислительной техники;
- средства взаимодействия аппаратного и программного обеспечения;
- средства взаимодействия человека с аппаратными и программными средствами.

Информатика занимается исследованием информации, ее свойств, критериев и структур в естественных и искусственных информационных коммуникациях, предусматривает изучение принципов, моделей, алгоритмов хранения, преобразования, анализа и синтеза информации, а также их программную реализацию.

Информация (от лат. *Informatio* – осведомление, разъяснение, изложение) в широком смысле – абстрактное понятие, имеющее множество значений в зависимости от контекста.

В настоящее время не существует единого определения термина «информация». В практическом смысле информация – это:

1. Сведения об окружающем мире и протекающих в нем процессах.
2. Сообщения, осведомляющие о положении дел, о состоянии чего-либо.

(Ожегов С.И. Словарь русского языка. 22-е изд., 1990.)

Качество информации определяется степенью ее соответствия нуждам потребителей и характеризуется следующими свойствами:

– *объективность* информации характеризует ее независимость от чьего-либо мнения или сознания, а также от методов получения. Более объективна та информация, в которую методы получения и обработки вносят меньший элемент субъективности;

– *полнота* – информацию можно считать полной, когда она содержит минимальный, но достаточный для принятия правильного решения набор показателей. Как неполная, так и избыточная информация снижает эффективность принимаемых на ее основании решений;

– *достоверность* – свойство информации быть правильно воспринятой. Объективная информация всегда достоверна, но достоверная информация может быть как объективной, так и субъективной;

– *адекватность* – степень соответствия реальному объективному состоянию дела. Неадекватная информация может образоваться при создании новой информации на основе неполных или недостоверных данных;

– *доступность* информации – мера возможности получить ту или иную информацию;

– *актуальность* информации – это степень соответствия информации текущему моменту времени;

– *эмоциональность* – свойство информации вызывать различные эмоции у людей. Это свойство информации используют производители медиаинформации. Чем сильнее вызываемые эмоции, тем больше вероятности обращения внимания и запоминания информации.

Информация на пути от источника к потребителю проходит через ряд преобразователей – кодирующие и декодирующие устройства, вычислительную технику, обрабатывающую информацию по определенному алгоритму, и т. д, при этом понятие «информация» заменяется более общим понятием «данные».

Данными называется информация, представленная в удобном для обработки виде. Преобразование и обработка данных позволяет получить новую информацию.

Обработка информации включает в себя множество операций с данными:

– сбор – накопление информации с целью обеспечения достаточной полноты для принятия решений;

– формализация – приведение данных, поступающих из разных источников, к одинаковой форме, для того чтобы сделать их сопоставимыми между собой;

– фильтрация – отсеивание избыточных данных, в которых нет необходимости для принятия решений;

– сортировка – упорядочение данных по заданному признаку с целью удобства использования;

– архивация – организация хранения данных в сжатой форме;

– защита – комплекс мер, направленных на предотвращение утраты, несанкционированного воспроизведения и модификации данных;

– транспортировка – прием и передача (доставка и поставка) данных между удаленными участниками информационного процесса; при этом источник данных в информатике принято называть сервером, а потребителя – клиентом;

– преобразование – перевод данных из одной формы в другую или из одной структуры в другую. Преобразование данных часто связано с изменением типа носителя, например, книги можно хранить в обычной бумажной форме, но можно использовать для этого и электронную форму.

Для выполнения разных действий с информацией необходимым является знание ее количества. Наименьшей единицей измерения информации при ее представлении является **бит**, который может принимать два значения: 0 или 1. Двоичная система измерения информации обусловлена тем, что электронные элементы, как правило, могут иметь два устойчивых состояния, например, "включено – есть сигнал", "выключено – нет сигнала" и т. д.

Для хранения, или передачи одного символа используют 8 бит. Так как каждый бит может быть представлен 0 или 1, то количество комбинаций из двух символов по 8 равняется $2^8 = 256$. Таким образом, с помощью 8 бит можно закодировать 256 разных символов, что достаточно для представления нескольких алфавитов, служебных символов и арабских цифр. Восемь бит, которые предназначены для хранения одного символа, приняли за основную единицу измерения ин-

формации и назвали ее **байт**. Информация иногда может быть представлена в миллионах, или миллиардах байт, поэтому были введены более "весомые" единицы измерения информации:

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

1 мегабайт (Мбайт) = 2^{10} Кбайт = 1024 Кбайт;

1 гигабайт (Гбайт) = 2^{10} Мбайт = 1024 Мбайт;

1 терабайт (Тбайт) = 2^{10} Гбайт = 1024 Гбайт;

1 петабайт = 2^{10} Тбайт = 1024 терабайта;

1 эксабайт = 1024 петабайта;

1 зеттабайт = 1024 эксабайта;

1 йоттабайт = 1024 зеттабайта.

Системология – наука о системах и системной организации процессов и явлений природы, науки, техники, общественных формаций, функциональных образований и структур. Системология обобщает различные данные о системах и выводит основные закономерности их возникновения, развития и преобразования. Системология трактует объекты как системы, содержащие структуру, и явления как системы с многоуровневой, сложной организацией взаимодействий и отношений, включая внутренние и внешние связи.

Система (*systema*) в переводе с греческого языка есть целое, составленное из частей, соединение. На понятийном уровне систему образует «совокупность взаимно связанных объектов». Отдельная система живет во взаимодействии с другими системами, и все их множество образует мировую систему. В мировую систему входят системы живой и неживой природы и искусственные системы, создаваемые человеком

Системологический подход в информатике исходит из понимания сложных систем как принципиально неполных, неточных и противоречивых в смысле получаемой информации о них. Причиной такого положения является тот факт, что в современной информатике наряду со строгими логическими, математическими, рациональными понятиями существует значительный объем данных, базирующихся на индивидуальных мнениях, коллективных идеях. При этом нестрогие, приближенные данные, нечеткие семантические знания и неформальные методы по их добыванию порой оказываются решающими при принятии решений.

История развития ВТ

История вычислений уходит глубокими корнями вдаль веков, также как и история развития человечества. Накопление запасов, дележ добычи, обмен – все подобные действия связаны со счетом. Для подсчета люди использовали собственные пальцы, камешки, палочки и узелки. Потребность в поиске решений все более и более сложных задач и, как следствие, все более сложных и длительных вычислений поставила человека перед необходимостью находить способы, изобретать приспособления, которые могли бы ему в этом помочь. Исторически сложилось так, что в разных странах возникли собственные денежные единицы, меры веса, длины, объемов и расстояний. Для перевода значений из одной системы измерения в другую требовались вычисления, которые чаще всего могли произво-

дять специально обученные люди – их иногда даже приглашали из других стран. Это, естественно, привело к созданию изобретений, помогающих счету. Такие счетные устройства использовались десятилетиями, став основным техническим средством облегчения человеческого труда.

В истории развития вычислительных средств исследователи выделяют следующие этапы: ручной, механический, электромеханический, электронный.

Ручной этап

Самым первым инструментом счета у древнего пещерного человека в верхнем палеолите, безусловно, были пальцы рук. Сама природа предоставила человеку этот универсальный счетный инструмент. У многих народов пальцы (или их суставы) при любых торговых операциях выполняли роль первого счетного устройства. Для большинства бытовых потребностей людей их помощи вполне хватало.

К счету по пальцам рук восходят многие системы счисления, например пятеричная (одна рука), десятеричная (две руки), двадцатеричная (пальцы рук и ног), сорокаичная (суммарное число пальцев рук и ног у покупателя и продавца). У многих народов пальцы рук долгое время оставались инструментом счета и на наиболее высоких ступенях развития. Известные средневековые математики рекомендовали в качестве вспомогательного средства именно пальцевый счет, допускающий довольно эффективные системы счета.

В древности использовались примитивные счетные приспособления:

- узелковое письмо, иногда с вплетенными камешками и нитями разного цвета (например, красная – число воинов, желтая – золото), длины и толщины;
- зарубки на костяных и деревянных дощечках.

Механический этап

Одним из первых устройств (V-VI вв. до н. э.), облегчающих вычисления, можно считать специальную доску для вычислений, названную «абак».

Абак (греч. *abax*, *abakion*, лат. *abacus* – доска, счетная доска) – счетная доска, применявшаяся для арифметических вычислений в Древней Греции, Риме, затем в Западной Европе до XVIII в. Такую доску изготавливали из глины, бронзы, камня или слоновой кости, вычисления на ней производились путем перемещения камешков, палочек или костей в углублениях. Со временем эти доски стали расчерчивать на несколько рядов и колонок. Греции абак существовал уже в V в. до н.э., у японцев он назывался «соробан», у китайцев – «суаньпань».

В Древней Руси при счете применялось устройство, похожее на абак и называемое «русский шот». В XVII в. это устройство уже обрело вид привычных русских счетов.

Своего рода модификацию абака предложил Леонардо да Винчи (1452-1519) в конце XV – начале XVI в. Он создал эскиз 13-разрядного суммирующего устройства с десятизубными колесами. Чертежи данного устройства были найдены среди двухтомного собрания Леонардо по механике, известного как "Codex Madrid". Это устройство – своеобразная счетная машина; ее основу составляют

стержни, на одном конце которых расположено меньшее колесо, на другом – большее; все стержни должны располагаться таким образом, чтобы меньшее колесо на одном стержне касалось большего на другом. Десять оборотов первого колеса должны приводить к одному полному обороту второго, десять оборотов второго – к одному полному обороту третьего и т.д.

Первая механическая машина была описана в 1623 г. профессором математики Тюбингенского университета В. Шиккардом, реализована в единственном экземпляре и предназначалась для выполнения четырех арифметических операций над 6-разрядными числами. Машина Шиккарда состояла из трех независимых устройств: суммирующего, множительного и записи чисел. Сложение производилось последовательным вводом слагаемых посредством наборных дисков, а вычитание – последовательным вводом уменьшаемого и вычитаемого. Для выполнения операции умножения использовалась идея умножения решеткой. Использованная принципиальная схема машины Шиккарда стала классической – она (или ее модификации) использовалась в большинстве последующих механических счетных машин вплоть до замены механических деталей электромагнитными. Хотя из-за недостаточной известности машина Шиккарда и принципы ее работы не оказали существенного влияния на дальнейшее развитие счетных устройств, она по праву открывает эру механической вычислительной техники.

В начале XVII столетия, когда математика стала играть ключевую роль в науке, все острее ощущалась необходимость в изобретении счетной машины. И в 1642 г. 19-летний французский математик и физик Блез Паскаль создал «суммирующую» машину, названную «Паскалиной», которая кроме сложения выполняла и вычитание. Паскаль сделал это, чтобы помочь своему отцу, который был налоговым сборщиком. Машина Паскаля представляла собой механическое устройство в виде ящичка с многочисленными, связанными одна с другой шестеренками. Складываемые числа вводились в машину при помощи соответствующего поворота наборных колесиков. На каждое из этих колесиков, соответствовавших одному десятичному разряду числа, были нанесены деления от 0 до 9. При вводе числа колесики прокручивались до соответствующей цифры.

В 1670-1680 гг. немецкий математик Вильгельм Готфрид Лейбниц (1646-1716) изобрел первый арифмометр, который мог выполнять все четыре арифметических действия. Арифмометр (от греч. *arithmos* – число, счет и *metro* – мера, измеритель) – настольная (портативная) механическая вычислительная машина, предназначенная для точного умножения и деления, а также для сложения и вычитания. Числа вводятся в арифмометр, преобразуются и передаются пользователю (выводятся в окна счетчиков или печатаются на ленте) с использованием только механических устройств. Главной частью арифмометра был так называемый ступенчатый валик – цилиндр с зубцами разной длины, которые взаимодействовали со счетным колесом. Передвигая колесо вдоль валика, можно было ввести его в зацепление с необходимым числом зубцов и обеспечить установку определенной цифры. В 1666 году Лейбницем была предложена также двоичная система счисления, но в то время она не могла быть использована.

В 1878 г. русский ученый Пафнутий Львович Чебышев предложил счетную машину, выполнявшую сложение и вычитание многозначных чисел. Наибольшую популярность получил арифмометр, сконструированный петербургским инженером Вильгодтом Теофилом Однером. Конструкция прибора оказалась весьма удачной, так как позволяла довольно быстро выполнять все четыре арифметических действия.

Электромеханический этап

К наиболее ранним прообразам современных цифровых электронно-вычислительных машин (ЭВМ) относится аналитическая машина английского математика Ч. Бэббиджа. В 1-й половине XIX в. он разработал проект машины для автоматического решения задач, в котором гениально предвосхитил идею современных кибернетических машин. Машина а содержала арифметическое устройство (мельницу) и память для хранения чисел (склад), т.е. основные элементы современных ЭВМ.

Классическим типом средств электромеханического этапа был счетно-аналитический комплекс, предназначенный для обработки информации на перфокарточных носителях. Первый такой комплекс был создан в США Герман Холлеритом в 1887 г. и состоял из ручного перфоратора, сортировочной машины и табулятора. Он предназначался для обработки результатов переписи населения.

Выдающийся английский математик Алан Тьюринг в 1936 году мысленно сконструировал абстрактную машину, представляющую вычислительную машину общего вида для формализации понятия алгоритма. Машина Тьюринга способна имитировать (при наличии соответствующей программы) любую машину, действие которой заключается в переходе от одного дискретного состояния к другому.

В состав машины Тьюринга входит бесконечная в обе стороны лента, разделённая на ячейки, и управляющее устройство с конечным числом состояний.

Управляющее устройство может перемещаться влево и вправо по ленте, читать и записывать в ячейки символы некоторого конечного алфавита. Выделяется особый пустой символ, заполняющий все клетки ленты, кроме тех из них (конечного числа), на которых записаны входные данные.

В управляющем устройстве содержится таблица переходов, которая представляет алгоритм, реализуемый данной машиной Тьюринга. Каждое правило из таблицы предписывает машине, в зависимости от текущего состояния и наблюдаемого в текущей клетке символа, записать в эту клетку новый символ, перейти в новое состояние и переместиться на одну клетку влево или вправо. Некоторые состояния машины Тьюринга могут быть помечены как терминальные, и переход в любое из них означает конец работы, остановку алгоритма.

Компьютер Mark I (Automatic Sequence Controlled Calculator – вычислитель, управляемый автоматическими последовательностями) – первый американский программируемый компьютер. Разработан и собран в 1941 г. по контракту с IBM молодым гарвардским математиком Говардом Эйкеном и другими инженерами этой компании на основе идей Ч. Бэббиджа. После успешного прохождения пер-

вых тестов в феврале 1944 г. компьютер был перенесен в Гарвардский университет и формально запущен там. Компьютер содержал около 765 тыс. деталей (электромеханических реле, переключателей и т. п.), достигал в длину почти 17 м, в высоту – более 2,5 м и весил около 4,5 т. Общая протяженность соединительных проводов составляла почти 800 км. Основные вычислительные модули синхронизировались механически при помощи 15-метрового вала, приводимого в движение электрическим двигателем мощностью 5 л. с. (4 кВт). Фактически Mark I представлял собой усовершенствованный арифмометр, заменявший труд примерно 20 операторов с обычными ручными устройствами, однако из-за наличия возможности программирования некоторые исследователи называют его первым реально работавшим компьютером.

Первый автоматический программируемый универсальный цифровой компьютер Z3 (разработчик – доктор Конрад Цузе) создавался с 1939 по 1941 г. Z3 продолжил берлинские разработки К. Цузе Z1 и Z2. Он управлялся перфолентой из использованной киноплёнки, а ввод и вывод производился с четырехкнопочной цифровой клавиатуры и ламповой панели. Машина была основана на релетехнологии и требовала приблизительно 2600 реле: 1400 – для памяти, 600 – для арифметического модуля и оставшиеся – как часть схем управления. Общая стоимость материалов составила в то время приблизительно 6500 долларов. Единственная модель Z3 была разрушена во время воздушного налета в 1944 г. Z3 – первое устройство которое можно назвать полностью сформировавшимся компьютером с автоматическим контролем над операциями.

Электронный этап. Поколения ЭВМ

Развитие вычислительной техники представляют как изменение поколений, каждое из которых имеет свои особенности и отличия.

Первое поколение (1940 – 1950).

Электронно-вычислительные машины (ЭВМ) первого поколения использовали ламповую элементную базу, обладали малыми быстродействием и объемом памяти имели неразвитые операционные системы, языки низкого уровня.

В начале 40-х годов Д. Атанасов (Болгария) и Дж. Моучли (США) предложили использовать в качестве элементной базы радиоэлектронные лампы для построения электронно-вычислительной машины (ЭВМ).

В 1946 г. была построена первая ЭВМ на радиолампах в США в Пенсильванском университете под руководством Джона Уильяма Моучли и Джона Преспера Эккерта, назвали ее ENIAC (Electronic Numerical Integrator and Computer). Она имела следующие параметры: 18 000 электронных ламп; занимала 135 м², весила 30 т, потребляла 150 кВт электроэнергии, тактовая частота выполнения операций – 100 кГц.

В 1949 г. был разработан первый большой полнофункциональный электронный цифровой компьютер с сохраняемой программой – EDSAC (Electronic Delay Storage Automatic Computer). Разработан Морисом Уилксом и сотрудниками математической лаборатории Кембриджского университета (Англия).

В 1949 году в СССР, в городе Киеве в Институте электротехники АН УССР под руководством Сергея Алексеевича Лебедева построена первая ЭВМ, ее назвали МЭСМ (малая электронная счетная машина). МЭСМ была расположена в зале площадью 60 кв. м. Общее количество электронных ламп – около 3500 триодов и около 2500 диодов, в том числе 2500 триодов и 1500 диодов в запоминающем устройстве. Суммарная потребляемая мощность – около 25 кВт.

В 1952 году в г. Москва построена БЭСМ (большая электронная счетная машина).

Второе поколение (конец 1950 – начало 1970-х годов).

Электронно-вычислительные машины второго поколения имели полупроводниковую элементную базу, изменяемый состав внешних устройств, языки программирования высокого уровня и принцип библиотечных программ.

Второе поколение начинается с ЭВМ RCA 501, появившейся в 1959 г. в США и созданной на полупроводниковой элементной базе. Новая элементная технология позволила резко повысить надежность вычислительной техники, уменьшить ее габариты и потребляемую мощность, а также значительно увеличить производительность. Это дало возможность создавать ЭВМ с большими логическими возможностями и производительностью, что способствовало распространению сферы применения ЭВМ на решение задач планово-экономических, управления производственными процессами и др. В рамках второго поколения все более четко проявляется дифференциация ЭВМ на малые, средние и большие.

В СССР второе поколение представляют такие известные серии ЭВМ, как "Наири", МИР (малые ЭВМ); "Минск", "Урал", "Раздан", М-220, БЭСМ-4 (средние ЭВМ) и "Днепр", М-4000 (управляющие ЭВМ). Наилучшей отечественной ЭВМ второго поколения по праву считается модель БЭСМ-6, созданная в 1966 г., имеющая основную и промежуточную (на магнитных барабанах) память объемом соответственно 128 и 512 Кб, быстродействие порядка 1 млн. операций в секунду и довольно обширную периферию (магнитные ленты и диски, графопостроители, разнообразные устройства ввода-вывода).

Наиболее же массовыми советскими ЭВМ второго поколения были модели «Минск-22» и «Минск-32», хорошо себя зарекомендовавшие в эксплуатации при решении широкого круга задач. По ряду архитектурных решений БЭСМ-6 и «Минск-32» можно отнести к моделям, промежуточным между вторым и третьим поколениями ЭВМ.

Третье поколение (1970-е – начало 1980-х годов).

Электронно-вычислительные машины третьего поколения использовали интегральные схемы, имели развитую конфигурацию внешних устройств и стандартизованные средства сопряжения, обладали большим быстродействием и значительными объемами основной и внешней памяти. Развитая операционная система обеспечивала работу в так называемом мультипрограммном режиме. К таким компьютерам относится IBM 360. Интегральная схема, которую также называют

кристаллом, представляет собой миниатюрную электронную схему, вытравленную на поверхности кремниевого кристалла.

Данный этап – переход к интегральной элементной базе и создание много-машинных систем. Расширение функциональных возможностей ЭВМ увеличило сферу их применения, что вызвало рост объема обрабатываемой информации и поставило задачу хранения данных в специальных базах данных и их ведения. Так появились первые системы управления базами данных – СУБД.

Изменились формы использования ЭВМ: введение удаленных терминалов (дисплеев) позволило широко и эффективно внедрить режим разделения времени и за счет этого приблизить ЭВМ к пользователю и расширить круг решаемых задач.

Обеспечить режим разделения времени позволил новый вид ОС, поддерживающих мультипрограммирование. Мультипрограммирование – это способ организации вычислительного процесса, при котором на одном процессоре попеременно выполняются несколько программ. Пока одна программа выполняет операцию ввода-вывода, процессор не простаивает, как это происходило при последовательном выполнении программ (однопрограммный режим), а выполняет другую программу (многопрограммный режим). При этом каждая программа загружается в свой участок внутренней памяти, называемый разделом. Мультипрограммирование нацелено на создание для каждого отдельного пользователя иллюзии единичного использования вычислительной машины в интерактивном режиме. В Советском Союзе это была ЕС ЭВМ (Единая система ЭВМ): ЕС-1010, ЕС-1020, ЕС-1030, ЕС-1040, ЕС-1060. В разработке этой серии участвовали Болгария, Венгрия, Чехия. Начался выпуск советских ЭВМ: МИР-31, МИР-32, АСВТ М-6000, АСВТ М-7000. Выпускались также более компактные ЭВМ: "Электроника-79", «Электроника-100», "Электроника-125", "Электроника-200". ЕС-1010 имеет быстродействие 10 тыс. операций в секунду. Быстродействие ЕС-1020 – 20 тыс. операций в секунду, ОЗУ – 64 Кб, внешняя память на магнитных лентах и дисках.

Четвертое поколение (2-я половина 1980-х гг. – 1-я половина 1990-х гг.).

Электронно-вычислительные машины четвертого поколения используют большие и сверхбольшие интегральные схемы (БИС и СБИС), виртуальную память, многопроцессорный принцип построения с параллельным выполнением операций, развитые средства диалога.

Развитие ЭВМ четвертого поколения пошло по двум направлениям:

1. Создание супер-ЭВМ – комплексов многопроцессорных машин. Быстродействие таких машин достигает нескольких миллиардов операций в секунду. Они способны обрабатывать огромные массивы информации. Сюда входят комплексы ILLIAS 4, Cray, Cyber, "Эльбрус-1", "Эльбрус-2" и др. Многопроцессорные вычислительные комплексы (МВК) "Эльбрус-2" активно использовались в Советском Союзе в областях, требующих большого объема вычислений. Вычислительные комплексы "Эльбрус-2" эксплуатировались в Центре управления космическими полетами, в ядерных исследовательских центрах.

2. Дальнейшее развитие на базе БИС и СБИС микро-ЭВМ и персональных ЭВМ (ПЭВМ). Первыми представителями этих машин являются Apple, IBM PC (XT, AT, PS/2), "Искра", "Электроника", "Мазовия", "Агат", ЕС-1840, ЕС-1841, ЕС-5017 и др.

Пятое поколение (конец 90-х годов).

Переход к компьютерам пятого поколения предполагает использование новых архитектур, ориентированных на создание искусственного интеллекта.

Основные требования к компьютерам пятого поколения:

– создание развитого человеко-машинного интерфейса (распознавание речи, образов);

– развитие логического программирования для создания баз знаний и систем искусственного интеллекта;

– создание новых технологий в производстве вычислительной техники;

– создание новых архитектур компьютеров и вычислительных комплексов.

Электронно-вычислительные машины пятого поколения характеризуются наряду с использованием более мощных СБИС применением принципа "управление потоками данных", новыми решениями в архитектуре вычислительной системы и использованием принципов искусственного интеллекта.

Лекция 2

АРХИТЕКТУРА ПК. НАЗНАЧЕНИЕ И ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ УСТРОЙСТВ ПК

Персональный компьютер – это электронное устройство, предназначенное для автоматизации создания, хранения, обработки и передачи данных (информации).

Архитектура персонального компьютера – это компоновка его основных частей, таких как процессор, ОЗУ, видеоподсистема, дисковая система, периферийные устройства и устройства ввода-вывода.

Классическая архитектура (архитектура Джона фон Неймана).

Классическая архитектура – это однопроцессорный компьютер, в котором все функциональные блоки связаны между собой общей шиной, называемой также системной магистралью.

Принципы Джона фон Неймана (сформулированы в 1944 г. при создании первой ламповой ЭВМ ЭНИАК):

1. Принцип использования двоичной системы счисления для представления данных и команд.

2. Принцип однородности памяти. Как программы (команды), так и данные хранятся в одной и той же памяти (и кодируются в одной и той же системе счисления – чаще всего двоичной). Над командами можно выполнять такие же действия, как и над данными.

3. Принцип адресуемости памяти. Структурно основная память состоит из пронумерованных ячеек; процессору в произвольный момент времени доступна любая ячейка.

