

**МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО СПЕЦИАЛЬНОГО
ОБРАЗОВАНИЯ**

ТАШКЕНТСКИЙ ГОСУДАРСТВЕННЫЙ

ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ

ИМЕНИ НИЗАМИ

И.У. НАЗАРОВ, А.А. ХАСАНОВ, О.Э. МАМАРАЖАБОВ

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

**УЧЕБНИК ДЛЯ СТУДЕНТОВ ВЫСШИХ ПЕДАГОГИЧЕСКИХ
УЧЕБНЫХ ЗАВЕДЕНИЙ**

Ташкент - 2020

Назаров И.У., Хасанов А.А., Мамаражабов О.Э. «Информационные технологии в образовании». Учебник для студентов высших педагогических учебных заведений, обучающихся по программе бакалавриата.

ANNOTATSIYA

Ushbu Ta'limda axborot texnologiyalari nomli darslik oliy ta'limning davlat ta'lim standarti talablariga muvofiq ravishda tayyorlangan. Darslik pedagogika oliy o'quv yurtlarining o'quv jarayonida zamonaviy axborot texnologiyalarini qo'llashning dolzarb muammolariga bag'ishlangan.

Darslikning nazariy qismida ta'limda axborot texnologiyalarini rivojlantirish masalalari, o'quv jarayonida axborot-kommunikatsiya texnologiyalaridan foydalanishning didaktik jihatlari bayon etilgan.

Amaliy va laboratoriya qismlarining maqsadi esa ta'lim sohasida axborot texnologiyalaridan amaliy foydalanish qobiliyatlarini shakllantirishdan iborat.

Darslik bakalavriat dasturi bo'yicha tahsil olayotgan pedagogika oliy ta'lim muassasalari talabalar uchun mo'ljallangan. Har bir bo'limda o'z-o'zini nazorat qilish uchun savollar va vazifalar mavjud.

АННОТАЦИЯ

Учебник подготовлен в соответствии с требованиями государственного образовательного стандарта высшего образования для обязательного изучения дисциплины «Информационные технологии в образовании». Он посвящен актуальным проблемам применения современных информационных технологий в учебном процессе вузов. В теоретической части излагаются вопросы развития информационных технологий обучения, дидактические аспекты использования возможностей информационных и коммуникационных технологий в учебном процессе. Целью практической и лабораторной части является отработка умений практического использования информационных технологий в образовании.

Учебник предназначен для студентов высших педагогических учебных заведений, обучающихся по программе бакалавриата. В каждом разделе имеются вопросы и задачи для самоконтроля.

ANNOTATION

The textbook is prepared in accordance with the requirements of the state educational standard of higher education for compulsory study of the discipline "Information technologies in education". It is devoted to the actual problems of using modern information technologies in the educational process of higher education institutions. The theoretical part describes the development of information technologies in education, didactic aspects of using the capabilities of information and communication technologies in the educational process. The purpose of the practical and laboratory part is to develop the skills of practical use of information technologies in education.

The textbook is intended for students of higher pedagogical educational institutions who are studying under the bachelor's program. Each section contains questions and tasks for self-monitoring.

Рецензенты:

д.т.н., профессор кафедры «Методика преподавания информатики»

Р.Х. Аюпов,

доцент кафедры «Мультимедийные технологии» ТУИТ,

доктор философии по т.н. (*PhD*) Д.К. Шарипов.

ВВЕДЕНИЕ

Предлагаемый учебник написан в соответствии с Государственным образовательным стандартом по дисциплине «*Информационные технологии в образовании*». В современном образовании все большую популярность приобретает компетентностный подход, под которым понимается реализация образовательных программ, направленных на формирование способности личности самостоятельно в определенном контексте применять полученные знания и умения в своей профессиональной деятельности.

В настоящее время происходит смена педагогической парадигмы, когда основной целью системы образования является подготовка специалистов двойной компетенции: с одной стороны, жестко связанных с профессией, а с другой - способных не только понимать возможности информационно-коммуникационных технологий (*ИКТ*), но и использовать их, адаптируя к выполнению практических задач, решению специализированных проблем, возникающих в различных областях деятельности человека. Информационные технологии составляют основу формирования общепрофессиональных компетенций современного студента любого профиля. Их использование является необходимым условием повышения результативности образования, развития более эффективных подходов к обучению, совершенствованию методики преподавания.

Современная проблема информатизации системы образования состоит не только в создании технических условий для внедрения компьютерных технологий в учебный процесс, но и в том, чтобы педагоги были ментально подготовлены к постоянному повышению своего уровня компетентности в области информационно коммуникационных технологий. В связи с этим одним из важных направлений является компетентностно-ориентированное образование в области информационных технологий.