4. Принцип последовательного программного управления. Программа состоит из набора команд, которые выполняются процессором друг за другом в определенной последовательности.

5. Принцип условного перехода. Команды из программы не всегда выполняются последовательно одна за другой. Возможно присутствие в программе команд условного перехода, которые изменяют последовательность выполнения команд в зависимости от значений данных. Сам принцип был сформулирован задолго до фон Неймана А. Лавлейс и Ч. Бэббиджем, однако он добавлен в общую архитектуру как дополняющий предыдущий принцип.

Структурная организация компьютера

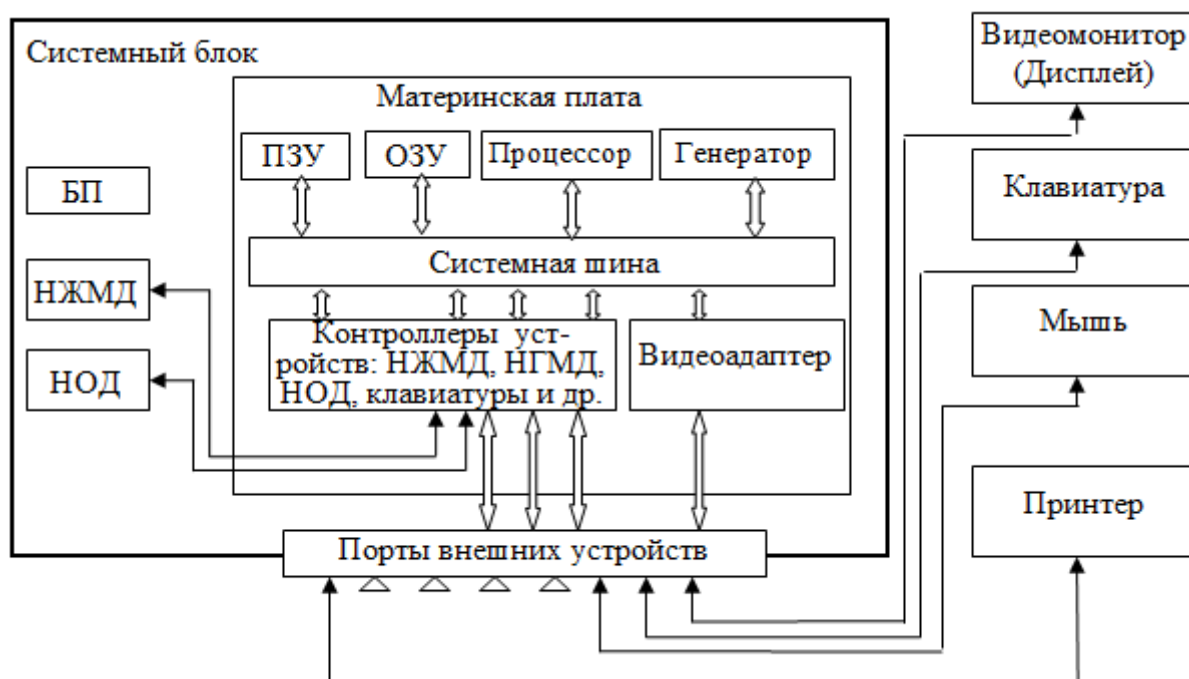
ПК состоит из **аппаратной** части (устройств) и **программной** (программ).

Аппаратная часть имеет модульный принцип конфигурации, то есть к компьютеру можно подключать или отключать от него отдельные блоки или устройства. Аппаратная часть определяет технические возможности ПК, поскольку они зависят от набора технических устройств, из которых состоит ПК.

Программная часть управляет работой аппаратной части и процессом обработки данных (информации).

Состав технических устройств, из которых состоит ПК, называется **аппаратной конфигурацией ПК**.

Конфигурация (состав) ПК может гибко изменяться по мере необходимости. Вместе с этим существует понятие **базовой конфигурации**, которую рассматривают как типичную. Как базовую рассматривают конфигурацию, состоящую из четырех устройств: системный блок, монитор, клавиатура, мышь.



Системный блок – основное устройство, в нем размещаются важные как внутренние, так и внешние устройства компьютера.

Устройства делятся на внутренние и внешние по способу расположения относительно центрального процессорного устройства (ЦПУ – Central Processing Unit, CPU). Все устройства, которые находятся вместе с ЦПУ на одной плате (материнской плате), называются **внутренними**. Внутренние устройства подключаются к материнской плате с помощью специальных разъемов – слотов.

К **внешним** (периферийным) устройствам, как правило, относят устройства ввода-вывода данных и устройства для долговременного хранения данных.

В состав системного блока входят: блок питания (предназначен для преобразования напряжения питания специально для ПК с 220В на 12В, 9В, 5В, 3В, 1,5 В), материнская плата, НЖМД, НОД, порты внешних устройств.

Внутренние устройства ПК

Материнская (системная) плата (motherboard) или чипсет – основная часть персонального компьютера, на ней размещаются устройства, которые определяют тип самого компьютера и его функциональные возможности. Материнская плата содержит две основные большие микросхемы:

1. **Северный мост** (North Bridge) – контроллер-концентратор памяти (MCH), который обеспечивает работу центрального процессора, оперативной памяти и видеоадаптера.

2. **Южный мост** (South Bridge) – контроллер-концентратор ввода-вывода (ICH), обеспечивающий работу контроллеров, интегрированных в материнскую плату устройств (локальной вычислительной сети, звуковой подсистемы, видеоадаптера в отдельном случае), а также взаимодействие с внешними устройствами посредством организации шинного интерфейса.

От микросхем чипсета зависят возможности работы установленных в вычислительной системе процессора, внешних устройств (видеокарты, жесткого диска и др.).

Процессор предназначен для выполнения арифметических и логических операций. Выполнен в виде одной отдельной микросхемы, которая имеет свое название. Название микросхеме присваивает фирма производитель, например, Athlon, Celeron, Sempron, Pentium и др. Как правило, название микросхемы процессора определяет и название самого компьютера. Процессор характеризуется разрядностью и тактовой частотой.

Разрядность – это максимальное количество разрядов (бит) над которым одновременно может выполняться машинная операция. Современные процессоры, как правило, имеют разрядность 32 или 64 бита.

Тактовая частота определяет скорость выполнения элементарных операций внутри процессора (сложения или вычитания), измеряется в МГц, ГГц.

Генератор вырабатывает тактовые импульсы, определяющие тактовую частоту процессора, и синхронизирует работу всех устройств компьютера.

Процессорный сокет, или разъем для установки процессора, является одной из ключевых характеристик материнской платы, поскольку именно он нала-

гает ограничения на выбор процессора для будущей системы. Изначально процессорный сокет был внедрен в устройство платы для повышения мобильности системы и обеспечения возможности замены процессора. Альтернативой существующему разъему процессора когда-то был так называемый слот. Он представлял собой разъем для установки платы, на которой располагался процессор. Выход в свет новых моделей процессоров часто сопровождается разработкой нового сокета для их установки. Таким образом достигается совместимость материнских плат и моделей процессоров.

Оперативно-запоминающее устройство (ОЗУ) – предназначено для временного хранения данных и программ, с которыми на данный момент работает ПК. Данные в ОЗУ хранятся только при включенном ПК. Если ПК выключить, данные, находившиеся в ОЗУ, уничтожаются и восстановлению не подлежат. ОЗУ – это набор микросхем. Характеризуется быстродействием и объемом (количеством информации, одновременно может храниться в ОЗУ).

Постоянно-запоминающее устройство (ПЗУ) – предназначено для длительного хранения постоянных данных, в том числе и когда компьютер выключен. Выполнено в виде отдельной микросхемы. Данные в микросхему помещаются во время ее изготовления и больше не меняются. Микросхема ПЗУ носит название BIOS. В ПЗУ хранятся микропрограммы, обеспечивающие проверку технического состояния устройств ПК при включении питания, проверку наличия устройств и связей между ними, а также программы, которые выполняют поиск операционной системы (ОС) и "загрузку" (запись) ее ядра в оперативно-запоминающее устройство, а также организуют ввод и вывод данных (обмен данными).

Загрузка ОС – это процесс определения устройства, на котором хранится ОС, а затем копирования основного блока (ядра системы) в ОЗУ, после чего управление работой ПК и его устройств выполняет ОС.

ОЗУ и ПЗУ образуют **внутреннюю память** ПК. Внутренняя память характеризуется очень высоким быстродействием и высокой стоимостью. На материнской плате ПК размещают от 2-х и более слотов для подключения модулей памяти ОЗУ.

Системная шина предназначена для передачи данных между внутренними устройствами ПК. Является набором проводников, по которым происходит обмен сигналами между всеми устройствами компьютера (передача данных, команд и синхронизирующих импульсов).

В современном ПК выделяют **шину адреса, шину данных и шину команд**. Шина характеризуется разрядностью, быстродействием. В современных компьютерах системные шины имеют от 64 до 512 разрядов, то есть одновременно данные передаются по 64 или более проводникам.

Контроллеры предназначены для управления работой внешних устройств и согласования скорости работы внутренних устройств с внешними (периферийными). Каждое внешнее устройство имеет свой контроллер.

Слоты – это специальные разъемы, предназначенные для подключения различных основных и дополнительных устройств к материнской плате.

Видеокарта (видеоадаптер) вместе с монитором образуют видеосистему персонального компьютера. Видеоадаптер управляет работой видеомонитора, имеет свою оперативную память (видеопамять), где хранятся данные, которые отображаются на мониторе. Основными характеристиками видеокарты является разрядность и объем ОЗУ, а также способ обмена данными между видеокартой и системной платой. Самыми современными являются видеокарты с объемом видеопамяти от 512 и более Мбайт и разрядностью от 128 и выше бит.

Звуковая карта является неотъемлемым атрибутом современного ПК и служит для ввода (вывода) различной звуковой информации в (из) ПК. Основными характеристиками звуковой карты является разрядность и способ вывода звука во внешнее устройство. Современные звуковые карты поддерживают многополосное разделение аудиоданных, что позволяет воспроизводить сложные аудиоэффекты в развлекательных и специализированных программах.

Сетевая плата (сетевой адаптер) предназначена для обеспечения обмена данными между ПК, со скоростью от 10 и выше Мбит/с. Сетевые платы бывают проводные, когда для связи между ПК необходим специальный кабель (витая пара или оптоволокно) и беспроводные (технология Wi-Fi).

Внешняя память

Внешняя память предназначена для длительного хранения программ и данных. Информация во внешней памяти хранится при выключении ПК. Размер внешней памяти значительно больше, чем внутренней, но скорость записи и считывания информации меньше (время обращения к памяти выше), стоимость внешней памяти значительно ниже внутренней.

К накопителям внешней памяти относятся:

1. Накопители на жестких магнитных дисках (НЖМД или HDD) или винчестеры, которые характеризуются большим объемом (порядка 100 и более Гбайт), достаточно высокой надежностью носителей данных и сравнительно высоким быстродействием. По способу организации передачи данных жесткие диски могут использовать параллельную передачу – P(parallel) ATA, SCSI, а также последовательную – S(serial) ATA, SAS. Основной тенденцией является постепенное вытеснение жестких дисков, использующих параллельную передачу данных их "последовательными" собратьями.

Винчестер содержит набор пластин, которые представляют собой чаще всего металлические диски (толщиной около 2 мм), покрытые магнитным слоем и соединенные между собой с помощью шпинделя (оси).

При магнитной записи информации с помощью записывающей головки происходит изменение магнитной индукции носителя. Носитель изготавливают из ферромагнитного материала. В зависимости от направления тока, протекающего по обмотке, создается магнитный поток, намагничивающий носитель в одном из двух направлений. Различные направления намагниченности носителя соответствуют логическому нулю или логической единицы.

При считывании информации с магнитного диска подвижной намагниченный носитель индуцирует в магнитной головке электродвижущую силу. Поляр-

ность возникающего на обмотке напряжения зависит от направления намагниченности носителя.

В современных НЖМД используется от 4 до 9 пластин, шпиндель вращается с высокой постоянной скоростью (как правило, 5400 или 7200 оборотов в минуту, в серверах – до 15000 оборотов в минуту). Для записи данных используются обе поверхности дисков. Для каждой такой поверхности существует собственная головка записи-считывания, которая считывает или изменяет записанные на поверхности данные. Головки могут перемещаться вдоль радиуса пластин, что вместе с вращением пластин дает головкам возможность доступа к любой части диска. Общий объем информации соответствует суммарной емкости магнитных пластин.

Вся конструкция винчестера находится в герметичном корпусе.

Кроме "стационарных" винчестеров разработаны накопители, которые подключаются к ПК только во время работы, а затем могут быть отключены и храниться с целью сохранения конфиденциальности в сейфе или в другом месте.

2. Оптические (лазерные) диски называют компактными (или дискам типа CD, DVD). Наиболее распространенными являются диски "только для чтения", то есть диски CD-R и DVD-R.

Диски CD-R выпускают диаметром 5,25 и 3,5 дюйма емкостью 700 – 800 Мбайт. В основу записи информации с помощью лазера положена модуляция интенсивности излучения лазера дискретными значениями 0 и 1. При записи логической единицы луч прожигает микроскопическое отверстие. Начинается запись с внутренних дорожек. Запись ведется с большой плотностью – 630 дорожек на миллиметр. Длина всей спиральной дорожки – около 5 км.

При считывании информации с оптического диска луч лазера отражается от поверхности диска, кроме мест, выжженных записывающим лучом.

Емкость дисков DVD-R обычно составляет 4,7 Гбайт. Большая емкость DVD диска обеспечивается меньшим размером ямок, выжигаемых лазером и меньшим наклоном канавок, направляющих лазер и закрученных по спирали. Соответственно чем меньше размер ямок и шаг дорожки, тем больше данных можно записать на ту же площадь диска. Для того, чтобы была возможность прожигать более маленькие ямки используется лазер другого цвета с меньшей длиной волны. Из-за этого в DVD-R используют другой, в сравнении с CD-R, материал записывающего слоя, чтобы поглощать волны нужной длины. Кроме того, существуют двухслойные DVD – DVD-R DL с емкостью 8,54 Гбайт.

Кроме однократно записываемых дисков распространение получили диски с возможностью перезаписи до 1000 раз (CD-RW, DVD-RW), основным недостатком которых является необходимость использования специального программного обеспечения для записи данных.

Технология записи и перезаписи диска другая. Запись осуществляется на диск из стекла, содержащего магнитный слой из специального сплава. С помощью лазера нагревают небольшой участок диска до определенной температуры и прикладывают магнитное поле нужного направления. После охлаждения данный участок запоминает направление намагниченности.

Для считывания данных используют эффект, который проявляется в изменении направления поляризации лазерного луча, отраженного от намагниченной поверхности.

3. Современным видом внешней памяти являются накопители, использующие **FLASH-технологии**. Как запоминающий элемент в устройствах такого типа применяются микросхемы, которые можно перепрограммировать многократно (EEPROM). Существует большое количество устройств данного типа, применяющихся в различных приборах бытовой техники и в ПК.

Для подключения FLASH-памяти в ПК используются универсальная последовательная шина (USB) или специализированное устройство card-reader (кард-ридер).

Основным положительным качеством данной технологии является отсутствие каких-либо механических устройств, задействованных в операциях чтения и записи данных, что значительно повышает надежность устройств, выполненных по данной технологии.

Кроме рассмотренных устройств внешней памяти существует еще ряд устройств, которые используют иные способы чтения и записи данных, но они не получили широкого распространения или из-за большой стоимости реализации, или из-за малого объема данных, которые можно на них хранить.

Устройства ввода данных

К **устройствам ввода данных** относят клавиатуру, мышь, графический планшет, сканер и др.

1. Клавиатура – стандартное устройство ввода данных и команд в персональную электронно-вычислительную машину (ПЭВМ), одно из основных технических средств взаимодействия пользователя с персональным компьютером (ПК). При нажатии любой клавиши на клавиатуре процессору передается код нажатой клавиши, который является кодом символа, изображенного на этой клавише. Программы компьютера (операционная система) продолжают работать с этими кодами. Программа, которая управляет работой клавиатуры, называется **драйвером клавиатуры**.

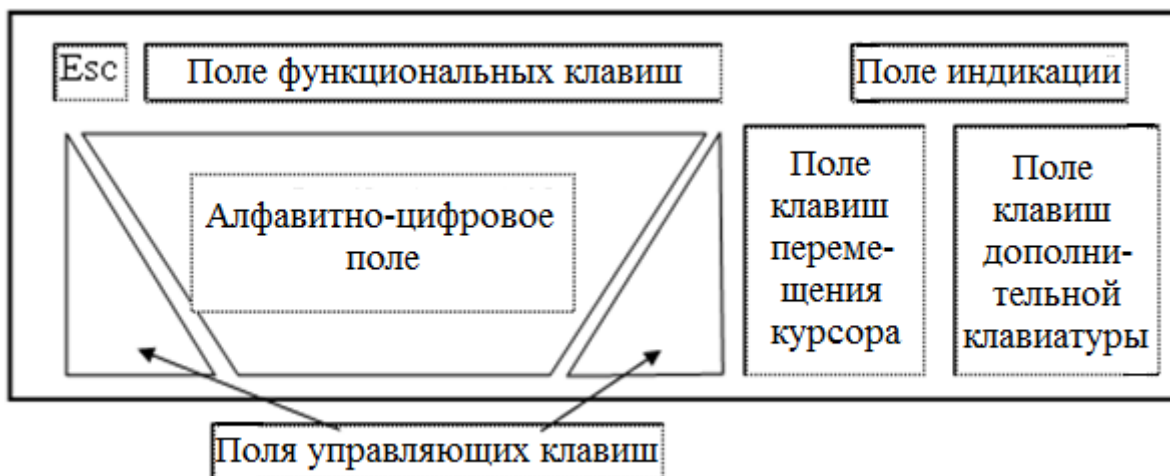
Клавиатура предназначена для ввода в ПК алфавитно-символьной информации (букв, цифр, знаков и др.), а также команд управления работой ПК. Для каждого символа выделяется клавиша, нажав на которую мы вводим код символа в ПК. Символ кодируется 8 или 16 битным двоичным числом.

Клавиатуры, применяемых на ПК, могут быть различных моделей, но все они имеют сходную структуру строения.

Обычно на клавиатуре выделяют шесть полей клавиш:

Основное поле – алфавитно-цифровое – расположено в центре клавиатуры и имеет форму перевернутой трапеции. На нем размещены цифры, буквы латинского и русского (украинского) алфавитов. Их расположение такое же, как на английских или русских (украинских) печатных машинках. На этом же поле расположены символы-разделители и специальные знаки. Клавиши алфавитно-

цифрового поля предназначены для ввода кода изображенных на них символов в память ПК.



Поле функциональных клавиш расположены выше алфавитно-цифрового поля в одну линию и имеет 12 клавиш, которые обозначаются: [F1], [F2], [F3], [F4], ..., [F12]. Назначение клавиш определяется программой, которая в данный момент выполняется на ПК.

Поле управляющих клавиш расположено слева и справа от алфавитно-цифрового поля в виде прямоугольных треугольников. Клавиши этого поля используются, в основном, для управления режимами работы клавиатуры.

Назначение управляющих клавиш:

[Shift] – служит для ввода прописных букв и других символов, расположенных в верхнем регистре клавиатуры. Например, чтобы ввести строчную "f", нужно нажать клавишу, на которой изображена "F", (клавишу [F]), а чтобы ввести прописную букву "F", нужно нажать клавишу [Shift] и, не отпуская ее, нажать клавишу [F].

[Caps Lock] – служит для фиксации режима ввода прописных букв. В этом режиме при обычном нажатии на буквенные клавиши вводятся прописные буквы, а при нажатии клавиши [Shift] – строчные. Повторное нажатие клавиши [Caps Lock] отменяет режим ввода прописных букв и включает режим ввода строчных букв.

Клавиши **[Ctrl]** и **[Alt]** используются в комбинации с другими клавишами для выполнения специальных команд (операций) или изменения их функций (наиболее часто используются в комбинации с клавишей [Shift]).

[Tab] (клавиша табуляции) – используется для перемещения курсора обычно на 14 позиций вправо, но ее назначение может быть и другим.

[Backspace] (шаг назад) – используется для удаления символов слева от курсора. Обозначение этой клавиши – длинная стрелка влево [←], она находится над клавишей [Enter].

[Enter] – для сообщения компьютеру о том, что пользователь закончил выполнять какие-то действия, например, набрал команду, а также для окончания ввода строки (абзаца) и перемещения курсора на следующую строку.

[Esc] (Escape) – используется для отмены какого-либо действия, выхода из режима программы и т.п.

Поле управления курсором расположено отдельно, справа от алфавитно-цифрового поля. Предназначено для перемещения курсора по экрану в направлении, соответствующем указателю на клавише. Содержит 10 клавиш. Четыре из них: **[↑]** **[↓]** **[→]** **[←]** (стрелки вверх, вниз, вправо, влево) – перемещают курсор в указанном направлении. Остальные шесть:

[Home] и **[End]** – служат для перемещения курсора в начало или конец строки соответственно.

[PgUp] (Page up) и **[PgDn]** (Page Down) – перемещают курсор на "страницу вверх" или "страницу вниз" (под страницей подразумевается экран дисплея).

[Del] (Delete) – служит для удаления символов, находящихся под курсором или справа от него.

[Ins] (Insert) – служит для переключения двух режимов ввода символов: ввода с раздвижкой символов (вставка) и ввода с замещением ранее набранных символов (замена).

Поле дополнительных клавиш (дополнительной клавиатуры) расположено отдельно, справа от поля управления курсором. Предназначено для управления курсором или ввода цифровой информации. Имеет кнопку переключения режима работы **[Num Lock]**, которая включает режим ввода цифр, а также клавиши **[Insert]**, **[Delete]**, **[Enter]** и клавиши ввода символов арифметических операций.

Поле индикаторов режимов расположено в правом углу клавиатуры над полем дополнительных клавиш. Индикатор "**Num Lock**" показывает, что включен цифровой режим работы дополнительного поля клавиатуры. Индикатор "**Caps Lock**" показывает, что включен режим прописных букв. Индикатор "**Scroll Lock**" показывает, что включен режим блокировки прокрутки.

Особые комбинации клавиш. Некоторые действия при работе с компьютером осуществляются нажатием не одной клавиши, а их комбинации. Это называют одновременным нажатием клавиш. Одновременное нажатие клавиш означает, что пользователь должен нажать первую клавишу и, не отпуская ее, нажать следующую клавишу. Примеры одновременного нажатия клавиш и их назначение:

[Ctrl]+[Break] – принудительное завершение работы программы или команды.

[Ctrl]+[Alt]+[Del] – вызов диспетчера задач.

[Shift]+[Print Scr] – включает и выключает режим копирования на принтер выводимой на экран информации. Клавиша **[Prt Scr]** (Print Screen) используется для копирования содержания экрана в буфер обмена (скриншот), что позволяет вставлять изображения с видеомонитора в текстовые и прочие документы в виде рисунка.

[Ctrl]+[Num Lock] – прерывает выполнение программ. Для продолжения их выполнения нужно нажать любую клавишу.

[Ctrl]+[Home] – перемещает курсор в начало текста.

[Ctrl]+[End] – перемещает курсор в конец текста.

2. Манипулятор "мышь" (mouse) используется для удобства взаимодействия пользователя с операционной системой и ее приложениями. Мышь представляет собой небольшую коробочку с двумя (тремя) клавишами, которая легко умещается в ладони. Вместе с проводом для подключения к компьютеру это устройство действительно напоминает мышь с хвостиком. При перемещении мыши по столу или другой поверхности (обычно по коврику) на экране компьютера передвигается указатель мыши (как правило, в виде стрелки). Когда необходимо выполнить какое-либо действие, например, выбрать пункт меню, пользователь устанавливает указатель мыши на этот пункт и нажимает левую кнопку мыши.

Ознакомимся с некоторыми понятиями, связанными с использованием мыши.

Щелкнуть мышью (Click) – означает установить указатель мыши на некоторый объект, нажать левую кнопку мыши (ЛКМ) и отпустить ее (кратковременное нажатие приблизительно на полсекунды). Применяется для выделения объекта.

Дважды щелкнуть мышью (Double Click) – означает установить указатель мыши на некоторый объект, а затем дважды нажать и отпустить ЛКМ. Приводит к запуску программы или открытию файла (папки).

Два последовательных щелчка ЛКМ на имени объекта позволяют переименовать данный объект.

Переместить объект мышью (Drag And Drop) – означает установить указатель мыши на некоторый объект, нажать ЛКМ и, удерживая ее, переместить объект на новую позицию, после чего отпустить кнопку мыши.

Щелчок правой кнопкой мыши (ПКМ) – открывает контекстное меню.

Специальное перетаскивание – нажать ПКМ и не отпуская переместить на новое место объект, отпустить ПКМ – появится контекстное меню, содержащее команды: копирование объекта, создания ярлыка объекта и др. Выбор операции в контекстном меню производится щелчком ЛКМ по необходимой команде.

Клавиатура и мышь могут быть как проводными так и беспроводными.

3. Сканеры – предназначены для ввода в компьютер графических и текстовых объектов (рисунков, фотографий, текста книг). Сканеры характеризуются: разрешением сканирования (количество распознаваемых точек на одном дюйме), измеряется в dpi; скоростью сканирования (время сканирования листа формата А4), измеряется в dpi/сек; размером сканируемого объекта.

Сканеры разделяются на ручные, планшетные, роликовые и проекционные.

Ручные сканеры – это достаточно простые компактные устройства, предназначенные для сканирования вручную небольших фрагментов изображения (репродукций, фотографий, документов, штрих-кодов) с невысоким разрешением.

Планшетные сканеры – универсальные устройства, популярные среди пользователей ПК. Они входят в группу настольных или напольных сканеров и предназначены для ввода текстовых документов, изображений, слайдов и фотоленок, а также объемных образцов.

Роликовые сканеры – камера неподвижна, перемещение документа относительно камеры осуществляется путем протягивания документа через систему роликов.

Проекционные сканеры – являются настольными или напольными устройствами, основной особенностью которых является сканирование образцов трехмерных проекций.

Принцип действия сканеров: копируемое изображение освещается источником света. При этом луч света осматривает (сканирует) каждый участок оригинала. Отраженный от бумажного листа луч света попадает на прибор, осуществляющий преобразование оптической картинки в электрические сигналы.

Графический планшет – это устройство, выполненное в виде небольшого сенсорного экрана и предназначенное, в основном, для работы фотохудожников и дизайнеров.

Устройства вывода данных

К устройствам вывода относятся монитор (дисплей) и различные виды печатающих устройств.

1. Дисплей (видеомонитор) предназначен для отображения на экране алфавитно-цифровой и графической информации, с которой в данный момент работает пользователь ПК.

Дисплей управляется видеоконтроллером (видеоадаптером), который находится в системном блоке. Видеоконтроллер имеет свою личную память, которую называют видеопамятью. Все, что мы видим на экране дисплея, находится в видеопамети.

Видеоконтроллер может задавать два режима работы дисплея: текстовый или графический. В текстовом режиме на экран выводятся только символы. Текущее место экрана, куда выводится текущий символ, отмечается мерцающим значком, который называют **курсором** или **маркером**.

В графическом режиме на экран может выводиться любое изображение. Оно формируется с помощью точек, называемых **пикселями**.

Основные характеристики дисплея: разрешение, количество цветов (палитра) и размер экрана.

Разрешение – это количество точек (пикселей), которые размещаются на одном дюйме. Чем больше количество пикселей, расположенных на одном дюйме, тем качество изображения лучше.

Размер экрана дисплея измеряют по диагонали в дюймах. Дюйм – 2,54 см. Используют мониторы от 15 дюймов и выше.

Существуют следующие виды дисплеев: с электронно-лучевыми трубками (ЭЛТ), плазменными и жидкокристаллическими.

Внутри дисплея с электронно-лучевой трубкой находится специальная электронная пушка, испускающая поток электронов ("луч"), которые под действием магнитного поля движутся к люминесцентному покрытию. Взаимодействие узкого потока электронов с этим покрытием дает световое излучение, которое воспри-

нимается глазом как цветная точка. Изображение формируется путем пробежки луча с определенной скоростью вдоль всей площади экрана монитора.

Принцип работы плазменных дисплеев подобный работе мониторов с ЭЛТ, но люминофор светится не под воздействием потока электронов, как в ЭЛТ, а под воздействием плазменного разряда. Каждая ячейка (пиксель) плазменного дисплея является флуоресцентной мини лампой, которая способна излучать только один цвет. Изменение величины напряжения, приложенного к пикселю, меняет яркость свечения пикселя, за счет чего получают необходимые цветовые оттенки.

В жидкокристаллических дисплеях между двумя стеклянными панелями расположены жидкие кристаллы. Жидкий кристалл – это промежуточное состояние между жидкой и твердой фазами вещества. Он проводит или не проводит свет в зависимости от приложенного к нему напряжения или температуры.

Основные преимущества жидкокристаллических мониторов - это существенное уменьшение излучения по сравнению с ЭЛТ, отсутствие мерцания изображения, небольшие габариты, малый вес и др.

К недостаткам можно отнести ограниченный угол обзора. При взгляде на матрицу под углом, превышающим определенное значение, цвета становятся бледными или искажаются.

2. Печатающие устройства предназначены для вывода данных (информации) на бумагу, специальную пленку и другие виды носителей символьной и графической информации.

По способу нанесения информации на бумагу различают матричные, струйные, лазерные и светодиодные принтеры.

Принтеры характеризуются скоростью печати, качеством и форматом (размером бумаги, на который можно печатать).

Качество печати определяется способностью принтера нанести печатные точки на один квадратный сантиметр или дюйм (dpi).

Матричные принтеры для формирования требуемого изображения используют фиксированную матрицу, состоящую из 9 или 24 печатающих игл. Для отображения необходимой информации используется контрастная лента или набор из лент основных цветов. Недостатком данных принтеров является низкое качество и скорость печати, положительное качество - низкая стоимость.

Печатающие головки **струйных принтеров** вместо иголок содержат тонкие трубочки - сопла, через которые на бумагу выбрасываются капельки чернил. Печатающая головка струйного принтера содержит от 12 до 64 сопел, диаметры которых тоньше человеческого волоса. Недостатки принтера - низкая устойчивость печати к внешним воздействиям, достоинство - очень высокое качество печати.

В **лазерных принтерах** используется электрографический принцип создания изображения. Процесс печати включает создание невидимого рельефа электростатического потенциала в слое полупроводника с последующей его визуализацией. Визуализация осуществляется с помощью частиц сухого порошка - тонера, наносимого на бумагу. Принтеры характеризуются высокой скоростью, высоким качеством и низкой стоимостью печати.

Светодиодный принтер – один из видов принтеров, которые представляют собой параллельную ветвь развития технологии лазерной печати. Для формирования картинки на барабане используется линейка светодиодов. Метод имеет только один недостаток - качество печати чуть уступает лазерному. Скорость печати здесь зависит от количества цветов: черно - белые принтеры работают чуть медленнее лазерных аналогов, зато цветная светодиодная печать быстрее. По всем остальным параметрам светодиодные принтеры ушли далеко вперед – они мало стоят, а главное, они считаются менее вредными для здоровья, чем лазерные.

В настоящее время распространение получает особый вид принтеров – 3D-принтеры. **3D-принтер** – это устройство, использующее метод послойного создания физического объекта по цифровой 3D-модели. Их назначение заключается не в выводе информации из ПК на бумагу, а в создании объемных объектов под управлением специальных программ с помощью ПК. 3D-печать может осуществляться разными способами и с использованием различных материалов (пластика, фотополимерной смолы, силикона, воска, металла), но в основе любого из них лежит принцип послойного создания (выращивания) твёрдого объекта. Существующие на сегодняшний день 3D-принтеры используют две основные технологии - лазерную и струйную, которые в свою очередь разделяются на отдельные виды, в зависимости от используемого материала.

3. Плоттеры (раньше их называли графопостроителями) используются в инженерной практике для печати графической информации больших форматов (более А3). Различают струйные и лазерные плоттеры.

Для доступа к ресурсам глобальной сети Интернет и организации связи между удаленными компьютерами используются специальные устройства – модемы, предназначенные для организации обмена данными между компьютерами с использованием традиционных телефонных или оптоволоконных линий с помощью специальных правил – протоколов.

Модемы различаются по исполнению (внешние или внутренние), по принципу работы (аппаратные или программные), по типу сети, к которой производится подключение, а также по поддерживаемым протоколам передачи данных.

Внешние модемы подключаются через COM, LPT, USB или Ethernet-порт.

Внутренние модемы дополнительно устанавливаются внутрь системного блока.

В аппаратных модемах все операции преобразования сигнала, поддержка физических протоколов обмена производятся встроенным в модем вычислителем.

В программных модемах все операции по кодированию сигнала, контролю ошибок и управлению протоколами реализованы программно и производятся центральным процессором компьютера.

По типу сети и соединения различают:

Модемы для телефонных линий – наиболее распространённые в XX веке.

Кабельные модемы – используются для обмена данными по специализированным кабелям.

Радиомодемы – работают в радиодиапазоне, используют собственные наборы частот и протоколы.

Беспроводные модемы – работают по протоколам сотовой связи.

Спутниковые модемы – используются для организации спутникового Интернета. Принимают и обрабатывают сигнал, полученный со спутника.

PowerLine-модемы – используют технологию передачи данных по проводам бытовой электрической сети.

Лекция 3

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ПК. ФАЙЛОВАЯ СИСТЕМА

Классификация программного обеспечения

Работа устройств ПК осуществляется под управление программ. Совокупность программ, которые используются на компьютере, называют **программным обеспечением**.

Программное обеспечение можно разделить на две основные части: системное и прикладное ПО.

Системное программное обеспечение включает:

Операционные системы – это пакет программ, которые предназначены для управления работой ПК, распределения его ресурсов, управления работой других прикладных программ, а также для организации взаимодействия пользователя с ПК. Например: ОС Windows, Linux, Unix и др.

Системы программирования (инструментальное программное обеспечение) – это пакет программ, которые предназначены для облегчения и частичной автоматизации процесса разработки и отладки программ. Например: Visual Basic, C#, Java и др.

Сервисные программы (утилиты) расширяют возможности ОС, например, позволяют проверять техническое состояние устройств, выполняют архивацию и разархивацию информации, поиск и удаление вирусов и т.д.

Прикладное программное обеспечение делится на:

Прикладное программное обеспечение общего назначения – это комплекс программ, который получил широкое использование среди большого количества разных категорий пользователей. Например: Microsoft Office.

Прикладные программы специализированного назначения – это программы, предназначенные для решения широкого круга задач в какой-либо области деятельности. Например, для решения математических задач в разных отраслях используется пакет МАТЕМАТИСА, для решения инженерных задач MathCad, бухгалтерских задач пакет 1С-Бухгалтерия и т.д.

Прикладные программы специального назначения используются в специфической деятельности пользователя. Например: программы расчета освещенности дорог и улиц, насыщенности воздуха ядовитыми веществами и т.д.

Общие сведения об операционных системах ПК

Операционные системы (ОС) состоят из программных модулей, которые позволяют пользователю руководить машиной, а также обеспечивать взаимодействие программ с внешними устройствами и с другими программами.

ОС можно назвать интеллектуальными системами, потому что они могут оценивать ситуацию и принимать решение или рекомендовать пользователю, какие действия нужно выполнять в той или другой ситуации. Взаимодействие пользователя с ОС осуществляется в диалоговом режиме. То есть пользователь задает команду ОС – система анализирует ее, и если она верная, то выполняет и т.д.

В составе всех ОС можно выделить три основные части: командный язык, файловую систему, и систему управления внешними устройствами.

Командный язык – это набор команд, которые вводятся пользователем из клавиатуры или с помощью других устройств, например мыши, и немедленно выполняются.

Команда – это зарезервированное служебное слово, которое имеет свои правила записи (формат записи) и не может использоваться с другой целью.

Командный язык – это та часть ОС, которая осуществляет поддержку взаимодействия пользователя со всеми ресурсами ПЭВМ. Часть ОС, которая осуществляет анализ и выполнение команд пользователя, называют командным процессором ОС.

Файловая система предназначена для организации хранения файлов и определения места их нахождения с целью выполнения различных действий (операций) над ними.

Система управления внешними устройствами – это совокупность специальных программ, которые называют драйверами. Каждый тип внешнего устройства обслуживается индивидуальным драйвером. Драйверы стандартных внешних устройств иногда хранятся в ПЗУ. Драйверы других внешних устройств находятся на системном диске.

Файловая система

Файловая система – это совокупность программ, которые обеспечивают работу с файлами и папками, а также, сама совокупность файлов и папок, которые хранятся на долговременных запоминающих устройствах.

Папка – это место на машинном носителе информации, которое имеет имя, и в котором регистрируются файлы или другие папки (подпапки). Папки еще называют каталогами или директориями.

Файл – это совокупность данных, которые хранятся на машинном носителе информации под одним именем и воспринимаются ОС как единое целое.

Каждый файл имеет свой идентификатор (метку, которая присуща только ему). Идентификатор состоит из двух частей: **имени** и **типа** файла. **Тип файла** определяет пользователь или программа, с помощью которой он создается. Имя файла отделяется от типа точкой. Тип файла еще называют **расширением** файла. Тип файла не обязательная часть имени и может отсутствовать. Примеры имен файлов:

format.com
setup.exe
otch_06.docx
stat1.xls

Каждый файл регистрируется в папке (каталоге, директории).

Папка, таким образом, будет иметь содержание, то есть перечень файлов и папок (каталогов), которые в нем находятся (зарегистрированы). Каждый файл характеризуется: именем, типом, размером датой и временем создания, местом нахождения. В папке хранятся характеристики файлов.

Над файлами и папками можно выполнять следующие действия: создавать, просматривать содержимое, копировать, перемещать, переименовывать, удалять.

В именах файлов могут использоваться знаки-заменители символов "*" и/или "?".

Имена файлов, в которых используются символы-заменители "*" и "?" называют **шаблонами** имен файлов, потому что в этом случае можно работать с файлами, которые имеют одинаковые признаки имен.

Знак "*" в имени файла заменяет любое количество произвольных символов, а знак "?" – любой, но только один символ.

Примеры:

f*.com

t?r.bas

*.com

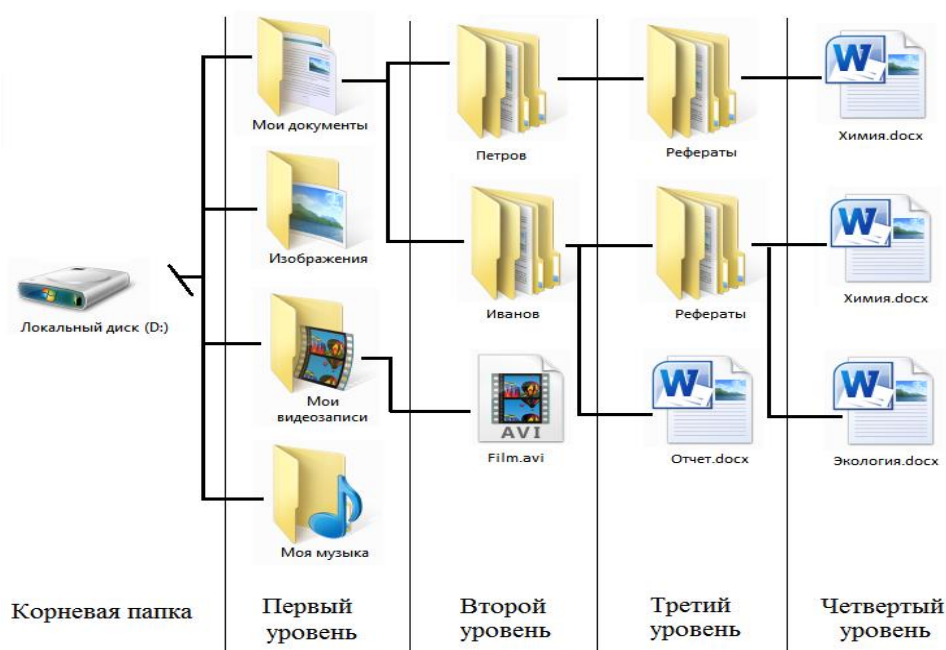
??r.*

.

tor.*

Файлы на дисках хранятся в папках (каталогах), которые организованы как иерархическая структура.

Каждый диск начинается с корневой (главной) папки (каталога). Она изображается символом обратный слеш "\". В корневой папке (каталоге) регистрируются папки (каталоги) первого уровня и их файлы, в папках (каталогах) первого уровня регистрируются папки (каталоги) второго уровня и их файлы и так далее. Такая структура изображается в виде иерархического дерева папок (каталогов):



Для иерархической файловой системы не достаточно указать только имя файла, а нужно еще указать место его нахождения (регистрации). Место нахождения указывается перечнем имен папок разделенных символом обратной слеш, которые нужно пройти (открыть), чтобы найти необходимый файл. Перечень имен файлов называют **маршрутом**. Маршрут указывает путь, который нужно пройти.

Если путь начинается со знака "\", то место положения файла определяется от корневой папки текущего диска. **Текущий диск** – диск, с которым в настоящее время работает ОС.

Пример:

D:\Мои документы\Иванов\Отчет.docx

Путь к файлу Имя файла

Полное имя файла ≤ 255 символов

Лекция 4. ОС MICROSOFT WINDOWS. ТИПЫ ОКОН

Появление операционных систем семейства Microsoft Windows знаменует начало новой эры в области операционных систем для персональных компьютеров, эти системы обеспечивают новый, упрощенный подход к работе на компьютере.

Первая версия графической оконной системы Windows 1.0 выпущена в 1985 г. в качестве дополнения к операционной системе MS DOS. Была признана неудачной из-за невозможности обеспечения существовавшими тогда ПЭВМ функционирования графической среды.

Версия Windows 2.0 в 1988 г. сняла этот недостаток, но распространения она не получила.

Windows 3.1 в 1992 г. была официально объявлена операционной системой, но работала вместе с MS DOS.

Современное семейство ОС Windows (Windows95, Windows NT, Windows 98, Windows 2000, Windows XP, Windows 7, Windows 8, Windows 10) – это 32 (64)-разрядные операционные системы обеспечивающих одновременную работу нескольких приложений (программ). Для общения с пользователем в них используется графический интерфейс в виде унифицированных окон, который намного облегчает общение пользователя с операционной системой.

Главная особенность интерфейса этих систем:

- наличие рабочего стола и панели задач;
- наличие развитой структуры меню команд;
- стандартизация взаимодействия пользователя с приложениями (программами разработанными для работы в ОС Windows), которые выполняются под управлением операционной системы на ПК;

- наличие так называемого контекстного меню. Открывается это меню, если щелкнуть правой кнопкой мыши (ПКМ) по значку объекта. Содержание меню зависит от состояния объекта и его типа.

Вся работа, которую выполняет пользователь, используя ОС Windows, происходит в окнах.

Окно – это контейнер, содержание которого графически отображает содержание папки или приложения (программы).

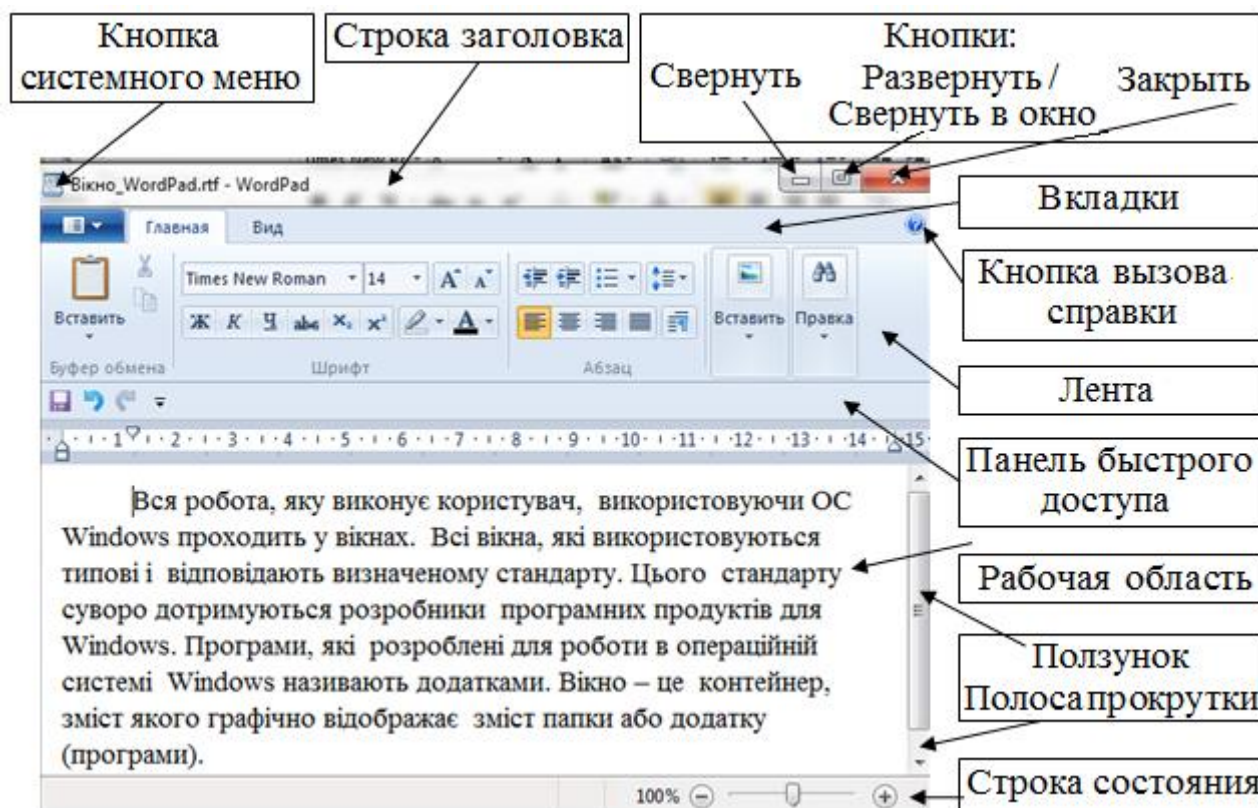
Все окна имеют типовую структуру и отвечают определенному стандарту. Программы, разработанные для работы в операционной системе Windows, называют приложениями.

Структура типового окна – это набор элементов из которых состоит окно и порядок их размещения.

В Windows различают окна приложений, окна папок, диалоговые окна и окна свойств.

Окна приложений

Структуру типового окна приложения рассмотрим на примере окна программы WordPad:



Кнопка системного меню открывает набор команд, предназначенных для изменения размера и положения окна на рабочем столе, в основном при использовании клавиатуры. Для вызова системного меню можно использовать комбинацию клавиш [Alt]+[Пробел].

В строке заголовка выводится имя документа и приложения, которому принадлежит это окно.

За заголовок выполняется перемещение окна на Рабочем столе с помощью мыши.

Слева от имени находится кнопка пиктограммы **системного меню**. При нажатии на нее открывается системное меню управления окном.

В правой части строки заголовка расположены три кнопки управления окном.

Кнопка **Свернуть** – позволяет свернуть окно в пиктограмму на Панель задач. При этом выполнение программы, запущенной в данном окне, продолжается. Для восстановления окна на Рабочем столе достаточно щелкнуть по соответствующей пиктограмме левой кнопкой мыши (ЛКМ).

Кнопка **Развернуть/Свернуть в окно** выполняет две функции – при нажатии впервые окно раскрывается на весь экран, при повторном нажатии окно принимает исходный размер.

При нажатии на кнопку **Закреть** происходит полное закрытие окна. Выполнение программы, которая была запущена в этом окне, прекращается.

Вкладки – открывают на ленте доступ к командам, которые позволяют проводить операции с содержимым окна или с окном в целом. Для различных приложений набор вкладок может быть различным, но в окнах всех приложений присутствуют: **Файл, Главная, Вид**.

Вкладка **Файл** используются для работы с файлами (документами). Позволяет создавать новые файлы (документы), сохранять, открывать, перемещать, изменять параметры.

Вкладка **Главная** чаще используется для копирования, перемещения и форматирования документов.

Вкладка **Вид** предназначена для изменения параметров просмотра информации в окне.

Кнопка  используется для получения справочной информации.

На ленте размещены все необходимые для работы инструменты, которые сгруппированы по назначению на отдельных вкладках.

Панель быстрого доступа предназначена для размещения команд, которые часто используются пользователем. Для помещения команды на панель быстрого доступа необходимо щелкнуть правой кнопкой мыши на названии команды и в контекстном меню выбрать команду **Добавить на панель быстрого доступа** или выполнить команду **Файл → Параметры → Панель быстрого доступа**, а в открывшемся диалоговом окне выполнить настройку быстрого доступа.

Рабочая область – это окно документа, с которым работает пользователь.

Полоса прокрутки появляется по правому и нижнему краям рабочей области, если информация в окне не может отображаться полностью, потому что размер документа слишком велик или размер окна слишком мал. С помощью полос прокрутки можно просматривать все содержание окна. Полоса прокрутки имеет ползунок и две концевые кнопки. Прокрутку выполняют тремя способами:

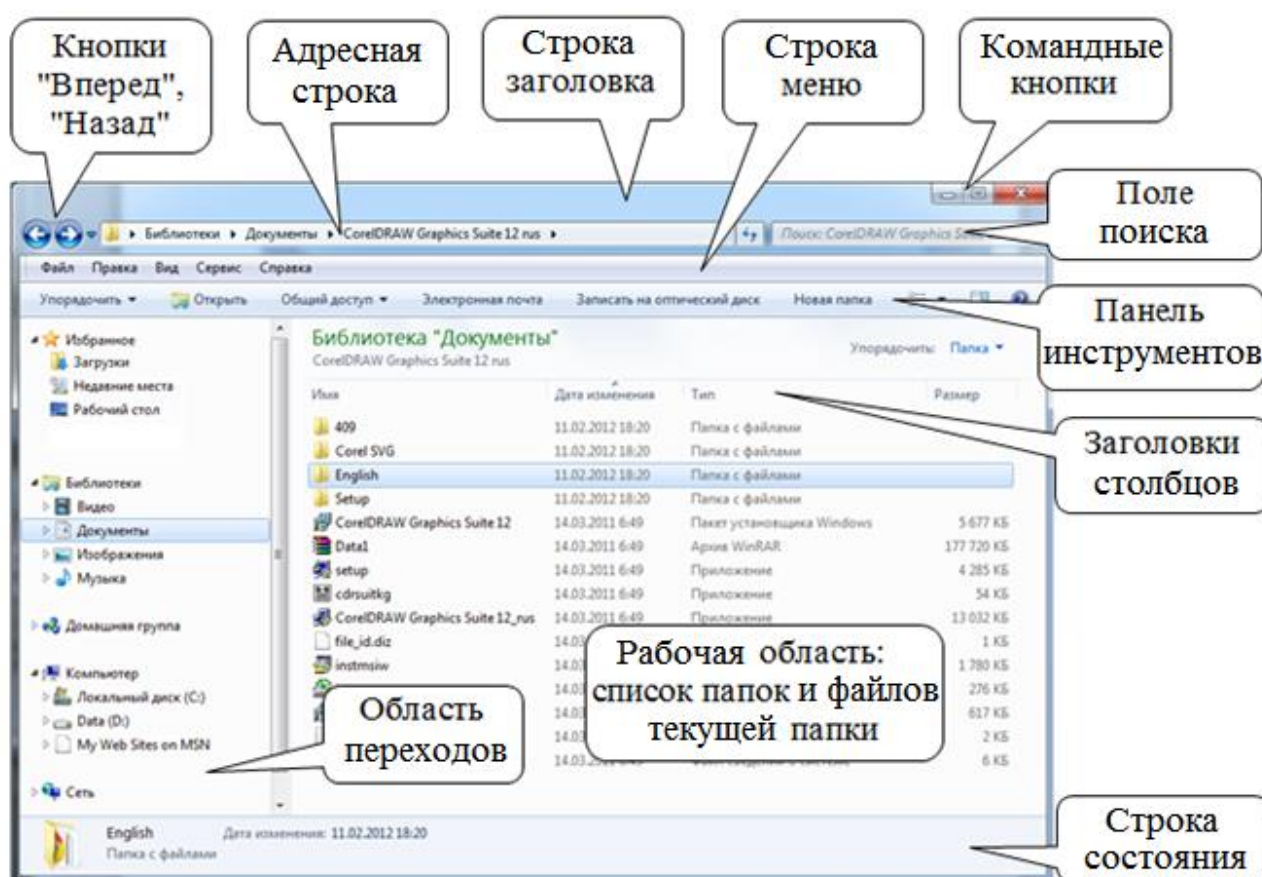
- нажимая на одну из конечных кнопок;

- перетаскивая ползунок;
- нажимая на полосу прокрутки выше или ниже ползунка.

Строка состояния предназначена для отображения вспомогательной информации.

Окна папок

Окна папок по структуре немного отличаются от окон приложений. По умолчанию окно папки разделено на две части разделительной полосой. С левой стороны расположена область переходов со структурой папок, с правой стороны – область просмотра содержания папки, которая выбрана в левой области.



Область переходов предоставляет доступ к библиотекам, файлам и папкам компьютера и сети.

В разделе **Избранное** находятся папки **Загрузки**, **Недавние места**, **Рабочий стол**. В папке **Загрузки** хранятся результаты загрузок из сети. Папка **Недавние места** предоставляет возможность быстрого доступа к 30 папкам, которые открывались в последнее время. Папка **Рабочий стол** позволяет открыть содержимое рабочего стола в окне папки.

В разделе **Библиотеки** находятся документы пользователя, сгруппированные по назначению.

Разделы **Компьютер** и **Сеть** открывают иерархическую структуру, с помощью которой можно перейти к папке (файлу), к которым есть доступ в компьютере или в сети соответственно.

Строка заголовка – первая строка окна папки, имеет только командные кнопки, с помощью которых можно изменять размер окна и сворачивать его в пиктограмму на Панель задач. За заголовок выполняется перемещение окна на Рабочем столе с помощью мыши.

Кнопки **Вперед** и **Назад** позволяют быстро перемещаться к папкам, которые были открыты ранее.

В **адресной строке** отображается иерархический путь к текущей папке. Щелкнув по имени папки в адресной строке можно быстро отобразить содержимое этой папки в окне. Папки в адресной строке разделяются стрелками ►. При нажатии на эти стрелки выводится список вложенных папок, используя который можно быстро перейти в окно нужной папки.

Поле поиска предназначено для организации поиска необходимых объектов (файлов, папок) в текущей папке.

Строка меню – необязательный элемент окна папки. Для отображения этой строки необходимо на данной папке нажать кнопку [Alt] или выполнить команду **Упорядочить** → **Представление** → **Строка меню**. Пункты строки меню открывает доступ к командам, которые позволяют проводить операции с содержимым окна папки или с окном в целом.

Панель инструментов предназначена для выполнения действий над содержимым и элементами окна папки. Состав кнопок на панели инструментов зависит от типа папки и от информации, которая в ней содержится.

Заголовки столбцов предназначены для отображения в режиме **Таблица** названия столбцов, в которых представлены свойства объектов (файлов, папок), находящихся в текущей папке. Щелчком по заголовку столбца можно упорядочить информацию в окне по соответствующему параметру (по дате создания, по размеру и т. п.).

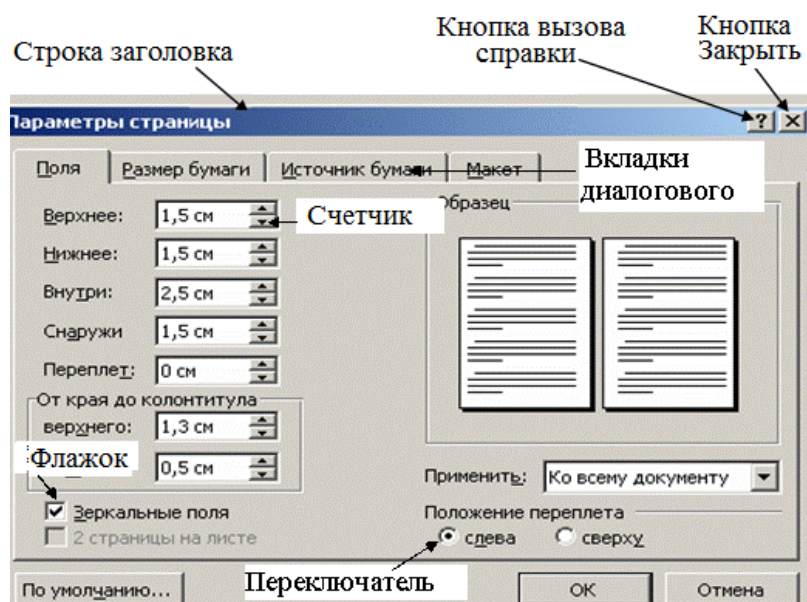
Диалоговые окна

Диалоговые окна используют в тех случаях, когда пользователь должен вмешаться в работу операционной системы или приложения с целью изменения режимов работы, изменения некоторых параметров, ввода дополнительной информации и т.д. Часто в таких окнах для пользователя выводится информация, на основе которой он должен принять решение об изменениях соответствующих режимов в работе или увидеть реакцию системы на его действия. Диалоговое окно не изменяет своего размера.

Рассмотрим элементы, которые могут входить в диалоговое окно.

Диалоговое окно имеет **строку заголовка**, в которую входят: название окна (отражает основное назначение окна), кнопка вызова справки и кнопка закрытия окна.

Рабочее поле состоит из **вкладок**, которые организованы в виде отдельных полей (как страницы в записных книжках с алфавитом). Каждая вкладка имеет свое название и отображает параметры, которые можно изменить используя элементы данной вкладки.



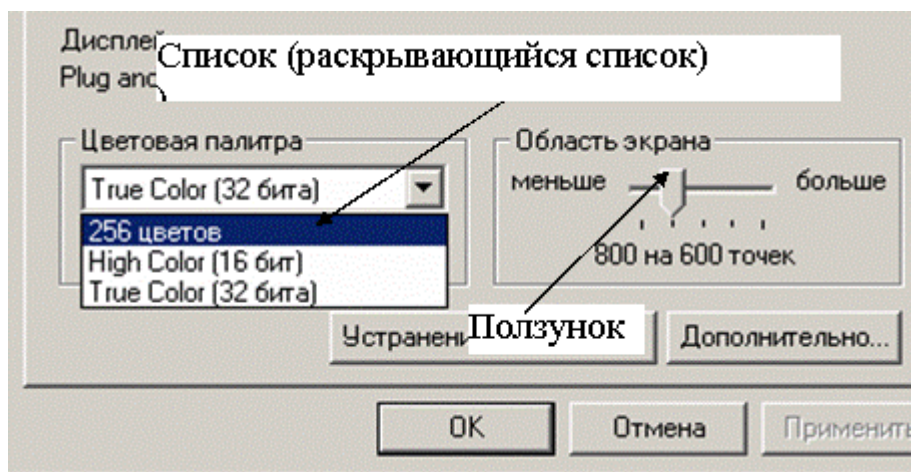
Флажок предназначен для выбора параметров или установки соответствующих режимов работы. Флажок считается установленным, если в окне рядом с названием параметра стоит пометка в виде галочки. Состояние флажка не зависит от состояния других флажков, расположенных в том же окне.

Переключатель – выбор отмечается появлением точки в центре круга. Из списка параметров может быть включен только один переключатель.

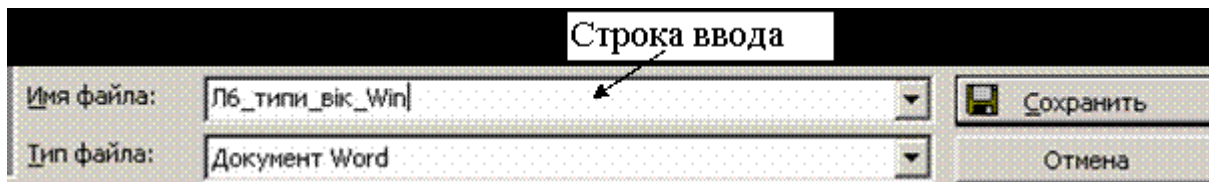
Счетчик – в нем отображаются конкретные числовые значения, имеет две кнопки управления для увеличения или уменьшения значения параметра.

Список – используются открытые или раскрывающиеся списки. Раскрывающийся список нужно сначала открыть с помощью кнопки выбора, а затем выбрать нужные параметры с помощью мыши. В открытом списке элемент выбирают с помощью полосы прокрутки.

Ползунок - используется для изменения значения некоторого параметра в соответствующем интервале.



Строка ввода – используется для ввода текстовой информации. Для ввода данных ее необходимо активизировать, щелкнув мышью.



Кнопки окончания работы с диалоговыми окнами – после окончания установки параметров для продолжения работы компьютера приложению необходимо щелкнуть одну из кнопок.

Сброс – уничтожает всю введенную информацию и обеспечивает возврат к стандартным условиям.

Применить – позволяет сохранить изменения параметров в окне, не закрывая окна.

ОК – кнопка закрывает окно и сохраняет все изменения.

Отменить – закрывает окно без сохранения изменений параметров.

Окна "Свойства"

Окно **Свойства** - это особый вид диалоговых окон. Основное назначение – получение справочной информации о конкретном объекте с возможностью изменения некоторых его параметров. Для открытия такого окна сначала необходимо выбрать объект, а затем выполнить команду **Файл** → **Свойства** или с помощью контекстного меню объекта.

Лекция 5

БОРЬБА С ВИРУСАМИ. АРХИВАЦИЯ ФАЙЛОВ

Компьютерные вирусы и их классификация

Компьютерный вирус – это специально написанная небольшая по размерам программа, которая может приписывать себя к другим программам (т.е. "заражать" их), а также выполнять различные нежелательные действия на компьютере.

Программа, в которой находится вирус, называется зараженной. Когда такая программа начинает работать, то сначала управление получает вирус. Вирус находит и "заражает" другие программы, а также выполняет вредоносные действия, например, портит файлы или таблицу размещения файлов на диске, заражает оперативную память и т.д.

Компьютерные вирусы классифицируются в зависимости от:

- среды размещения;
- особенностей алгоритма;
- способов заражения;
- степени воздействия (безвредные, опасные, очень опасные).

В зависимости от среды размещения основными типами компьютерных вирусов являются:

- программные вирусы;
- загрузочные вирусы;

- макровирусы;
- сетевые вирусы.

Программные вирусы – это вредоносный код, который внедрен внутрь исполняемых файлов (программ), имеющие расширение .com и .exe. Вирусный код может воспроизводить себя в теле других программ – этот процесс называется размножением.

После определенного времени, создав достаточное количество копий, программный вирус может перейти к разрушительным действиям – удалять информацию, которая хранится на жестком диске, что приводит к нарушению работы программ и операционной системы. Этот процесс называется **вирусной атакой**.

Загрузочные вирусы поражают не программные файлы, а загрузочный сектор магнитных носителей (жестких дисков).

Макровирусы поражают документы, созданные в приложениях, и имеют средства для выполнения макрокоманд. К таким документам относятся документы текстового процессора Word, табличного процессора Excel и т.д. Заражение происходит при открытии файла документа в окне программы, если в ней не отключена возможность выполнения макрокоманд.

Сетевые вирусы пересылаются с компьютера на компьютер, используя для своего распространения компьютерные сети, электронную почту и другие каналы.

По алгоритмам работы различают следующие компьютерные вирусы:

- черви (сетевые вирусы);
- вирусы-невидимки (стелс-вирусы);
- троянские программы;
- программы-мутанты;
- логические бомбы и др.

Черви распространяются через глобальные и локальные компьютерные сети, способны к самовоспроизведению. В отличие от других вирусов червям для размножения не требуется заражать другие файлы, они просто копируют сами себя, вследствие чего могут временно выводить из строя сеть за счет интенсивного распространения. Способны испортить файлы на зараженном компьютере, организовать проведение сетевых атак и рассылку спама.

Вирусы-невидимки – это вирусы, полностью или частично скрывающие свое присутствие в системе. Такие вирусы могут оставаться незамеченными для антивирусных программ. Вирусы-невидимки постоянно находятся в оперативной памяти и изменяют информацию, загружаемую в оперативную память, что приводит к неправильной работе программ.

Трояны или **троянские кони** не размножаются. Попадая на компьютер, трояны не проявляют себя в первое время, но когда они начнут действовать, то могут просто уничтожить операционную систему. Трояны инсталлируют на зараженный компьютер другие программы, которые **способны удалять файлы**, разрушать информацию на жестком диске и обнаруживать уязвимости системы безопасности. Это дает им полный доступ к системе и позволяет внешнему пользователю копировать и пересылать конфиденциальную информацию.

Мутанты – сложно обнаруживаемые вирусы вследствие применяемых алгоритмов шифрования и модификации. При самопроизводстве мутанты создают алгоритмы шифровки-расшифровки, благодаря которым копии одного и того же вируса отличаются от оригинала, что усложняет их идентификацию и ликвидацию.

Логическая бомба – программа, которая запускается при определённых временных или информационных условиях для осуществления вредоносных действий, как правило, несанкционированного доступа к информации, искажения или уничтожения данных.

Желательно не допускать появления вирусов в ПК, но при заражении компьютера вирусом важно его обнаружить.

Признаки заражения ПК вирусом:

- медленная работа компьютера;
- зависания и сбои в работе компьютера;
- изменение размеров файлов;
- уменьшение размера свободной оперативной памяти;
- значительное увеличение количества файлов на диске;
- исчезновение файлов и каталогов или искажение их содержимого;
- изменение даты и времени модификации файлов.

Способы защиты от компьютерных вирусов:

- установить антивирусную программу и регулярно пользоваться им для проверки компьютера, своевременно обновлять базу данных антивирусной программы;- не запускать программы, полученные из Интернета или в виде вложенных файлов в сообщения электронной почты без проверки на наличие в них вируса;

- проверять все внешние диски на наличие вирусов, прежде чем копировать или открывать файлы, которые на них содержатся, или выполнять загрузку компьютера с таких дисков;

- регулярно сканировать жесткие диски в поисках вирусов. Сканирование обычно выполняется автоматически при каждом включении ПК и при размещении внешнего носителя в устройство считывания (при сканировании антивирусная программа ищет вирус путем сравнения кода программ с кодами известных ей вирусов, хранящимися в базе данных);

- регулярно архивировать файлы с целью создания резервных копий данных, что позволит минимизировать ущерб от вирусной атаки.

Антивирусные программы

Существует достаточно много программных средств антивирусной защиты. Современные антивирусные программы осуществляют:

- запуск антивирусных программ и обновления вирусной базы данных;
- проверку, проявление и удаление известных вирусов в памяти, файлах и системных областях дисков;

- выявление неизвестных вирусов;
- проверку электронной почты;
- защиту от сетевых атак.

К наиболее эффективным и популярным антивирусным программам относятся: Антивирус Касперского, AVAST, Norton AntiVirus и др.

Антивирус Касперского – это классическая защита компьютера от вирусов, троянских и шпионских программ, а также от любого другого вредоносного ПО.

Шпионское ПО – это программы, которые осуществляют сбор информации о конфигурации компьютера, деятельность пользователя, запоминают последовательность нажатия клавиш на клавиатуре во время ввода паролей и любую другую конфиденциальную информацию без согласия самого пользователя.

Основные функции:

- защита от известных и новых интернет-угроз;
- защита от вирусов, троянских программ и червей;
- защита от шпионского и рекламного ПО;
- проверка файлов, почты и интернет-трафика в режиме реального времени;
- защита от вирусов при работе с ICQ и другими IM-клиентами;
- защита от всех типов клавиатурных шпионов;
- выявление всех видов руткитов (программ для скрытия следов присутствия вредоносной программы в системе);
- автоматическое обновление антивирусных баз.

Антивирусная программа AVAST! существует в бесплатной версии, русифицирована и имеет удобный интерфейс, осуществляет защиту файловых систем, электронной почты, имеет средства автоматического обновления баз и т. д.

Программа Norton AntiVirus постоянно находится в памяти компьютера и осуществляет такие задачи как мониторинг памяти и сканирования файлов на диске.

Для обеспечения защиты компьютера от вирусов должен быть всегда включенным автозащита ПК. Автозащита работает в фоновом режиме, не прерывая работу ПК.

Автозащита автоматически:

- обнаруживает и защищает ПК от всех типов вирусов, включая макровирусы, вирусы загрузочных секторов, вирусы резидента памяти и троянских коней, червей и других вредоносных вирусов;
- защищает компьютер от вирусов, которые передаются через Интернет, проверяя все файлы, которые загружаются из Интернета.

Архивация

Архивация – это сжатие одного или более файлов с целью экономии памяти и размещение сжатых данных в одном архивном файле. **Архивация данных** – это уменьшение физических размеров файлов, в которых хранятся данные, без изменения количества информации за счет применения специальных методов кодирования информации.

Архивация производится в следующих случаях:

- когда необходимо создать резервные копии наиболее важных файлов;
- когда необходимо освободить место на диске;
- когда необходимо передать файлы по E-mail.

Архивный файл – это набор из одного или нескольких файлов, помещенных в сжатом виде в единый файл, из которого их можно при необходимости извлечь в первоначальном состоянии. Создание архивных файлов называют упаковкой, сжатием или **архивацией файлов**. Обратное действие, то есть получение файлов в первоначальном состоянии с архивных, называют распаковкой, извлечением из архива, или **разархивирование файлов**. Архивный файл имеет смысл, позволяет узнать, какие файлы содержатся в архиве.

В оглавлении архива для каждого файла, содержащегося в нем, сохраняется следующая информация:

- имя файла;
- размер файла на диске и в архиве;
- сведения о местонахождении файла на диске;
- дата и время последней модификации файла;
- код циклического контроля для проверки целостности архива;
- степень сжатия.

Любой из архивов имеет свой степень сжатия. Он вычисляется по формуле:

$$K_{сж} = V_{ис} / V_{сж},$$

где $K_{сж}$ – коэффициент сжатия; $V_{ис}$ – объем исходного файла (до сжатия) в Байт; $V_{сж}$ – объем файла после сжатия в Байт.

Коэффициент сжатия показывает, во сколько раз уменьшился размер файла после архивирования.

Для создания архивов используются специализированные программы – архиваторы (упаковщики). **Архиваторы** – это программы (комплекс программ), которые выполняют сжатие и восстановление сжатых файлов в первоначальном виде. Современные архиваторы отличаются алгоритмами, скоростью работы, степенью сжатия. В настоящее время лучшим архиватором для Windows является архиватор WinRAR.

Архиватор WinRAR

WinRAR – 32-разрядная версия архиватора RAR для Windows – мощное средство создания архивов и управления ими. Есть несколько версий RAR для различных операционных систем: Windows, Linux, UNIX и т.д.

Основные возможности WinRAR:

1. Позволяет распаковывать архивы CAB, ARJ, LZH, TAR, GZ, ACE, UUE, BZ2, JAR, ISO, и обеспечивает архивирование данных в формате ZIP и RAR.
2. Обеспечивает полную поддержку архивов ZIP и RAR.
3. Имеет специальные алгоритмы, оптимизированные для текста и графики. Для сжатия мультимедиа можно использовать только формат RAR.

4. Поддерживает технологию перетаскивания (drag&drop).
5. Имеет интерфейс командной строки.
6. Создает самораспаковывающиеся архивы, обеспечивает защиту их паролями.
7. Обеспечивает поддержку многотомных архивов, то есть осуществляет разбивку архива на несколько томов (например, для записи большого архива на диски). Расширение томов: RAR, R01, R02 и т.д. При самораспаковке архивов первый том имеет расширение EXE.
8. Имеет средства восстановления, позволяющие восстанавливать утраченные части многотомного архива. Обеспечивает восстановление физически поврежденных архивов.
9. Для начинающих имеет режим Мастер (Wizard), с помощью которого можно легко осуществить все операции над архивами.

WinRAR способен создать архив в двух разных форматах: RAR и ZIP. Рассмотрим преимущества каждого формата.

Архив в формате ZIP

Основное преимущество формата ZIP – его популярность. Например, большинство архивов в Internet – это архивы ZIP. Поэтому приложение к электронной почте лучше всего создавать в формате ZIP. Можно также создать самораспаковывающийся архив. Такой архив немного больше, но может быть извлечен без внешних программ. Другое преимущество ZIP – скорость. Архив ZIP обычно создается быстрее, чем RAR.

Архив в формате RAR

Формат RAR в большинстве случаев обеспечивает значительно лучшее сжатие, чем ZIP. Кроме того, формат RAR обеспечивает поддержку многотомных архивов, имеет средства восстановления поврежденных файлов, архивирует файлы практически неограниченных размеров. Необходимо отметить, что при работе в файловой системе FAT32 архивы могут достигать только 4 гигабайт. Работа с большими размерами архива поддерживается только в файловой системе NTFS.

Программа архивации Microsoft Backup (резервное копирование)

Запуск программы осуществляется: **Пуск - Программы - Стандартные - Службные - Архивация данных**. Откроется мастер архивации и восстановления в обычном режиме. С этого режима можно перейти в расширенный режим для работы с мастером архивации, мастером восстановления и мастером аварийного восстановления ОС.

Программа архивации позволяет защитить данные от случайной потери в случае, если в системе возникает сбой оборудования или носителя информации. С помощью Backup можно создать резервную копию данных на жестком диске, а затем создать архив на другом носителе данных. Носителем архива может быть логический диск или отдельное устройство (съемный диск).

Программа архивации создает снимок состояния, является точной копией содержимого диска на определенный момент времени, в том числе открытых файлов, используемых системой. Во время выполнения программы архивации пользователь может продолжать работать с ОС без риска потери данных.

Программа архивации предоставляет следующие возможности:

1. Архивацию выбранных файлов и папок на случай сбоя жесткого диска или случайного удаления файлов (архивировать можно на жесткий диск или съемный диск). Backup восстанавливает файлы и папки, которые архивируются на жесткий диск.

2. Архивацию данных состояния системы. Программа позволяет архивировать и восстанавливать копии важных системных компонентов, таких как реестр, загрузочные файлы и база данных службы каталогов.

Лекция 6

ТАБЛИЧНЫЙ ПРОЦЕССОР MICROSOFT EXCEL

Общие сведения о табличных процессорах


Для представления данных в удобном виде очень часто используют таблицы. Программное обеспечение компьютера позволяет представлять таблицы в электронной форме, а это дает возможность не только отражать, но и обрабатывать данные.

Класс программ, используемых для обработки таблиц, называется **электронными таблицами (ЭТ)**, или **табличными процессорами (ТП)**. К ним относятся VisiCalc (первая программа разработана в 1979 г.), *Super Calc*, *Lotus 1-2-3*, *Microsoft Excel*, *OpenOffice.org Calc*, АБАК и др.

Особенности ЭТ заключаются в возможности использования формул для описания связи между значениями различных ячеек. Расчет по формулам выполняется автоматически. Изменение содержания любой ячейки приводит к пересчету значений всех ячеек, которые с ней связаны формульными отношениями и, тем самым, к обновлению всей таблицы в соответствии с изменением данных.

Использование ЭТ упрощает работу с данными и позволяет получать результаты без проведения расчетов вручную или специального программирования.

ЭТ позволяют решать инженерные, экономические, бухгалтерские задачи, обрабатывать статистические, банковские данные. ЭТ широко используются в делах менеджмента, маркетинга и других отраслей знаний.

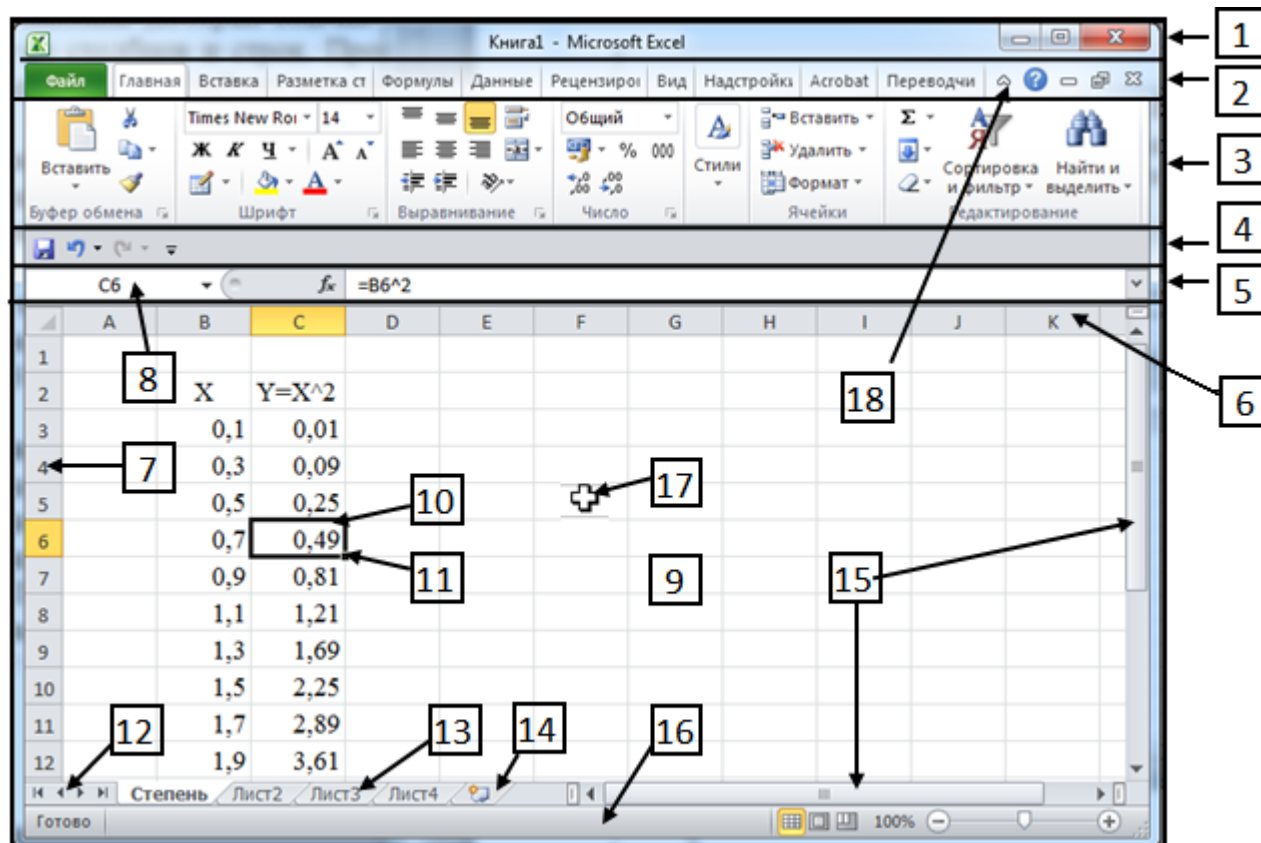
Мы будем рассматривать и изучать ТП Microsoft Excel. Excel входит в состав пакета Microsoft Office. Запуск приложения выполняется командой "**Пуск** → **Все программы** → **Microsoft Office** → **Microsoft Excel**" или щелчком по пиктограмме ярлыка Microsoft Excel .

После запуска Excel на рабочем столе появится окно приложения Microsoft Excel, которое имеет структуру типичного окна приложения Windows.

Интерфейс программы MS Excel 2010

Окно табличного процессора Excel 2010 имеет типичную структуру окна

приложения.

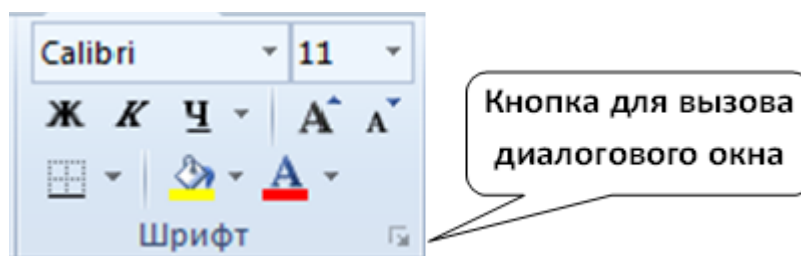


Основные элементы окна:

1. Строка заголовка – включает: системное меню; название документа; название программы, которой принадлежит это окно; элементы управления (кнопки **Свернуть**, **Свернуть в окно/Развернуть**, **Заккрыть**).

2. Строка вкладок – каждая вкладка предназначена для решения конкретной задачи. Если нажать (щелкнуть ЛКМ) название вкладки открывается строка перечня групп данной вкладки (лента). **3. Лента (строка перечня групп открытой текущей вкладки).**

Группа – объединение кнопок для решения задачи какого-либо направления. На каждой вкладке решение задач разбивают на ее составляющие и объединяют в группы. Группы в свою очередь состоят из команд, которые обозначены кнопками и реализуют выполнение действий с объектами. Названия групп расположены в нижней части строки групп. С правой стороны от названия группы расположены значки со стрелками – это кнопки вызова диалоговых окон.



Кроме стандартного набора вкладок могут появляться вкладки, которые называют **Контекстными инструментами**. Их появление связано с задачей, которую необходимо выполнить, например, необходимо изменить размеры рисунка, контрастность, художественные эффекты. При выделении рисунка поверх вкладок появляется вкладка **Контекстный инструмент**, которая носит название "**Работа с рисунками, Формат**".

4. Панель быстрого доступа по умолчанию расположена в верхней части окна Excel. Она предназначена для размещения команд, часто используемых пользователем. Может настраиваться пользователем, путем добавления или удаления в нее новых команд.

5. Строка формул – предназначена для отображения содержимого текущей ячейки, ввода и редактирования формул.

6. Строка имен столбцов – предназначена для отображения имен столбцов. Указывает местонахождение ячеек по горизонтали.

7. Столбец имен (номеров) строк – предназначен для отображения имен строк. Указывает местонахождение ячеек по вертикали.

8. Поле отображения имени текущей ячейки. Может использоваться для задания ячейке нового имени. Для этого необходимо щелкнуть два раза ЛКМ (в поле появится курсор ввода), удалить старое имя, ввести новое и нажать кнопку Enter.

9. Рабочее поле листа представляет собой поле, состоящее из ячеек, образованных пересечением столбцов и строк. В рабочем поле выполняются все действия пользователя с данными.

10. Курсор – прямоугольник, обрамляющий текущую ячейку. Перемещение курсора по рабочему полю выполняется клавишами перемещения курсора или щелчком ЛКМ по необходимой ячейке.

11. Маркер автозаполнения – предназначен для копирования формул и содержимого ячейки в смежные ячейки.

12. Кнопки перехода – с левой стороны на первый, или предыдущий листы; с правой – на следующий, или последний лист.

13. Ярлычки листов – ярлык текущего листа располагается поверх остальных и выделен светлым цветом.

14. Кнопка вставить лист – предназначена для вставки нового листа в таблицу.

15. Полосы прокрутки – используются для просмотра части рабочего поля листа, которая находится за пределами размера окна.

16. Строка состояния – горизонтальная строка в нижней части книги. В строке состояния отображаются данные о текущем состоянии содержимого окна.

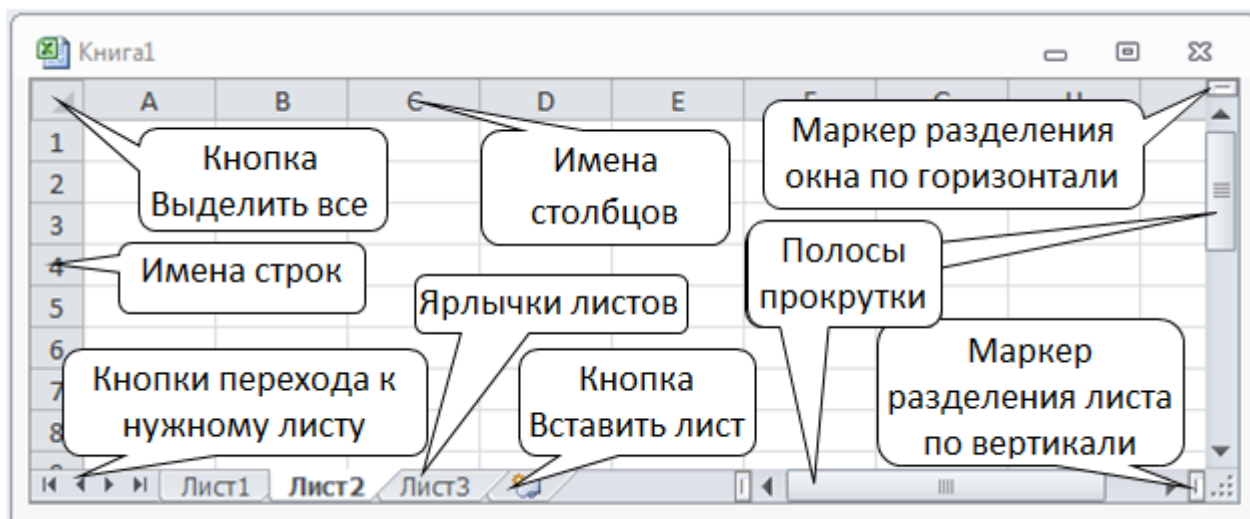
17. Указатель мыши – показывает визуально местоположение указателя мыши на рабочем поле.

18. Кнопка Свернуть/Развернуть ленту – разворачивает или сворачивает ленту, содержащую инструменты групп.

Документ Excel называется **рабочей книгой**. Рабочая книга состоит из набора рабочих листов, каждый из которых имеет табличную структуру и может

содержать одну или несколько таблиц.

Окно рабочей книги занимает большую часть окна программы Excel и предназначено для отображения содержимого рабочих листов. Содержание и структура рабочей книги показаны на рисунке.



Маркеры разделения листов – предназначены для разделения рабочего поля листа на две части либо по горизонтали, либо по вертикали, либо одновременно по вертикали и горизонтали. При этом Excel позволяет работать с каждой частью отдельно. Разделение выполняется перетаскиванием маркера разделения ЛКМ. Отмена разделения также выполняется перетаскиванием мышью границ разделения.

Каждый рабочий лист имеет название, оно отображается на ярлычке листа. Ярлычок листа находится в нижней части книги. С помощью ярлычка можно переключаться на другой рабочий лист текущей книги (щелкнуть ЛКМ по ярлычке листа). Чтобы переименовать рабочий лист необходимо щелкнуть два раза ЛКМ на его ярлычке, появляется курсор ввода, Вам необходимо удалить старое имя и ввести новое.

Основные понятия и определения ТП Excel

Рабочий лист состоит из строк и столбцов.

Столбцы – вертикально расположенные полосы, идентифицированы большими буквами латинского алфавита (A, B, C, ..., Z) или комбинациями из двух - трех латинских букв (AA, AB, ..., AZ, BA, BB, ..., BZ, ..., ZZ, AAA, AAB ..., AAZ, ABA, ABB, ..., XFD). Имена столбцов находятся в верхней части таблицы. Всего рабочий лист может содержать 16384 столбца (2^{14}).

Строки – горизонтально расположенные полосы, идентифицированы числами (1, 2, 3 ...), находящимися в крайней левой части таблицы. Строки имеют последовательную нумерацию от 1 до 1048576 (2^{20}).

Ячейка – место пересечения столбца и строки, имеет вид прямоугольника. Каждая ячейка имеет свой уникальный адрес, который состоит из имени столбца и

номера строки, например: B2, A12, P43, AZ42, BAZ1267. Адрес ячейки определяет местонахождение (расположение) ячейки.

По умолчанию адрес ячейки является ее именем. В ЭТ можно работать как с отдельными ячейками, так и с группами ячеек, образующими диапазон (блок) ячеек. При записи диапазона имена ячеек разделяются оператором диапазона (двоеточием), например: A1:D6. Диапазон ячеек может быть представлен в виде части строки A3:G3, в виде части столбца F2:F10, а также в виде прямоугольника. В последнем случае указывается верхняя левая ячейка прямоугольника и нижняя правая, например: B3:G12. Если необходимо обратиться к нескольким несмежным ячейкам, их имена перечисляют с помощью оператора объединения (точка с запятой), например: A3;B12 или F2:F10;G3:G11.

Текущая ячейка – это ячейка, с которой работает пользователь в настоящее время. Она выделена курсором в виде рамки, расположенной вокруг ячейки. Внизу с правой стороны рамки находится небольшой квадрат, который называют маркером автозаполнения.

Маркер автозаполнения используют для быстрого заполнения таблиц. Excel может автоматически продолжать ряд чисел, комбинаций чисел и текста, дат и времени по заданному образцу. Для быстрого заполнения ячеек данными нужно выделить диапазон ячеек и переместить маркер автозаполнения в нужную сторону (т.е. заполняется строка или столбец). Также с помощью маркера автозаполнения можно быстро заполнить диапазон ячеек последовательностью чисел, дат или элементов встроенных списков, таких как дни недели, месяцы или годы.

Со строками и столбцами можно выполнять следующие действия:

Изменять размер столбцов и строк – для этого необходимо указатель мыши установить на правую границу столбца или нижнюю границу строки и перетащить мышью влево или вправо (вниз или вверх) в зависимости от того, что необходимо сделать (уменьшить размер столбца (строки) или увеличить).

Выделять столбцы (строки) – для этого необходимо щелкнуть мышью по имени столбца (строки). Выделенные столбцы (строки) отмечаются цветом.

Удалять столбцы (строки) – для этого столбцы (строки) необходимо выделить, а затем выполнить команду Главная → Ячейки → Удалить. При этом удаляются данные, находившиеся в этом столбце (строке), а данные, которые размещались в смежном с правой стороны столбце (в нижней строке), перемещаются в удаленный столбец (строку).

Очищать столбцы (строки) – для этого столбцы (строки) необходимо выделить, а затем выполнить команду Главная → Редактирование → Очистить. Данная команда позволяет удалять данные из столбцов (строк) и оставляет очищенные столбцы (строки) пустыми. При этом нужно выбрать необходимый вариант очищения, например: очистить только данные (аналогичный результат получается при нажатии клавиши [Delete]) или очистить только форматы и т.д.

Вставлять столбцы (строки) – для этого необходимо выделить столбец (строку) перед которым мы хотим вставить новый, "чистый" столбец (строку) и выполнить команду Главная → Ячейки → Вставить → Вставить столбцы на

лист (**Вставить строки на лист**). Все данные в столбцах смещаются вправо (в строках – вниз) и появляется новый, пустой столбец (строка) с тем же именем.

Копировать данные столбцов (строк) – для этого выделить столбец (строку), из которого копируются данные, и выполнить команду **Главная** → **Буфер обмена** → **Копировать**, потом выделить столбец (строку), в который мы хотим скопировать данные, и выполнить команду **Главная** → **Буфер обмена** → **Вставить**. Эти же действия можно выполнить, используя комбинацию клавиш клавиатуры [**Ctrl+C**], а затем [**Ctrl+V**], или команды контекстного меню **Копировать - Вставить**.

Перемещение данных столбцов (строк) выполняется как и копирование, но при этом используются: команды **Главная** → **Буфер обмена** → **Вырезать**, **Главная** → **Буфер обмена** → **Вставить**, комбинация клавиш клавиатуры [**Ctrl+X**], [**Ctrl+V**] или команды контекстного меню **Вырезать - Вставить**.

Выделять, копировать, перемещать, удалять данные можно в нескольких столбцах (строках) одновременно. Для этого их необходимо выделить. Чтобы выделить несколько столбцов (строк) одновременно необходимо, удерживая на клавиатуре клавишу [**Ctrl**], щелкать мышью по заголовкам столбцов (строк), а затем выполнить необходимую команду, как это описано выше.

Все перечисленные выше операции можно выполнять также с отдельными ячейками ЭТ или с диапазонами ячеек.

Типы данных

Данные используемые в Excel могут быть следующих типов:

1. Текст – это любая последовательность символов, как правило, используется для оформления заголовков таблиц, столбцов или строк, а также для вывода комментариев.

2. Числовые данные – это целые или действительные числа, которые могут быть отрицательными или положительными. В действительных числах, как правило, целая часть числа от десятичной отделяется запятой (,). Например: 8; 78; -3,67; 45,92. Действительные числа могут быть представлены в экспоненциальной (научной) форме, например:

$378,982 \cdot 10^3 \rightarrow 378,982E3$;
 $-567845,34 \cdot 10^{-4} \rightarrow -567845,34E-4$.

Программа Excel сохраняет только первые 15 цифр числа. Последние цифры превращаются в нули об этом необходимо помнить при работе с числами, которые имеют значимость более 15 цифр. Например: число 12345678901234567890 будет храниться как 12345678901234500000. Точно так будут представлены и очень малые числа: $0,12345678901234567890 \rightarrow +0,112345678901234500000$.

3. Дата и время

Дату можно вводить в любом из следующих форматов: 8 февраля 2013; 8 февр. 13; 8.2.13; 8/2/13; 08/02/13.

Время может вводиться в следующих форматах: 1:00; 2:30 AM; 5:00:01PM. AM – от 0 до 12⁰⁰ дня. PM – от 12⁰⁰ до 24⁰⁰.

Дата и время в Excel могут быть представлены как числовые данные в виде

действительных чисел и над ними допускается производить арифметические вычисления.

4. Формула – это последовательность аргументов, объединенных знаками арифметических или логических операций, начинающаяся со знака равно (=).

Формула показывает, что и в какой последовательности нужно вычислять (обрабатывать). В качестве аргументов могут быть использованы текст, числа, имена ячеек, другие формулы, встроенные (стандартные) функции. Если в формулах используются имена ячеек, их называют ссылками на ячейки.

Операции в Excel

В Excel используются следующие арифметические операции:

- + сложение;
- вычитание;
- * умножение;
- / деление;
- ^ возведение в степень.

Логические операции:

- > больше;
- < меньше;
- >= больше или равно;
- <= меньше или равно;
- = равно;
- <> не равно.

Порядок вычислений в формуле:

1. встроенные функции;
2. возведение в степень;
3. умножение или деление;
4. сложение или вычитание;
5. логические операции (операции сравнения).

Для изменения порядка выполнения операций в формулах используются круглые скобки. Сначала вычисляются операции во внутренних скобках и т.д. Количество скобок, которые открываются, должно равняться количеству скобок, которые закрываются.

Порядок вычисления формул в ячейках Excel определяет автоматически.

Функция – небольшая программа, реализующая вычисления, наиболее часто используемые пользователями, например: вычисления логарифмов, тригонометрических функций и т.п.

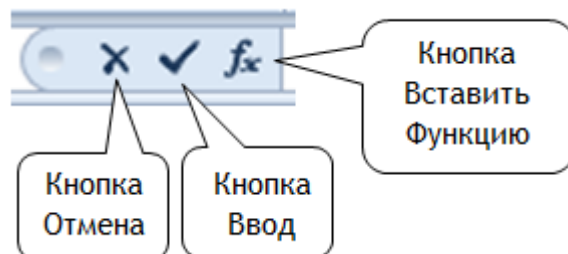
Лекция 7

ФОРМУЛЫ И ФУНКЦИИ В EXCEL

Ввод формул в Excel. Содержание ячейки, значение ячейки

Формула – это последовательность аргументов, объединенных знаками арифметических или логических операций, начинающаяся со знака равно (=).

Содержание ячейки – это данные, которые пользователь вводит в ячейку (текст, числовые данные, формулы, функции, изображения и т. д.). Данные можно вводить в ячейку с помощью клавиатуры, они будут отображаться в текущей ячейке и в строке формул, а в поле управления строки формул появятся кнопки управления.



Кнопка **Отменить** – предназначена для отмены ввода или редактирования данных, дублируется клавишей [Esc].

Кнопка **Ввод** – предназначена для ввода данных в ячейку, дублируется клавишей [Enter].

Кнопка **Вставить функцию** – предназначена для ввода или редактирования стандартных функций, используемых в Excel (вызов Мастера функций).

Для сохранения набранных данных в ячейке необходимо щелкнуть мышью за пределами этой ячейки или нажать клавишу [Enter], или щелкнуть по кнопке **Ввод**.

Значение ячейки – это результат обработки Excel содержания ячейки. В результате обработки содержания ячейки, значение ячейки может быть: текст, число, логическое значение 1 либо 0 (истина или ложь) или значение ошибки.

Копирование формул, относительные и абсолютные адреса ячеек

При копировании формул пользователь может использовать любой из следующих способов копирования.

Первый способ – с помощью команд вкладки Главная.

Выделить ячейку с формулой или диапазон ячеек и выполнить команду: Главная → Буфер обмена → Копировать, формула будет помещена в буфер обмена. Затем выделить ячейку или диапазон ячеек, куда мы хотим поместить эту формулу, и выполнить команду Главная → Буфер обмена → Вставить.

Второй способ – используя комбинацию клавиш клавиатуры.

Выделить ячейку (диапазон ячеек) и нажать комбинацию клавиш клавиатуры [Ctrl]+[C], затем выделить ячейку или диапазон ячеек, куда мы хотим скопировать формулу, и нажать комбинацию клавиш [Ctrl]+[V].

Третий способ – используя маркер автозаполнения.

Если необходимо скопировать формулу в одну или несколько смежных ячеек, необходимо навести указатель мыши на маркер заполнения, чтобы получить указатель мыши в виде тонкого крестика (+), нажать левую кнопку мыши (ЛКМ) и переместить маркер заполнения в одну, или в несколько смежных ячеек. Формулы будут скопированы в указанные ячейки.

Четвертый способ – используя контекстное меню. Выделить ячейку с формулой или диапазон ячеек и щелкнуть ПКМ, в контекстном меню щелкнуть кнопку **Копировать**, формула будет помещена в буфер обмена. Затем выделить ячейку или диапазон ячеек, куда мы хотим поместить эту формулу, щелкнуть ПКМ и в контекстном меню щелкнуть кнопку **Вставить**.

Особенности копирования формул заключаются в том, что Excel при копировании редактирует формулы таким образом, чтобы они сохраняли свою суть и в новой позиции. Например, необходимо вычислить сумму выделенных средств для каждого региона (строка **Всего** на рисунке). Формула для вычисления суммы вводится в одну из ячеек строки **Всего**, например, в B5, а затем **копируется** из ячейки B5 в диапазон ячеек C5:E5. Скопированная формула в каждой ячейке диапазона в соответствии с новым местом нахождения ссылается на другие ячейки, чем формула в ячейке B5, но суть выполненной операции остается: формула вычисляет сумму выделенных средств для определенного региона.

		B5 fx =B2+B3+B4			
	A	B	C	D	E
1	Работы	Север	Юг	Восток	Запад
2	Ремонт	32	20	35	40
3	Реконструкция	40	60	100	55
4	Новые дороги	100	70	50	120
5	Всего	=B2+B3+B4	=C2+C3+C4	=D2+D3+D4	=E2+E3+E4

При ссылках ячейки в формулах Excel используются понятия относительных и абсолютных адресов ячеек.

Относительные адреса ячеек при копировании формулы изменяются в соответствии с новым местоположением формулы. Примеры относительных адресов: A23, B2, FG34 и т.д.

Пример изменения имен ячеек при копировании формулы с относительными адресами: необходимо рассчитать формулу $Y=X^2$, где X изменяется от 1 до 2 с шагом 0,2:

		B7 fx =A7^2	
	A	B	
1	X	Y=X^2	
2	1	=A2^2	
3	1,2	=A3^2	
4	1,4	=A4^2	
5	1,6	=A5^2	
6	1,8	=A6^2	
7	2	=A7^2	

Абсолютные адреса – это адреса ячеек, которые при копировании формул не меняют своего адреса или изменяют его частично. Для того, чтобы адреса ячеек не изменялись при копировании формул используется специальный формат

записи адресов ячеек или ячейкам присваиваются имена, созданные пользователем.

Можно использовать следующие форматы записи абсолютных адресов:

Абсолютный адрес строки. При копировании не изменяется только номер строки. Пример записи: B\$2.

Абсолютный адрес столбца. При копировании не изменяется только имя столбца. Пример записи: \$B2.

Абсолютный адрес ячейки. При копировании не изменяется имя столбца и номер строки. Пример записи: \$B\$2.

На рисунке приведен пример записи различных форматов абсолютных адресов ячеек в формуле и результаты копирования формул. Столбец В – копирование формул с относительными адресами. Столбец С – копирование формул с абсолютным адресом ячейки. Столбец D – копирование формул с абсолютным адресом строки.

	A	B	C	D
1	X	Y=X^2	Z=Y+Конст	F=Z+Конст
2	1	=A2^2	=B2+\$B\$8	=C2+B\$8
3	1,2	=A3^2	=B3+\$B\$8	=C3+B\$8
4	1,4	=A4^2	=B4+\$B\$8	=C4+B\$8
5	1,6	=A5^2	=B5+\$B\$8	=C5+B\$8
6	1,8	=A6^2	=B6+\$B\$8	=C6+B\$8
7	2	=A7^2	=B7+\$B\$8	=C7+B\$8
8	Конст	14,5		

Использование имен, созданных пользователем

У пользователя Excel есть возможность присвоить ячейкам имена по своему желанию, например, чтобы они несли смысловую нагрузку. Если ячейке A1 присвоить имя **Ширина**, ячейке A2 – имя **Длина**, ячейке A3 – имя **Площадь**, тогда формулу в ячейке **Площадь** можно записать: =Длина*Ширина, т.е. для использования в формулах имен ячеек необходимо поместить имя ячейки в формулу вместо адреса ячейки.

	Площадь				
	A	B	C	D	E
1	16				
2	23				
3	368				

В именах ячеек можно использовать символы: . (точка), \ (обратный слеш), _ (подчеркивание), буквы и цифры. Имена, присвоенные пользователями, не должны совпадать с именами адресов, которые уже используются в Excel, например: A21, CB2, AH5 и т.д.

Способы присваивания имен ячейкам или диапазонам ячеек:

Первый способ.

Для присваивания нового имени ячейке необходимо выполнить следующие действия:

- выделить ячейку или диапазон ячеек;
- щелкнуть на поле имени, оно превратится в стандартное поле редактирования;
- ввести имя, которое мы хотим присвоить ячейке (диапазону ячеек);
- нажать клавишу Enter.

Второй способ.

Использовать команду **Формулы** → **Определенные имена** → **Присвоить имя** → **Создание имени**. В диалоговом окне ввести:

- в поле **Имя:** – имя, которое присваивается ячейке;
- в поле **Область** – в какой книге будет зарегистрировано имя;
- в поле **Диапазон** – на какой части книги действительно это имя.

Текстовый оператор конкатенации

В Excel над текстовыми значениями ячеек можно выполнять операцию конкатенации. Она объединяет в одно целое два и более элемента текстовых данных. Эта операция использует оператор & и записывается как формула, например: =A1&B1 (ячейка A2 на рисунке). Результатом выполнения операции конкатенации будет значение ячейки A2 "операция конкатенации":

	A	B
1	операция	конкатенации
2	операция конкатенации	

Связующие формулы

Формулы в Excel в основном ссылаются на ячейки того же рабочего листа, на котором проводятся вычисления. Вместе с тем Excel позволяет использовать в формулах значения из ячеек, расположенных на других рабочих листах и даже на других рабочих книгах.

Для создания формулы, которая будет ссылаться на ячейки другого рабочего листа, необходимо перед адресом ячейки ввести имя рабочего листа, на котором она расположена, а затем восклицательный знак.

Например, формула вычисляет произведение значения ячейки B3, расположенной на рабочем листе 4, и числа 15. Формула будет иметь вид: =Лист4!B3*15.

При создании формул, которые ссылаются на ячейки другой книги, следует указать имя книги в квадратных скобках, затем – имя листа с восклицательным знаком, а затем – имя ячейки.

Например, если для расчета необходимо сослаться на ячейку B3, которая находится на Листе4 в книге "Расходы", формула запишется в следующем виде: =[Расходы.xlsx]Лист4!B3*15.

Использование функций

Функция – небольшая программа, реализующая вычисления, наиболее часто используемые пользователями, например: вычисления логарифмов, тригонометрических функций и т.п.

Функции в формуле записываются как отдельные элементы формулы или как отдельная формула.

Правило (формат) записи функций: записывается имя функции, а за ним в круглых скобках указываются аргументы, которые отделяются друг от друга точкой с запятой (;). Если функция представлена как отдельная формула, ее запись начинается со знака равенства (=). Общий формат записи функции будет иметь вид:

Имя_функции (аргумент1; аргумент2; ...; аргументN)

В качестве аргумента функции может выступать:

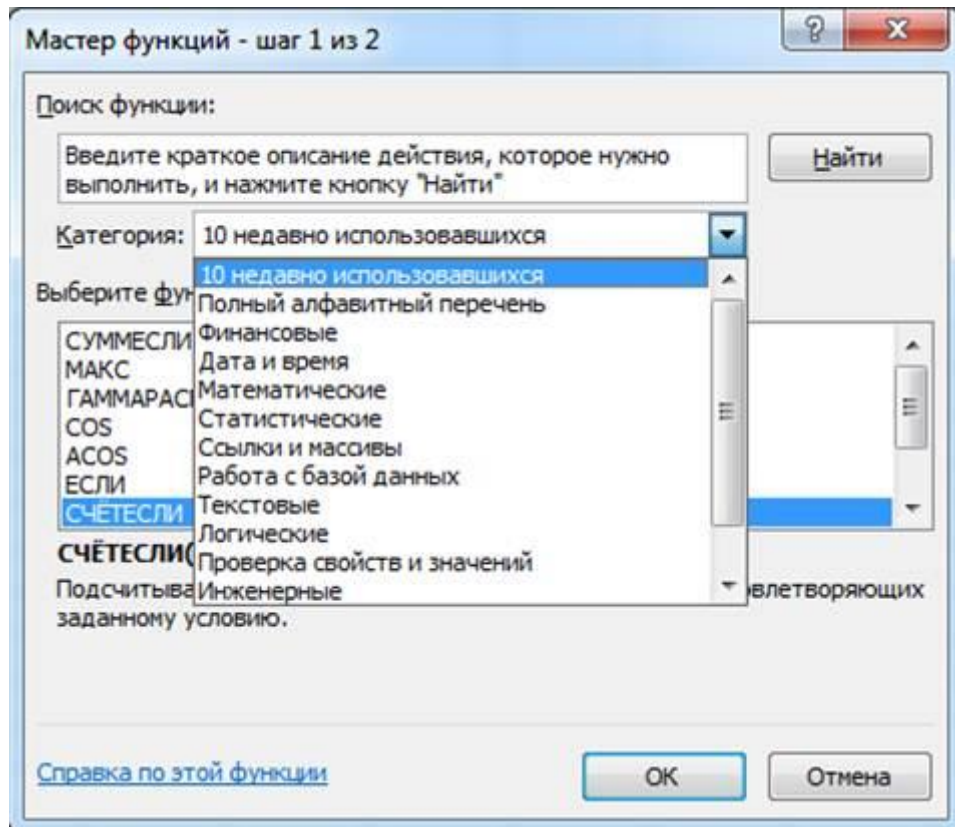
- логическое выражение (условие);
- числовые константы;
- ссылки на ячейки (диапазон ячеек);
- значения ошибок,
- другие функции.

Использование одних функций в качестве аргументов в других функциях называют вложением функций, а функции – вложенными. В Excel 2010 допускается до 64 вложений одних функций в другие. В Excel используются также функции, которые не имеют аргументов. Например, функция $\text{Пи}()$ не имеет аргумента, но правило записи функции не меняется: записывается имя функции, после которого записываются круглые скобки. Функция $\text{Пи}()$ возвращает значение числа $\pi = 3,141592654$.

Рассмотрим пример использования и записи функции СРЗНАЧ , которая используется для вычисления среднеарифметического значения диапазона чисел. Используем эту функцию для вычисления среднего балла успеваемости группы ДЭК11 по дисциплине Математика:

C9		fx =СРЗНАЧ(С3:С8)			
	A	B	C	D	E
1	Группа ДЭК11		Оценка по дисциплине		
2	№ п/п	Фамилия И.О.	Математика		
3	1	Вовчар Ч.О.	63		
4	2	Иванов А.П.	78		
5	3	Митрофанов А.С.	92		
6	4	Новоградский Ю.С.	35		
7	5	Попидгора Л.М.	80		
8	6	Шевчук М.В.	72		
9	Средний балл		70		
10					

Для удобства работы с функциями в Excel используется мастер функций. Чтобы вызвать его, необходимо выбрать (сделать текущей) ячейку, т.е. установить курсор в то место, куда будет помещаться функция, затем выполнить команду **Формула** → **Библиотека функций** → **Вставить функцию** или нажать кнопку f_x в строке формул. В результате появится диалоговое окно с заголовком **Мастер функций - шаг 1 из 2**. Вставка функции с помощью Мастера функций выполняется в двух шагах. На первом шаге окно Мастера функций имеет три поля.



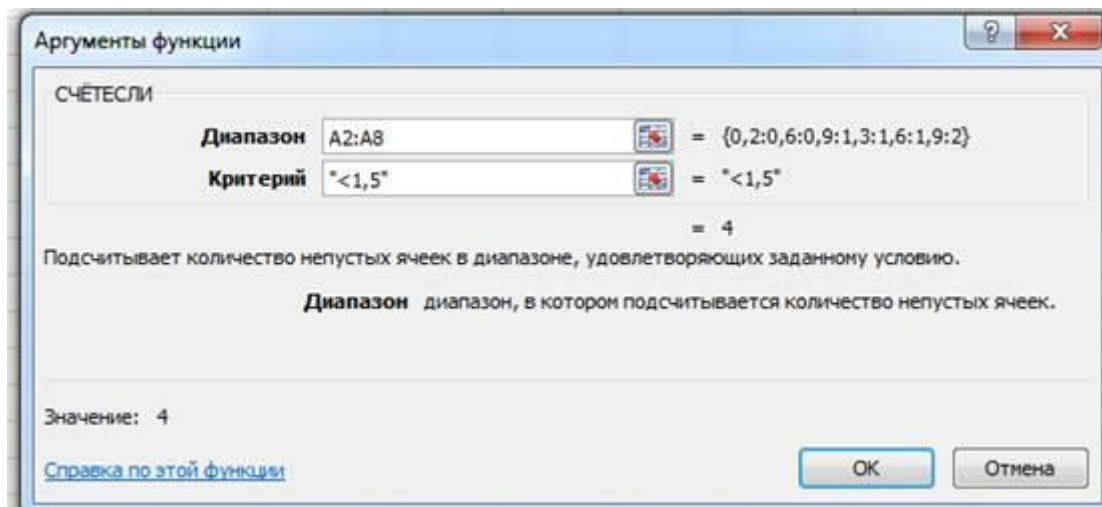
Поле **Поиск функции** используется в том случае, если пользователь не знает, какую функцию необходимо использовать для выполнения действия, где ее искать. Тогда пользователь вводит в это поле название действия, которое функция должна выполнить и нажимает кнопку **Найти**.

Поле **Категория** предназначено для выбора категории, в которой пользователь планирует найти и использовать функцию.

Поле **Выберите функцию**. После выбора категории в этом поле следует найти и выделить необходимую функцию. При этом ниже поля **Выберите функцию** появляется информация о формате записи функции и ее назначении. Если выбранная функция нас удовлетворяет, щелкаем по кнопке **ОК** и переходим ко второму шагу мастера функций:

На втором шаге мастера функций выводится диалоговое окно, которое носит название **Аргументы функции**. Это окно в зависимости от формата функции имеет несколько полей, которые предназначены для ввода аргументов. Как правило, аргументы вводятся с клавиатуры либо методом выделения мышью необходимых параметров в таблице. Таким образом, выполняется автоматический пере-

нос аргументов в диалоговое окно. При верном заполнении аргументов уже в диалоговом окне можно видеть результаты вычислений. После ввода всех аргументов - щелкнуть кнопку **ОК**. Результат вычислений появится в выделенной ячейке.



Функция может иметь несколько видов записи, например:

=СРЗНАЧ(С3:С8) – в качестве аргументов выступает диапазон ячеек;

=СРЗНАЧ(С3,С4,С5,С8) - в качестве аргументов выступает перечень ячеек;

=СРЗНАЧ(3,4,5,2,4,3) - в качестве аргументов выступают константы.

Если диапазону ячеек присвоить имя, например ОцД11, то функцию можно записать в виде: =СРЗНАЧ(ОцД11).

Особенности записи математических выражений в Excel

Математическая запись	Запись в Excel	Примечание
$\text{Cos } X$	=Cos(X)	X - числовая константа
$\text{Cos } X^2$	=Cos(X^2)	
$\text{Cos}^2 X$	=Cos(X)^2	
$\text{Cos}^3 X^2$	=Cos(X^2)^3	
$\sqrt[3]{\text{cos}^2 X^3}$	=Cos(X^3)^(2/3)	
e^X	=Exp(X)	
$\text{Lg}_3 9$	=Log(число;основание)	=Log(9;3)

Работа с массивами

Массив – это диапазон ячеек, в которых расположены данные одного типа. Массивом может быть строка, столбец, блок ячеек.

При выполнении операций с массивами используются специальный формат записи операторов.

Например, есть два массива, состоящие из пяти элементов каждый. Необходимо получить третий массив, элементы которого являются суммой соответствующих элементов двух первых.

		fx {=B2:F2+B3:F3}				
	A	B	C	D	E	F
1	Расчет расходов					
2	Сергей	2	4	6	7	8
3	Петр	3	5	3	6	7
4	Всего	5	9	9	13	15

Для создания в строке 4 итоговых значений необходимо выделить диапазон ячеек B4:F4, в который будет записан результат. В выделенный диапазон ввести формулу =B2:F2+B3:F3 и нажать комбинацию клавиш [Ctrl]+[Shift]+[Enter]. В результате Excel поместит формулу массива в каждую ячейку диапазона B4:F4. Введенная формула в ячейках диапазона B4:F4 указывает на то, что Excel должен выполнить следующие действия: B2+B3, C2+C3 и т.д.

Редактирование формул массива производится практически так же, как и другие формулы. Для редактирования формул массива необходимо щелкнуть по любой ячейке из диапазона массива с формулой, в поле ввода отредактировать формулу, после окончания редактирования формулы нажать комбинацию клавиш [Ctrl]+[Shift]+[Enter]. Изменения будут внесены во все ячейки диапазона.

Лекция 8 ПОСТРОЕНИЕ ГРАФИКОВ И ДИАГРАММ

Основные понятия

Диаграммы (от греч. *diagramma* изображение, рисунок, чертеж) являются средством наглядного представления данных и облегчают выполнение сравнений, выявление закономерностей и тенденций данных.

В программе Excel термин диаграмма используется для обозначения всех видов графического представления числовых данных. Диаграммы существенно информативнее чисел, так как они обеспечивают визуальное представление данных. Графический формат диаграммы упрощает понимание больших объемов информации и связей между различными рядами данных. Диаграмма также может давать общее представление о ситуации, позволяя анализировать представленные данные и находить важные тенденции.

При построении диаграммы используют следующие понятия: ряд данных, категории, легенда.

Ряд данных – это набор связанных между собой данных в пределах отдельной строки или столбца, отображаемых на диаграмме. На одной диаграмме можно отображать несколько рядов данных.

Категории – это аргументы функции на оси X, которые служат для упорядочения значений в рядах данных.

Легенда – это условные обозначения значений различных рядов данных на диаграмме.

Диаграмма в Excel представляет собой объект, внедренный на один из листов рабочей книги. Она может располагаться на том же листе, на котором находятся данные, или на любом другом листе (часто для отображения диаграммы отводят отдельный лист). При построении диаграммы необходимо указать область данных, по которым будет строиться диаграмма. Диаграмма сохраняет связь с данными, на основе которых она построена, и при обновлении этих данных немедленно изменяет свой вид.

Для построения диаграмм в Excel 2010 предусмотрена группа инструментов **Диаграммы**, располагающихся на вкладке **Вставка**. Большинство диаграмм строятся в прямоугольной системе координат. По горизонтальной оси X откладываются значения аргумента (категории), а по вертикальной оси Y – значения функции (ряды данных). На один рисунок может быть выведено одновременно несколько диаграмм.

При построении диаграммы в Excel 2010 появляется контекстный инструмент **Работа с диаграммами**, в состав которого входят три вкладки: **Конструктор**, **Макет**, **Формат**, которые позволяют редактировать выделенную диаграмму.

Типы диаграмм

В Excel можно строить диаграммы разных типов. Каждый тип диаграммы позволяет представлять различные данные наиболее наглядным способом. Рассмотрим основные типы диаграмм.

1. Гистограмма или столбиковая диаграмма. Являются самым распространенным типом диаграмм. Гистограмма представляет собой набор вертикальных столбиков, высота которых определяется значениями данных. В гистограммах категории обычно формируются по горизонтальной оси, а значения – по вертикальной. Гистограмма полезна для сравнения значений числовых данных в одном или нескольких рядах данных, а также для представления изменения данных во времени или по категориям.

Подтипы гистограмм:

Гистограмма с группировкой. Гистограммы с группировкой сравнивают значения по категориям и выводят их в виде плоских вертикальных прямоугольников.

На **объемной гистограмме с группировкой** данные отображаются в трехмерном виде.

Гистограмма с накоплением и объемная гистограмма с накоплением показывают отношение отдельных составляющих к их совокупному значению, сравнивая по категориям вклад каждой величины в общую сумму.

Нормированные гистограммы с накоплением и объемные нормированные гистограммы с накоплением сравнивают по категориям процентный вклад каждой величины в общую сумму.

Объемную гистограмму можно использовать для сравнения данных как по категориям, так и по рядам, поскольку диаграммы этого типа показывают катего-

рии вдоль горизонтальной оси и вдоль оси глубины, а по вертикальной оси выводятся значения.

Для цилиндрических, конических и пирамидальных диаграмм доступны те же типы представлений (с группировкой, с накоплением, нормированная с накоплением и объемная), что и для прямоугольных гистограмм. Они показывают и сравнивают данные аналогичным образом. Единственное различие заключается в том, что эти типы диаграмм вместо прямоугольников содержат цилиндрические, конические и пирамидальные фигуры.

2. Линейчатая диаграмма – это гистограмма, столбики которой располагаются не вертикально, а горизонтально.

3. Графики показывают тенденции изменения со временем значений рядов данных. На графиках данные категорий равномерно распределяются вдоль горизонтальной оси, а все значения равномерно распределяются вдоль вертикальной оси. Для некоторых видов графиков значения данных изображаются в виде маркеров.

4. Круговая диаграмма строится по одному ряду числовых данных и показывает долю каждого числового значения в сумме значений. Круговые диаграммы показывают вклад каждой величины в общую сумму в двухмерном или трехмерном виде.

Вторичная круговая диаграмма и вторичная гистограмма применяются, чтобы детализировать фрагмент круговой диаграммы. Основная проблема их построения заключается в расположении данных.

Разрезанная круговая диаграмма и объемная разрезанная круговая диаграмма демонстрирует вклад каждого значения в общую сумму, подчеркивая отдельные значения.

5. Диаграммы с областями подчеркивают величину изменений с течением времени и используются для привлечения внимания к суммарному значению в соответствии с тенденцией.

6. Точечная диаграмма показывает отношения между числовыми значениями в нескольких рядах данных. Точечная диаграмма имеет две оси значений, при этом одни значения выводятся вдоль горизонтальной оси (оси X), а другие – вдоль вертикальной оси (оси Y). На точечной диаграмме эти значения объединяются в одну точку и выводятся через неравные интервалы. Точечные диаграммы обычно используются для иллюстрации и сравнения числовых значений.

Точечные диаграммы с маркерами позволяют сравнивать пары значений.

На **точечных диаграммах с плавными линиями** точки соединены сглаживающими линиями. Такие линии могут отображаться с маркерами или без них. Сглаживающую кривую без маркеров следует использовать, если точек данных достаточно много.

На **точечных диаграммах с прямыми линиями и точечных диаграммах с прямыми линиями и маркерами** точки данных соединяются прямыми линиями.

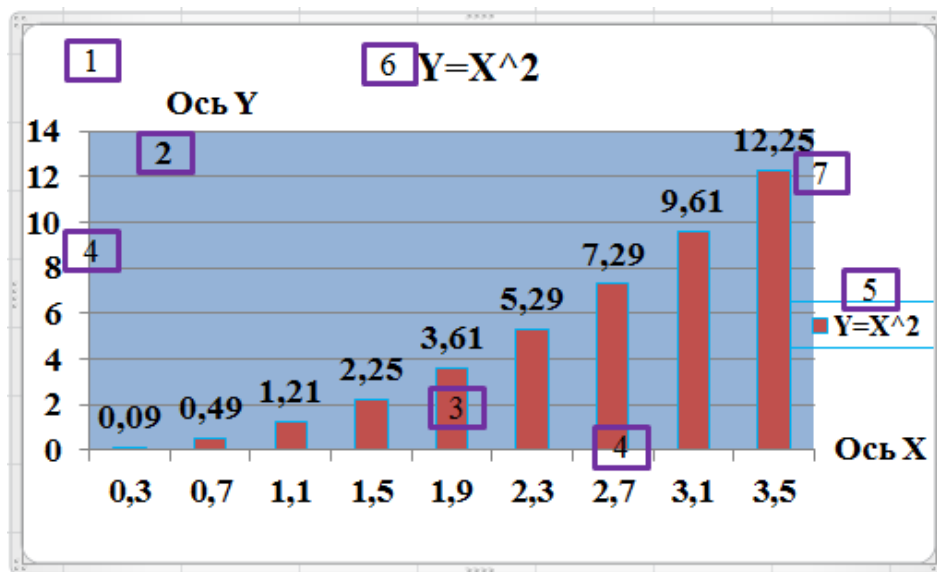
7. Биржевые диаграммы чаще всего используются для иллюстрации изменений цен. Для создания биржевой диаграммы необходимо правильно упорядочить данные.

8. В поверхностных диаграммах показаны изменения значений по двум измерениям в виде непрерывной кривой. Цветовые полосы в поверхностной диаграмме не отражают наборы данных – они указывают различие между значениями. В диаграмме такого типа используется трехмерное представление данных, которое можно представить в виде резинового листа, натянутого поверх столбцов объемной гистограммы. Как правило, такие диаграммы используются для отображения отношений между большими объемами данных, которые трудно показать иным способом.

9. На кольцевых диаграммах данные отображаются в виде колец, каждое из которых представляет ряд данных.

Элементы диаграмм

Все диаграммы в Excel состоят из отдельных элементов. Некоторые из них отображаются по умолчанию, другие можно добавлять по мере необходимости. Рассмотрим основные элементы диаграмм на примере гистограммы.



1. Область диаграммы – область размещения диаграммы и всех ее элементов.

2. Область построения. На плоских диаграммах – ограниченная осями область, содержащая все ряды данных. На объемных диаграммах – ограниченная осями область, содержащая все ряды данных, имена категорий, подписи делений и названия осей диаграммы.

3. Элементы (точки) данных – отдельные значения, отображаемые на диаграмме в виде полос, столбцов, линий, секторов, точек или других объектов, называемых маркерами данных. Маркеры данных одного цвета образуют ряды данных.

4. Ось – линия, ограничивающая область построения диаграммы, используемая в системе координат в выбранных единицах измерения. Ось Y (ось значений) обычно расположена вертикально, по данным оси Y строится диаграмма. Ось X (категорий) обычно расположена горизонтально.

5. Легенда диаграммы – рамка, в которой определяются способы обозначений и цвета рядов или категорий данных на диаграмме.

6. Названия диаграммы и осей – описательный текст, поясняющий отображаемые на диаграмме данные.

7. Метки данных – подпись с дополнительными сведениями о маркере данных, представляющем одну точку данных или значение ячейки листа.

Лекция 9. АЛГОРИТМИЗАЦИЯ. СРЕДА VB

Алгоритмизация вычислительного процесса

Алгоритм – понятное и точное предписание исполнителю, совершить последовательность действий, направленную на достижение поставленной цели или решение поставленной задачи.

Своим происхождением слово алгоритм обязано древнегреческому математику Евклиду (3 в. до н.э.), который называл так сформулированные им правила нахождения наибольшего общего делителя двух чисел (алгоритм Евклида) в честь узбекского математика Аль-Хорезми. За 400 лет до этого Аль-Хорезми (787-860 р. до н.э.) сформулировал правила выполнения четырех арифметических действий над многозначными числами.

Любой алгоритм должен обладать следующими свойствами:

Дискретность – алгоритм должен представлять собой последовательность выполнения простых, заранее определенных шагов. Для выполнения каждого шага алгоритма требуется определенный отрезок времени, т. е. получение результата на основании исходных данных происходит дискретно.

Определенность – алгоритм совершенно точен, однозначен и понятен для исполнителя. Его неоднократное применения к одним и тем же начальным условиям приводит к одним и тем же результатам.

Формальность – процесс выполнения алгоритма носит механический характер, что позволяет применять для выполнения алгоритма различные машины и механизмы.

Результативность – при выполнении алгоритма обязательно получается предусмотренный результат за определенное конечное количество шагов.

Массовость – возможность применения конкретного алгоритма при решении многих задач данного типа.

Типы алгоритмов

Линейные алгоритмы – это алгоритмы, у которых все действия выполняются последовательно одно за другим. При этом все действия выполняются только один раз.

Разветвляющиеся алгоритмы – это алгоритмы, у которых кроме последовательности действий есть операции условия. Условие поставлено в виде вопроса таким образом, что на него можно ответить однозначно да или нет.

Циклические алгоритмы – это алгоритмы, у которых одна и та же последовательность действий выполняется несколько раз, причем при каждом новом выполнении этой последовательности изменяется один или несколько исходных данных (параметров) и выходной результат.

Циклические алгоритмы делятся на **арифметические и итерационные**.

Арифметические – это алгоритмы, у которых заранее известно или может быть высчитано количество повторений.

Например, налить десятилитровое ведро литровой кружкой.

Итерационные – это алгоритмы, у которых действия выполняются до указанного результата.

Способы описания алгоритмов

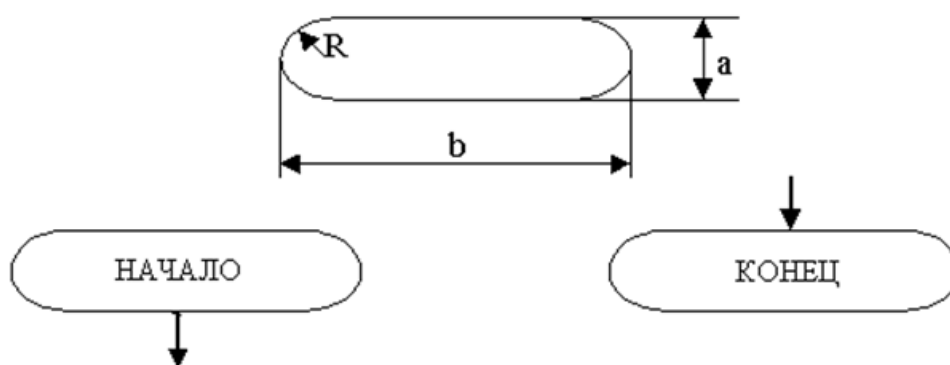
Словесный – описание с помощью текста. Например, инструкция использования телефона-автомата.

Математический – описание при помощи математических формул.

Графический – описание с помощью графических схем и специальных геометрических фигур, которые называются блоками. Каждый блок означает определенное действие.

Основные блоки схем алгоритмов

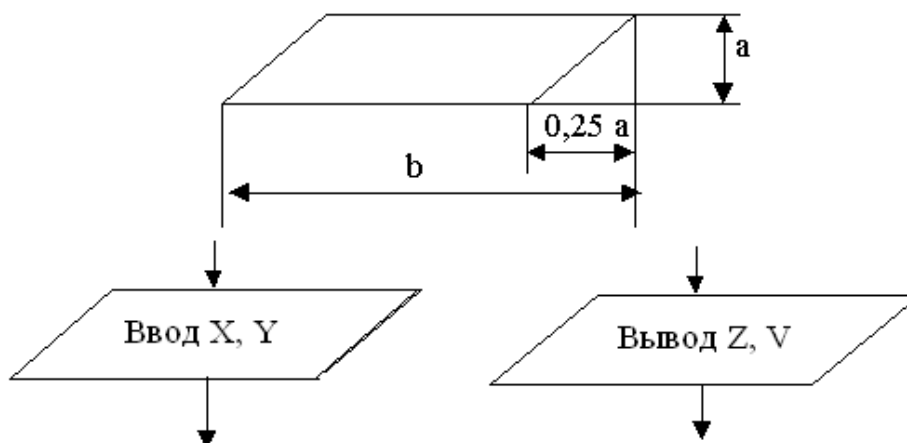
Блоки начала и конца алгоритма. Любой алгоритм начинается с блока НАЧАЛО и заканчивается блоком КОНЕЦ. Начало имеет один выход, конец – один вход.



Размер "a" должен выбираться из ряда 10, 15, 20, 25 мм. И т.д. Размер $b = 2 \cdot a$; $R = 2/a$.

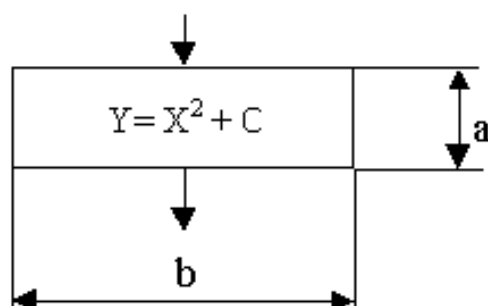
Блок ввода-вывода. Изображаются одинаково в виде параллелограмма с одним входом и одним выходом.

Блок ввода предназначен для присваивания значений переменным во время выполнения программы. В блоке ввода пишется слово "Ввод" и перечень переменных, которым необходимо присвоить значения.

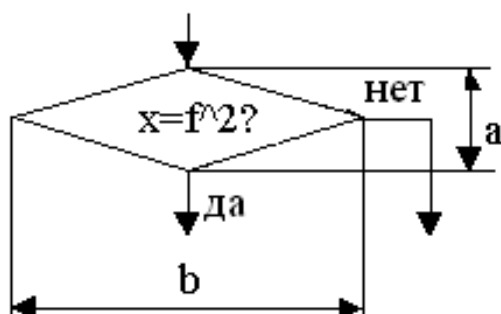


Блок вывода – для вывода данных на видеомонитор или на печать во время выполнения программы. В блоке вывода пишется слово "Вывод" и перечень переменных, значения которых необходимо вывести.

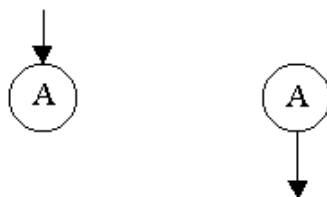
Вычислительный блок. Предназначен для записи выражений, значения которых необходимо вычислить.



Логический блок (блок проверки условия) предназначен для реализации разветвляющихся вычислительных процессов. В этом блоке записывается условие, на которое можно ответить только «да» или «нет». Если условие выполняется, то последующее выполнение алгоритма идет по ветви «да», если не выполняется – по ветви «нет». Блок имеет один вход и два выхода.



Разрыв линии связи алгоритма для переноса на другую страницу или продолжение линии связи после разрыва выполняется в виде **блоков разрыва и продолжения связи**.



Среда программирования VB

Алгоритмический язык Basic создан в 1963 году работниками Дартмудского колледжа Джоном Кеменем и Томасом Курцем как универсальный язык программирования для начинающих. Сначала он предназначался для быстрого написания программ специалистами разных предметных отраслей (инженерами, математиками, экономистами и др.), которые не являлись профессиональными программистами. На протяжении своей истории Basic дополнялся новыми инструментами средствами программирования. Вершиной развития языка стал VB, в котором собственно программирование перешло на второй план, уступив визуальному проектированию. VB используется для решения вычислительных задач, работы с базами данных, с мультимедийными средствами (графикой, звуком, мультипликацией), для автоматизации офисных работ, решения задач в Интернете.

Процесс программирования на VB включает в себя 4 этапа:

- 1) разработка и создание экранной формы;
- 2) разработка и ввод программного кода в компьютер;
- 3) запуск программы на выполнение и отладка программного кода, если есть ошибки;
- 4) сохранение разработанного приложения и преобразование проекта в exe-файл.

Для запуска среды VB необходимо выполнить команды: **Пуск ► Программы ► Microsoft Visual Studio 6.0 ► Microsoft Visual Basic 6.0** или щелкнуть по соответствующей пиктограмме на Рабочем столе или на панели быстрого запуска.

При стандартной настройке пакета на экране появится окно **New Project** (Новый проект). В нем содержатся 3 вкладки: **New, Existing, Recent**.

Вкладка **New** (Новый) содержит пиктограммы для создания новых проектов различных видов.

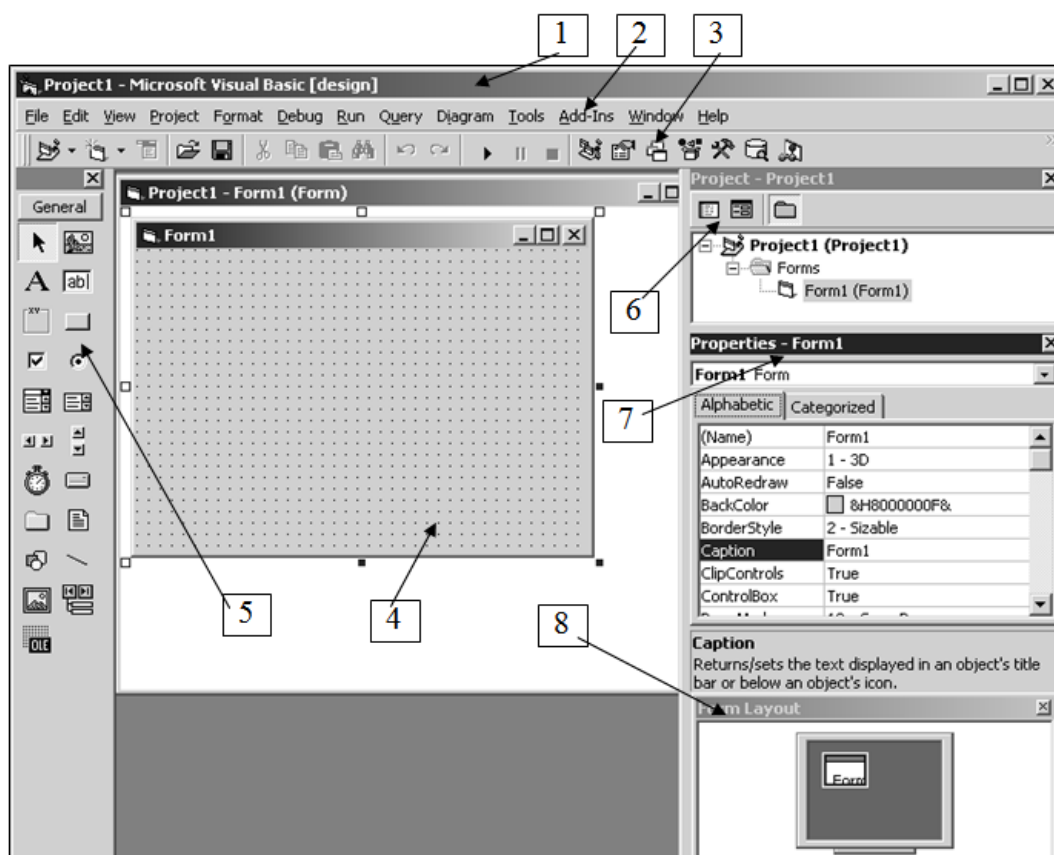
Вкладка **Existing** (Существующие) позволяет просмотреть все файлы ранее созданных проектов и выбрать необходимый.

Вкладка **Recent** (Свежие) предоставляет список проектов, работа с которыми велась во время предыдущего сеанса.

Проектом называют комплект файлов, которые используются для разработки программного приложения.

В окне **New Project** на вкладке **New** (Новый) выбрать стандартное приложение **VB Standard EXE** и нажать кнопку **Ok**.

Окно среды проектирования VB:



1 – **Строка заголовка.** В ней находится название проекта и его экранной формы. В квадратных скобках указывается режим работы – проектирование или выполнение.

2 – **Строка меню.** Содержит список команд, которые используются во время работы с Visual Basic.

3 – **Панель инструментов.** В ней находятся команды, которые чаще всего используются.

4 – **Окно экранной формы** – основная рабочая область программиста, которая по умолчанию называется **Form1**. Размеры формы можно изменять с помощью мыши, как и любое окно. На форме размещают различные элементы управления, необходимые для создания приложения.

5 – Окно элементов управления **Toolbox**. Содержит набор элементов управления, необходимых для разработки окна прикладной программы.

Элементы управления – это инструменты для создания объектов экранной формы.

6 – Окно проводника проектов **Project Explorer**. Предназначено для отображения структуры разрабатываемого проекта. Структура проекта отображается в виде "дерева" компонентов проекта Visual Basic.

7 – Окно свойств **Properties** дает возможность изменять свойства элементов управления, расположенных на форме. В окне свойств перечислены свойства элементов управления (по алфавиту или по категориям). Набор свойств зависит от типа элементов управления. Список свойств состоит из двух столбцов: в левом перечислены названия свойств, а в правом – их значения.

8 – Окно схемы формы **Form Layout** предназначено для установки места расположения формы будущей программы на рабочем столе после запуска программы.

Создание экранной формы

На форму добавляют различные элементы управления, необходимые для создания приложения. Установить на форме любой элемент управления можно двумя способами:

1. Дважды щелкнуть мышью по кнопке соответствующего инструмента на панели элементов управления **ToolBox**. При этом в центре экранной формы появится объект управления данного класса. Далее можно с помощью мыши переместить его в нужное место формы и установить его размеры.

Один раз щелкнуть мышью по кнопке соответствующего инструмента на панели **ToolBox**, при этом кнопка станет "нажатой" (утопленной). После этого поместить указатель мыши на экранную форму в место, где будет находиться левый верхний угол размещаемого объекта. Далее нажать левую кнопку мыши и, не отпуская ее, переместить указатель мыши в место, где будет находиться правый нижний угол будущего объекта. Этот процесс будет сопровождаться растягиванием пунктирной рамки – границы создаваемого объекта. Размеры и положение прямоугольника, который появился, можно корректировать.

Ввод программного кода

Программа – это запись алгоритма в виде программного кода на одном из языков программирования.

Программный код разрабатываемого проекта записывается в окне редактора кода, которое скрыто за окном экранной формы. Программный код почти всегда привязывает к какому-нибудь событию, например, щелчку по кнопке. Для привязки программного кода к определенному событию следует дважды щелкнуть объект, для которого надо написать код. Откроется окно редактора кода, в котором появится так называемая заготовка. Текст программного кода необходимо вводить между первой и последней строками заготовки.

Запуск программы и отладка программного кода

Запустить программу на выполнение можно клавишей **F5** или кнопкой **Start** (Начать) на панели инструментов.

После запуска программа преобразуется в машинные коды – язык, понятный компьютеру. Это преобразование называется **трансляцией**. Существует два вида трансляции.

Интерпретация – перевод программы в машинные коды построчно, по од-

ному оператору и одновременное их выполнение.

Компиляция – выполнение программы осуществляется после того, как процесс ее перевода полностью завершен.

Процесс поиска и устранения ошибок называется **отладкой программы**. Языки программирования имеют строгие правила записи команд. Эти правила записи образуют синтаксис языка программирования. Нарушение правил записи команд приводит к синтаксическим ошибкам, а нарушение алгоритма выполняемой задачи – к логическим ошибкам. Синтаксические ошибки обнаруживает транслятор во время перевода программы на машинный язык. Поиск логических ошибок занимается программист.

Сохранение приложения и создание exe-файла

Программное приложение, созданное в VB, сохраняется в двух файлах:

1. Файл формы с расширением `frm`, в котором заключено описание формы и ее компонентов.

2. Файл проекта с расширением `vbp`, содержащий информацию о строении проекта.

Для сохранения файла формы следует выполнить команду **File** → **Save Form As...** . В диалоговом окне **Save Form As** в текстовом поле **Папка** открыть папку Вашей группы, в текстовом поле **Имя файла** задать имя сохраняемой программы. Убедиться, что в текстовом поле **Тип файла** записано **Form Files (*.frm)** и нажать кнопку **Сохранить**.

Для сохранения файла проекта следует выполнить команду **File** → **Save Project As...** . В диалоговом окне **Save Project As** в текстовом поле **Имя файла** ввести сохраняемой программы (желательно то же, что и для формы, чтобы не путаться). Убедиться, что в текстовом поле **Тип файла** записано **Project Files (*.vbp)** и нажать кнопку **Сохранить**.

Для создания exe-файла следует выполнить команду **File** → **Make Project.exe**. в появившемся окне проверить имя и щелкнуть **Ok**.

Созданный exe-файл позволяет двойным щелчком на нем запускать созданное приложение без загрузки Visual Basic.

Лекция 10

ОСНОВНЫЕ КОМПОНЕНТЫ ЯЗЫКА VISUAL BASIC. ЛИНЕЙНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ

Алфавит языка VB

Visual Basic – это язык программирования. Как любой другой язык, имеет свой алфавит. Алфавит VB включает:

26 латинских букв: A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z;

10 арабских цифр: 1, 2, 3, 4, 5, 6, 7, 8, 9, 0;

26 символов: ! # \$ % ^ & * () + - / \ < > = ? @ _ : ; ' . , " пробел.

В Visual Basic приблизительно 200 встроенных операторов и функций. Каждый из них имеет четкую структуру (синтаксис), т.е. правила грамматики, пунктуации и орфографии. Для описания операторов применяются разные ключевые слова.

Нужно придерживаться определенных правил записи программ. В каждой строке кода есть оператор, который может иметь дополнительные параметры. Операторы могут располагаться последовательно в одной строке, тогда они разделяются двоеточием.

Логическую строку можно разделять на несколько физических. Разделителем строк служит пропуск, который идет за символом подчеркивания (_). Это дает возможность сформировать длинные строки так, чтобы они полностью помещались на странице экрана.

Строка программы в Visual Basic может содержать максимум 1023 символа и не более чем 10 разделителей.

В программе могут быть использованы комментарии. Они предназначены для объяснения отдельных фрагментов программы и игнорируются во время выполнения программы. Для выделения начала комментария можно использовать или апостроф ('), или оператор Rem, их действие одинаковое.

Rem – это оператор и потому он должен находиться в отдельной строке. Апостроф можно ставить в любом месте строки, при этом текст комментария располагается по правую сторону от этого символа.

Основные понятия о данных

Информация, предназначенная для хранения и обработки на компьютере, называется **данными**. Если данные изменяют свое значение во время выполнения программы, они называются **переменными**. Данные, которые имеют постоянное значение, называются **константами**. VB предусматривает 11 способов хранения и представления данных в компьютере. Эти способы называются типами данных.

Тип данных	Символ	Описание	Диапазон значений	Память
Byte		Двоичные данные	От 0 до 255	1 байт
Boolean		Логический	True или False	2 байта
Integer	%	Целые числа	От -32768 до 32767	2 байта
Long	&	Целые числа (длинные)	От -2147483648 до +2147483647	4 байта
Single	!	Действительные числа (одинарной точности)	От -3.4E+38 до +3.4E+38	4 байта
Double	#	Действительные числа (двойной точности)	От -1.79*D308 до +1.79*D308	8 байтов

Тип данных	Символ	Описание	Диапазон значений	Память
String	\$	Символьный	От 0 до 65,4 тыс. символов	10 байт + длина строки
Currency	@	Число с фиксированной десятичной точкой	От -922'337'203'685'477,5807 до 922337203685477,5807	8 байтов
Date		Дата	От January 1, 100 до December 31, 9999	8 байтов
Object		Объект	Содержит ссылку на объект	4 байта
Variant		Произвольный тип	Любой из перечисленных	От значения

Все данные в программе имеют **имя** (идентификатор). Имена используются для обозначения объектов в программе.

Правила образования имен:

- первым символом имени должна быть буква;
- имя может включать только буквы, цифры и символ подчеркивания (_);
- в имени может быть не больше 40 символов;
- ключевые слова нельзя использовать как имена.

Желательно давать объектам имена, которые отвечают контексту и несут смысловую нагрузку.

Правильные имена:

A2

Color

StartTime

Неправильные имена:

23B (так как начинается с цифры)

CM*PER*INCH (так как есть *)

File (так как ключевое слово)

Понятие переменной

Под переменной понимается имя ячейки памяти, куда могут быть «записаны» (помещены) какие-либо данные. При этом, для того чтобы выполнить какие-либо операции над данными необходимо записать выполнение этой операции над именем этой ячейки. По умолчанию VB не требует объявления переменных, а автоматически создает переменную типа Variant. Для более эффективного использования ресурсов компьютера, переменную следует объявлять. Объявить переменную – это значит назначить имя переменной, задать ее тип и дать указание VB зарезервировать память для заданного типа переменной.

С понятием имени переменной тесно связано понятие присваивания значений. В ячейку памяти в разные моменты времени могут быть помещены различные данные в то время как имя ячейки не будет изменяться.

Таким образом, присваивание значения переменной – это процесс помеще-

ния данных в ячейку с указанным именем. Присваивание значений записывается следующим образом:

A:=5.7

b := Zk

A:=20.8

Это значит, что переменной A присвоено значение 5.7, затем 20.8, а переменной b – "Zk", или, что то же самое, в ячейку A мы поместили 5.7, а затем 20.8, в ячейку b – текстовые данные "Zk".

При записи действий над переменными правомерна следующая запись:

A:=5

B:=25,5

A:=A+B

Такая запись означает: новое значение переменной A равно старому значению переменной A плюс значение переменной B.

При этом необходимо помнить, что правую и левую часть в таком выражении менять местами нельзя. В левой части может быть только имя переменной и не может быть число или арифметическое выражение.

В правой части может быть имя переменной, число или арифметическое выражение, но значения всех переменных, входящих в это выражение на момент выполнения операции должны быть определены, т. е. им должны быть присвоены значения.

Запись выражений по правилам VB

Выражения используются для выполнения операций над данными. В зависимости от данных и используемых операций выражения можно поделить на арифметические, логические и символьные.

В зависимости от типа выражения используются соответствующие операнды и знаки операций. Можно использовать следующие **знаки операций**:

+ – сложение (например, 2.36+12.5);

- – вычитание (например, 231-49);

* – умножение (например, 3*2);

^ – возведение в степень (например, 3^2=9);

/ – деление действительных чисел (с плавающей точкой: 3/2=1.5);

\ – деление целых чисел (например, 3\2=1);

Mod – вычисление остатка от деления (7Mod4=3).

Приоритет выполнения операций (в порядке уменьшения приоритета):

- 1) вычисление встроенных функций;
- 2) возведение в степень;
- 3) умножение и деление действительных чисел;
- 4) деление целых чисел;
- 5) вычисление остатка от деления;
- 6) сложение и вычитание.

Вычисления в выражениях осуществляются слева направо. Приоритет мож-

но изменить с помощью круглых скобок.

Примеры:

$14/5*2=5.6$ – операции одного приоритета выполняются слева направо;

$14\5*2=1$ – умножение имеет высший приоритет, а при делении целых чисел дробная часть числа отбрасывается;

$27^{1/3}=9$ – возведение в степень имеет наивысший приоритет;

$27^{(1/3)}=3$ – скобки изменяют последовательность операций.

В VB есть набор встроенных (стандартных) функций, который облегчает написание программ. Существуют математические функции, для обработки строк, для работы со временем и датами, для финансовых расчетов.

Основные математические функции:

Abs(x) - модуль (абсолютная величина) числа x;

Atn(x) - арктангенс угла x;

Cos(x) - косинус угла x;

Fix(x) - целое число, которое равняется числу x без дробной части;

Log(x) - возвращает значение натурального логарифма;

Rnd(x) - генерирует случайное число между 0 и 1;

Sqr(x) - квадратный корень от x;

Tan(x) - тангенс угла x;

Exp(x) - вычисляет e^x .

Строковые функции:

Len(x) - определяет длину строки;

Val(x) - превращает строку в число;

Str(x) - превращает число в строку;

Str() - создает строку символов.

Функции для работы со временем и датами:

Date - определяет и выдает текущую дату;

Time - определяет и выдает текущее время;

При записи математических выражений по правилам Visual Basic необходимо учитывать такие функциональные конструкции:

Математическая запись

$\arcsin x$

$\arccos x$

$\text{arcctg } x$

$\text{ctg } x$

$\log_b a$

$\lg a$

$\sqrt[n]{x}$

Запись по правилам Visual Basic

Atn(x/Sqr(1-x^2))

1.57-Atn(x/Sqr(1-x^2))

Atn(x)+1.57

1/Tan(x)

Log(a)/Log(b)

Log(a)/Log(10)

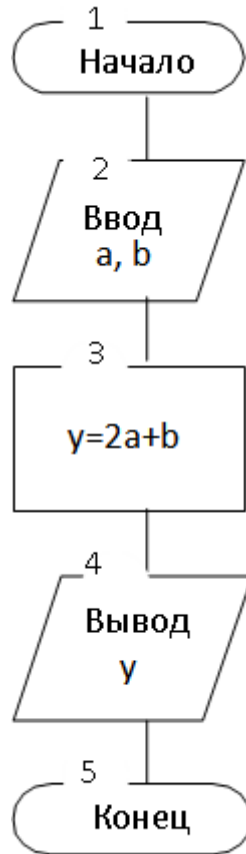
$x^{(1/n)}$

Линейные вычислительные процессы

Линейным называют вычислительный процесс, в котором операторы выполняются последовательно в порядке их записи по строкам.

Например, вычислить значение y по формуле: $y=2a+b$

Построим схему алгоритма линейного вычислительного процесса.



Выполним этапы программирования на языке VB.

1. Разработка экранной формы.

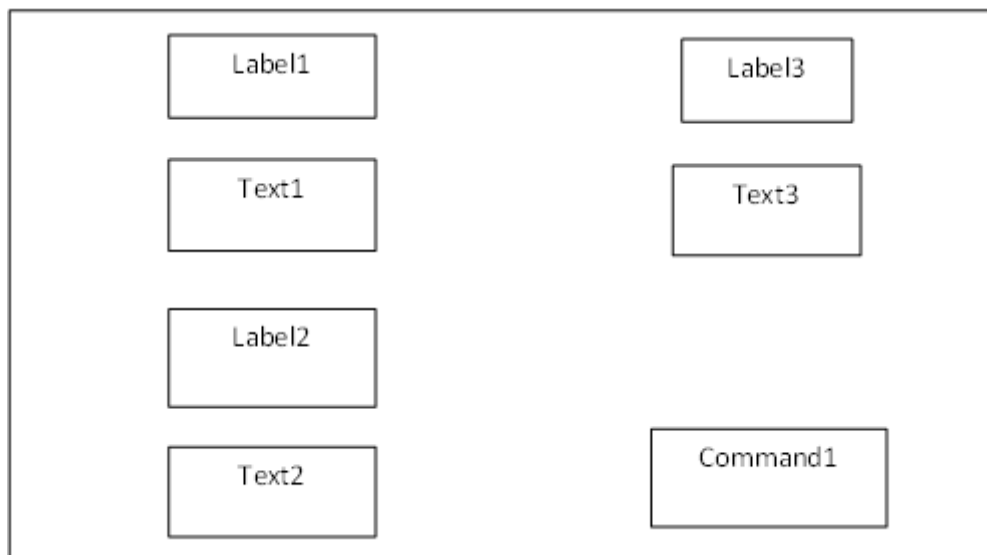
Для ввода-вывода данных в VB можно использовать инструмент управления TextBox, который позволяет вводить (выводить) текстовые строки произвольного формата во время выполнения приложения. По умолчанию в созданном текстовом поле элемента TextBox отображается значение свойства Text, которое совпадает с именем элемента (Text1, Text2...). Для очистки текстового поля следует в окне Свойств удалить текст, расположенный справа от свойства Text.

Для каждого элемента TextBox имеет смысл создать надпись, поясняющую назначение этого элемента. Для поясняющей надписи удобно использовать элемент Label, который применяется для отображения текста на форме. Чтобы получить на элементе Label необходимую надпись, следует в окне Свойств справа от свойства Caption удалить текст, соответствующий имени элемента, и ввести необходимый.

Для аккуратного размещения надписей текст в элементах Label можно расположить по центру элемента. Для этого надо в окне Свойств щелкнуть по стрелке справа от свойства Alignment (Выравнивание) и выбрать значение 2 – Center.

Для управления работой приложения на форму помещают командные кнопки `CommandButton`. Командные кнопки используются для выполнения определенных действий во время работы приложения, описанных с помощью программного кода.

Надпись на кнопке можно изменить. Для этого следует в окне Свойств удалить текст справа от свойства `Caption`, а затем набрать подходящую надпись, например, "Вычислить".



2. Разработка программного кода.

Условимся, что вычисление значения y будет происходить при щелчке на командной кнопке. Другими словами, будет программироваться событие `Click` для `Command1`.

Для создания программного кода окно редактора кода проще всего открыть двойным щелчком на командной кнопке. В окне редактора кода изначально уже введены 2 строчки:

```
Private Sub Command1_Click()  
End Sub
```

Между этими строчками следует вводить собственный программный код:

```
a = Text1.Text  
b = Text2.Text  
y = 2 * a + b  
Text3.Text = y
```

3. Запуск программы.

На экране появляется окно приложения, внешний вид которого соответствует разработанной форме. Далее следует ввести значения a и b в соответствующие текстовые поля, убедившись, что необходимое поле имеет фокус ввода, и щелкнуть на кнопке "Вычислить". Если все было выполнено без ошибок, в текстовом поле для вывода y появится результат вычислений.

Значения а и б можно задавать в окне свойств в поле свойства Text еще на этапе проектирования формы.

4. Сохранение.

Лекция 11

РАЗВЕТВЛЯЮЩИЕСЯ ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ

Разветвляющимися называются вычислительные процессы, в которых используются команды ветвления, а определение искомой величины производится по одному из нескольких направлений.

Ветвление имеет место, когда в программе выбирается один из двух альтернативных путей ее выполнения. В любых ветвящихся процессах порядок выполнения операторов изменяется с помощью операторов передачи управления, которые бывают следующих видов:

Оператор условного перехода линейной формы сокращенного формата

Используется, когда при выполнении какого-либо условия необходимо выполнить один оператор. Формат оператора:

If *условие* **Then** *оператор*

Если результат проверки условия является истиной, то выполняется оператор после слова Then. Если результат проверки условия ложь, то выполняется следующий по порядку оператор в программе.

Пример 1:

$$y = \begin{cases} 2 - x, & \text{если } x \geq 0 \\ 3x, & \text{если } x < 0 \end{cases}$$

...

If x >= 0 Then y = 2 - x

If x < 0 Then y = 3 * x

...

Оператор условного перехода линейной формы полного формата

Структура с двумя переходами может быть реализована оператором условного перехода полного формата:

If *условие* **Then** *оператор1* **Else** *оператор2*

Если результат проверки условия будет истиной, то выполняется оператор за словом Then. Если результат проверки условия будет ложным, то выполняется оператор за словом Else.

Пример 2:

```
...  
If x >= 0 Then y = 2 - x Else y = 3 * x  
..
```

Оператор условного перехода блочной формы

Если при выполнении условия требуется выполнить не один, а несколько операторов, то используются блочные операторы условного перехода, которые также могут быть сокращенного и полного формата.

Сокращенный формат:

```
If условие Then  
    Блок операторов  
End If
```

В случае истинности проверяемого условия будут выполнены операторы, расположенные после слова **Then**. Если условие является ложным, то выполняется следующий после данной конструкции оператор. Если в блоке операторов находится только один оператор, то данная структура все равно должна заканчиваться словосочетанием **End If**.

Пример 3:

```
...  
If x >= 0 Then  
    y = 2 - x  
    Text2.Text = y  
End If  
If x < 0 Then  
    y = 3 * x  
    Text2.Text = y  
End If  
...
```

Блочная конструкция полного формата:

```
If условие Then  
    Блок операторов 1  
[ElseIf условие Then  
    Блок операторов 2  
Else  
    Блок операторов 3]  
End If
```

Пример 4:

```
...
If x >= 0 Then
  y = 2 - x
ElseIf x < 0 Then
  y = 3 * x
End If
Text2.Text = y
...
```

В операторе условного перехода условие может быть как простым, так и составным с операторами Or или And. В одной программе могут использоваться разные формы операторов.

Пример 5:

$$y = \begin{cases} \operatorname{tg}^2 x, & \text{если } -1 \leq x < 0,2 \\ x^3, & \text{если } 0,2 \leq x < 1 \\ \sqrt{x}, & \text{если } 1 \leq x \leq 100 \end{cases}$$

```
...
If X < -1 Or X > 100 Then
  Text2.Text = "Нет решения"
Else
  If X >= -1 And X <= 0.2 Then y = Tan(X) ^ 2
  If X >= 0.2 And X < 1 Then y = X ^ 3
  If X >= 1 And X <= 100 Then y = Sqr(X)
  Text2.Text = y
End If
```

Оператор выбора

Если необходимо выполнить несколько операторов в случаях истинности каких-либо условий, пользуются оператором выбора. Формат оператора:

```
Select Case переменная
  Case значение1
    Блок операторов 1
  Case значение2
    Блок операторов 2
  ...
  Case Else
    Блок операторов N
End Select
```

Если переменная содержит значение 1, то выполняется блок операторов 1. Если содержимое переменной равно значению 2, то выполняется блок операторов 2 и т. д. Когда содержимое переменной не равно ни одному из приведенных значений, то выполняется блок операторов N, находящийся после словосочетания Case Else, которое является необязательным. В качестве значения может использоваться любое числовое или символьное выражение. Значение переменной может иметь разные формы записи. В случае, когда при нескольких значениях переменной необходимо выполнить один и тот же блок операторов, эти значения указываются через запятую. Если значение находится в ограниченном интервале, в операторе Case используется служебное слово To, с помощью которого указываются начальное и конечное значения диапазона.

Пример 6:

$$y = \begin{cases} \operatorname{tg}^2 x, & \text{если } -1 \leq x < 0,2 \\ x^3, & \text{если } 0,2 \leq x < 1 \\ \sqrt{x}, & \text{если } 1 \leq x \leq 100 \end{cases}$$

...

```
Select Case x
  Case -1 To 0.2
    y = Tan(x) ^ 2
    Text2.Text = y
  Case 0.2 To 1
    y = x ^ 3
    Text2.Text = y
  Case 1 To 100
    y = Sqr(x)
    Text2.Text = y
  Case Else
    Text2.Text = "Нет решения"
End Select
```

В операторе Case можно использовать операцию сравнения, которая определяется операторами отношения: <, >, ≤, ≥, <>, =. В качестве оператора сравнения используется служебное слово Is.

Пример 7:

```
x = Val(TextBox("Введите значение X"))
List1.AddItem (x)
Select Case x
  Case Is < -1
    y = "Нет решения"
  Case Is < 0.2
    y = Tan(x) ^ 2
```

```

Case Is < 1
  y = x ^ 3
Case Is <= 100
  y = Sqr(x)
Case Is > 100
  y = "Нет решения"
End Select
List2.AddItem (y)

```

Лекция 12 ЦИКЛИЧЕСКИЕ ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ

Циклическим называется вычислительный процесс, в котором имеются многократно повторяющиеся участки, называемые циклами. При каждом новом повторении цикла изменяются один или несколько входных параметров и результаты вычислений. Для реализации циклических процессов используются операторы циклов. В языке VB существуют 2 основных вида циклов:

- 1 – арифметические циклы, которые реализуются при помощи оператора цикла со счетчиком (параметром);
- 2 – итерационные, реализуемые с помощью оператора цикла с условием.

Арифметический цикл

Арифметический цикл (с параметром) используется, когда количество повторов заданного блока операторов известно заранее. Формат оператора:

```

For параметр цикла = нач.знач. To кон.знач. [Step шаг]
  Блок операторов (тело цикла)
Next [параметр цикла]

```

Если параметр Step=1, то этот параметр в операторе цикла не указывается.

Пример 1: Вычислить n!

```

...
Fact = 1
For i = 1 To n
  Fact = Fact * i
Next i
...

```

Пример 2: $y = \sum_{x=1}^{100} \frac{x^2}{3}$; $1 \leq x \leq 100$. Найти все $y < 50$.

```

...
For x = 1 To 100
  y = y + x ^ 2 / 3
If y > 50 Then Exit For

```

```
List1.AddItem (x)
List2.AddItem (y)
Next x
...
```

Когда оператор цикла выполняется первый раз, параметру цикла присваивается начальное значение, после чего текущее значение параметра цикла сравнивается с его конечным значением. Если текущее значение параметра цикла окажется больше, чем его конечное значение, то происходит завершение цикла, и блок операторов тела цикла ни разу не выполняется. Если текущее значение параметра цикла оказывается не больше конечного значения, блок операторов тела цикла выполняется, после чего происходит переход на начало цикла. Далее к текущему значению параметра цикла прибавляется шаг. После этого, текущее значение параметра цикла снова сравнивается с конечным значением и т. д. до тех пор, когда значение параметра цикла станет больше конечного значения.

Шаг приращения параметра цикла может быть отрицательным, в этом случае конечное значение параметра цикла должно быть меньше начального. Иногда необходимо завершить работу цикла при выполнении какого-либо дополнительного условия. В этом случае внутри цикла используют служебное словосочетание Exit For.

Итерационный цикл

Итерационный цикл (с условием) применяется в тех случаях, когда число повторений заранее неизвестно. Операторы цикла с условием подразделяются на циклы с предусловием и циклы с постусловием. В циклах с предусловием в зависимости от результата проверки условия операторы цикла могут не выполниться ни одного раза. В циклах с постусловием операторы цикла в любом случае будут выполнены хотя бы один раз. Существует 4 формата операторов цикла с условием. Первые два формата представляют собой циклы с предусловием, третий и четвертый форматы – циклы с постусловием.

1) **Do Until** *условие*
Блок операторов
Loop

Если результат проверки условия – ложь, то выполняются операторы тела цикла. Если результат проверки условия – истина, осуществляется переход на оператор, следующий после слова Loop.

Пример 2: Алгоритм Эвклида. Найти наибольший общий делитель чисел X1 и X2 (25 и 15; 20 и 4; 22 и 6).

```
...
Do Until X1 = X2
  If X1 > X2 Then
    X1 = X1 - X2
```

```
Else
  X2 = X2 - X1
End If
Loop
...
```

2) **Do While** *условие*
Блок операторов
Loop

Если условие истинно, то выполняются операторы тела цикла. Если условие ложно, то выполняется оператор, следующий за словом Loop.

Пример 3: Алгоритм Эвклида.

```
...
Do While X1 <> X2
  If X1 > X2 Then
    X1 = X1 - X2
  Else
    X2 = X2 - X1
  End If
Loop
...
```

3) **Do**
Блок операторов
Loop Until *условие*

В случае истинности условия происходит окончание цикла. Если условие ложно, выполняются операторы цикла.

4) **Do**
Блок операторов
Loop While *условие*

В случае истинности условия выполняются операторы цикла. Если условие ложно, происходит окончание цикла.

Циклы бывают простыми и сложными. Простым называют цикл, не содержащий ветвлений или других циклов; сложным – цикл, в котором имеются вложенные в него циклы. При этом первый цикл, охватывающий другие циклы, называют внешним, а все остальные – внутренними. Тело вложенного цикла должно целиком содержаться в теле внешнего цикла. Можно передавать управление из внутренних циклов в область действия внешних, но не наоборот.

Типовые задачи с циклическими процессами

1. Табулирование функций. Вычислить значение функции Y по аргументу X , изменяющемуся от X_H до X_K с постоянным приращением ΔX .

Пример. Вычислить значение Y по формуле

$$Y = \frac{\sin ax}{x}.$$

...

```
For X = Xn To Xk Step dX
```

```
Y = Sin (a * X) / X
```

```
List1.AddItem (X)
```

```
List2.AddItem (Y)
```

```
Next X
```

...

Элементы управления `ListBox` используются для отображения списка вводимых значений X и полученных результатов Y . Значения X и Y добавляются в окно элементов `ListBox` с помощью метода `AddItem`.

2. Вычисление суммы k чисел. Вычислить

$$S = \sum_{n=1}^K x_n,$$

где $n = 1, 2, \dots, k$; $X_n = (X_1, X_2, \dots, X_k)$. Вычисления выполняются с помощью рекуррентной формулы $S_n = S_{n-1} + X_n$, при этом задается $S_0 = 0$.

...

```
S = 0
```

```
For N = 1 To K
```

```
X = InputBox ("Ведите X")
```

```
List1.AddItem (X)
```

```
S = S + X
```

```
Next N
```

```
Text2.Text = S
```

...

3. Вычисление произведения k чисел. Вычислить

$$P = \prod_{n=1}^K x_n,$$

где $n = 1, 2, \dots, k$, $X_n = \{X_1, X_2, \dots, X_k\}$. Вычисления выполняются с помощью рекуррентной формулы $P_n = P_{n-1} * X_n$, при этом задается $P_0 = 1$.

...

```
P = 1
```

```
For N = 1 To K
```

```
X = InputBox ("Ведите X")
```

```
List1.AddItem (X)
```

```
P = P * X
```

Next N
Text2.Text = P
...

Лекция 13

ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ С МАССИВАМИ ДАННЫХ

Массивом называется упорядоченный набор однотипных элементов, объединенных под одним именем.

Массив характеризуется именем, типом, размером и размерностью.

Тип массива – это тип элементов, из которых состоит массив.

Размер массива определяет количество элементов в массиве.

Каждый элемент массива имеет свое значение и номер, называемый индексом. Индекс однозначно определяет местоположение элемента в массиве.

Размерность – это количество индексов, необходимых для определения местоположения элементов в массиве.

Массивы бывают одномерными и многомерными. Если размерность массива равна 1, массив является одномерным. Одномерные массивы называют векторами. Если размерность массива равна 2, массив является двумерным. Двумерные массивы называют матрицами.

Объявление массивов

При составлении программ с массивами, массивы следует объявлять с помощью оператора Dim, который может располагаться в секции General Declaration (общий раздел объявлений) или непосредственно в тексте программы.

Формат оператора:

Dim *имя массива* (**I**) [**As** *тип элементов массива*]

где I – размер массива.

Для каждого массива определяется верхняя и нижняя границы, в которых может изменяться значение индекса. По умолчанию нижней границей любого массива является 0. Например, оператор Dim A(9) объявляет массив из 10-ти элементов. Для индексации массива с номера 1 следует перед объявлением массива поместить оператор Option Base 1. Для индексации массива с любого номера следует в операторе Dim объявить обе границы массива. В этом случае формат оператора Dim примет следующий вид:

Dim *имя массива* (**In To Ik**) [**As** *тип элементов массива*]

где In – индекс первого элемента;

Ik – индекс последнего элемента.

Например, оператор Dim A (1 To 10) также объявляет массив A из 10-ти элементов, а оператор Dim B (1 To 5) As Byte объявляет массив B из 5-ти элементов типа Byte.

В VB массивы могут быть статические и динамические. В статических массивах значения верхней и нижней границ не могут быть изменены при выполнении программы, т. е. количество элементов в массиве известно и задается заранее (как в примерах). Динамические массивы используются в том случае, когда при составлении программы количество элементов массива неизвестно и будет определяться в процессе выполнения программы.

Ввод-вывод элементов массива

Ввод-вывод элементов массива производится в цикле, параметром которого является индекс элементов. Например:

```
Dim X(1 To 10) As Single
For I=1 To 10
X(I)= InputBox("Введите"+Str(I)+"-й элемент массива X")
Next I
Print "Массив X:"
For I=1 To 10
Print X(I);
Next I
```

Оператор печати Print можно использовать для вывода сообщений или результатов вычисления непосредственно на форму. Символ ; в операторе Print означает, что следующий элемент будет выводиться в той же строке через 1 пробел. Если ; заменить на , элементы будут выводиться по зонам (каждая зона – 16 позиций: 14 для символов, 1 для знака числа, 1 пробел). Если оператор Print не содержит ни запятой, ни точки с запятой, элементы выводятся с новой строки, т. е. в столбец. Пустой оператор Print производит перевод курсора на новую строку.

Ввод элементов массива может производиться непосредственно в программе с помощью оператора присваивания. Например:

```
X(1)=0.4
X(2)=15
X(3)=-3.8
и т. д.
```

Ввод-вывод элементов матрицы

```
Dim X(1 To 3, 1 To 4) As Single
For I=1 To 3
For J=1 To 4
X(I,J)= InputBox("Введите"+Str(J)+"-й элемент"+Str(I)+"-й строки матрицы X")
```

```

Next J, I
Print "Матрица X:"
For I=1 To 3
For J=1 To 4
Print X(I, J),
Next J
Print
Next I

```

Подсчет суммы, произведения, количества элементов массива

```

Dim X(1 To 10) As Single
S=0 : P=1 : K=0
For I=1 To 10
X(I)=InputBox("Введите"+Str(I)+"-й элемент массива X")
S=S+X(I) : P=P*X(I) : K=K+1
Next I
Print "Массив X:"
For I=1 To 10
Print X(I);
Next I
Print
Print "S=";S
Print "P=";P
Print "K=";K

```

**Подсчет суммы, произведения, количества элементов массива, больших
(меньших) заданного числа.**

```

Dim X(1 To 10) As Single
S=0 : P=1 : K=0
A=5
For I=1 To 10
X(I)=InputBox("Введите"+Str(I)+"-й элемент массива X")
If X(I)>A Then S=S+X(I) : P=P*X(I) : K=K+1
Next I
Print "Массив X:"
For I=1 To 10
Print X(I);
Next I
Print
Print "S=";S
Print "P=";P
Print "K=";K

```

Подсчет суммы, произведения, количества элементов массива на заданном интервале.

```
Dim X(1 To 10) As Single
S=0 : P=1 : K=0
A=3: B=7
For I=1 To 10
X(I)=InputBox("Введите"+Str(I)+"-й элемент массива X")
If X(I)>A And X(I)<B Then S=S+X(I) : P=P*X(I) : K=K+1
Next I
Print "Массив X:"
For I=1 To 10
Print Str(X(I));
Next I
Print
Print "S=";S
Print "P=";P
Print "K=";K
```

Нахождение минимального и максимального элементов массива, а также их номеров

```
Dim X(1 To 10) As Single
For I=1 To 10
X(I)=InputBox("Введите"+Str(I)+"-й элемент массива X")
Next I
Max=X(1) : IMax=1 : Min=X(1) : IMin=1
For I=1 To 10
If X(I)>Max Then Max=X(I) : IMax=I
If X(I)<Min Then Min=X(I) : IMin=I
Next I
Print "Массив X:"
For I=1 To 10
Print X(I);
Next I
Print
Print "Max=";Max
Print "IMax=";IMax
Print "Min=";Min
Print "IMin=";IMin
```

Упорядочивание элементов массива по возрастанию (убыванию) значений элементов

```
Dim X(1 To 10) As Single
For I=1 To 10
X(I)=InputBox("Введите"+Str(I)+"-й элемент массива X")
```

```
Next I
Print "Массив X до упорядочивания:"
For I=1 To 10
Print X(I);
Next I
Print
For N=1 To 10
For I=1 To 9
If X(I)>X(I+1) Then C=X(I) : X(I)=X(I+1) : X(I+1)=C
Next I, N
Print "Массив X после упорядочивания:"
For I=1 To 10
Print X(I);
Next I
```

ЛИТЕРАТУРА

1. Симонович, С.В. Информатика: базовый курс: учеб. пособие для студентов ВУЗов / С.В. Симонович – СПб.: Питер, 2007. (31)
2. Рогоза, М.Є. Информатика і комп'ютерна техніка: навч. посіб / М.Є. Рогоза, В.І. Клименко, Л.Ф. Крещенко та ін. – К.: Академія, 2006. (23)
3. Браун, С. Visual Basic 6: учеб. курс / С. Браун – СПб.: Питер, 2007. (24)
4. Методичні вказівки для виконання лабораторних і самостійних робіт по розділу «Табличний процесор Microsoft Excel» з дисциплін «Інформатика», «Комп'ютерна техніка і програмування» для студентів денної форми навчання всіх напрямів підготовки / ХНАДУ; уклад.: М. В. Костікова, І. В. Скрипіна, А. І. Кудін, В. О. Шевченко. – Х., 2015. – 78 с.
5. Методичні вказівки до лабораторних робіт з дисципліни "Інформатика", "Комп'ютерна техніка та програмування" для студентів напрямками підготовки "Автомобільний транспорт", "Транспортні технології", "Будівництво", "Екологія, охорона навколишнього середовища та збалансоване природокористування", розділ "Інтегроване середовище розробки Visual Basic" – Х., 2010. – 84 с.
6. Омельченко, Л.Н. Microsoft Windows 7. Самое необходимое / Л.Н. Омельченко, А.Ф. Тихонов – СПб.: БХВ-Петербург, 2010. – 368 с.
7. Гельман, В.Я. Решение математических задач средствами Excel: практикум / В.Я. Гельман – СПб.: Питер, 2003. (1)
8. Глушаков, С.В. Программирование в среде Windows: учеб. курс / С.В. Глушаков, И.В. Мельников, А.С. Сурядный – Х.: Фолио, 2001. (1)
9. <http://dl.khadi.kharkiv.edu/course/view.php?id=12> – Курс "Інформатика і системологія" для студентів потоку 1ДЕК: матеріали для самостійної роботи, лекції по курсу, методичні вказівки до підготовки та виконання лабораторних робіт, питання для підготовки до модульних контролів.
10. <http://files.khadi.kharkov.ua/mekhatroniki-transportnikh-zasobiv/> – Файловий архів ХНАДУ: методичні вказівки до підготовки та виконання лабораторних робіт.