Использование информационных технологий оказывает заметное влияние на содержание, формы и методы обучения. Возможности современной вычислительной техники в значительной степени адекватны

организационно-педагогическим и методическим потребностям высшего образования:

✓ вычислительные возможности - быстрое и точное преобразование любых видов информации (*числовой, текстовой, графической, звуковой и др.*);

✓ прием и выдача информации в самой различной форме (*при наличии соответствующих устройств*);

✓ комбинаторные - запоминание, хранение, структурирование, систематизация больших объемов информации, быстрое нахождение необходимой информации;

✓ графические - представление результатов работы в наглядной графической, видео - и анимационной формах;

Широкое использование информационно-коммуникационных технологий в образовательном процессе вузов позволяет обеспечить:

- вариативность и личностную ориентацию образовательного процесса;

- практическую ориентацию образовательного процесса с введением интерактивных деятельностных компонентов;

- формирование способностей и компетентностей, необходимых для продолжения образования в соответствующем профессиональном направлении.

В настоящем учебнике систематизирован материал для формирования ИКТ-компетентности студентов педагогических вузов в обучении. Содержание учебника *«Информационные технологии в образовании»* соответствует государственному образовательному стандарту высшего профессионального образования (*квалификация “бакалавр”*). Согласно требованиям, предъявляемым в государственных стандартах к результатам освоения основных образовательных программ бакалавриата, студент должен обладать следующими компетенциями:

- способен к обобщению, анализу, восприятию информации, постановке цели и выбору путей ее достижения;

- готов использовать основные методы, способы и средства получения, хранения, переработки информации, готов работать с компьютером как средством управления информацией;

- способен работать с информацией в глобальных компьютерных сетях;

- способен понимать сущность и значение информации в развитии современного информационного общества, сознавать опасности и угрозы, возникающие в этом процессе, соблюдать основные требования информационной безопасности, в том числе защиты государственной тайны;

- способен к подготовке и редактированию текстов профессионального и социально значимого содержания;

- готов применять современные методики и технологии, в том числе и информационные, для обеспечения качества учебно-воспитательного процесса;

- способен использовать возможности образовательной среды, в том числе, информационной, для обеспечения качества учебно-воспитательного процесса.

Учебник состоит из теоретической и практической частей. В теоретической части кратко изложены вопросы, связанные с использованием информационно-коммуникационных технологий в образовании. Он состоит из пяти глав:

Глава 1. Современные информационные технологии и их применение.

Глава 2. Современные технологии программирования.

Глава 3. Использование прикладных программ.

Глава 4. Компьютерные сети.

Глава 5. Информационные системы и их использование

Практическая часть посвящена выработке *умений* и *навыков* по использованию ИКТ в практической деятельности студентов.

Авторы считают своим долгом выразить искреннюю признательность доктору технических наук, профессору кафедры «*Методика преподавания Информатики*» ТГПУ имени Низами Аюпову Р.Х. за ценные советы и замечания, сделанные при рецензировании рукописи, а также Л.М. Константиновой, учителю высшей категории, члену научно-методического совета при Министерстве народного образования Республики Узбекистан, за помощь в редактировании и подготовке рукописи к печати.

ГЛАВА 1. СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИХ ПРИМЕНЕНИЕ

МОДУЛЬ 1

1-§. ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ ПРЕДМЕТ, ЦЕЛЬ И ЗАДАЧИ

Ключевые слова: *информационные технологии, информация, графика, обучающиеся, системные и прикладные программы, учебный процесс, компьютеризация, электронное обучение, система образования, новые образовательные технологии*

Определение информационных технологий. Современный этап развития общества характеризуется интенсивным развитием и расширением сферы распространения информационных технологий. Определяющее значение информационных технологий в современном мире связывают с тем, что они оказывают свое влияние на все *другие технологии* и на развитие общества в целом. В этой связи уровень *социально-экономического развития* отдельных государств не случайно связывают с развитием *информационных технологий*.

Как и любые другие технологии, *информационные технологии* служат инструментом для достижения поставленных целей. Эффективность использования информационных технологий при решении различных задач существенным образом зависит от уровня профессиональной подготовки пользователей. Отсюда подлинный ключ к информационным технологиям – это *знания и умения*, позволяющие ответить на вопросы «*что?*», «*как?*», «*когда?*» и «*почему?*» необходимо сделать для получения нужного результата.

Каждая информационная технология имеет свою специфику. Вместе с тем, независимо от специфики каждый вид технологии обладает общим набором компонентов (*процессы, методы, средства, система, деятельность*

и др.). Это позволяет сформулировать определение родового понятия «информационная технология».

Информационная технология (ИТ) – упорядоченная, системно организованная совокупность процессов, методов и средств количественного и качественного преобразования информации.

Предмет информационной технологии - обеспечение оптимальных взаимосвязей между всеми её компонентами, а также обеспечение ее гармоничного взаимодействия с внешней средой. Основными компонентами информационной технологии, как и любой другой технологии, являются исходный продукт (*предмет труда*), процессы, методы, средства, конечный продукт, субъекты, объекты, условия реализации (*среда*).

Цель информационной технологии - получение заданных информационных продуктов и услуг с наименьшими затратами.

Задачи информационной технологии:

+ разработка и использование эффективных производственных и технологических информационных процессов;

+ разработка оптимальных методов и средств преобразования информации;

+ обеспечение технологичности информационного производства.

№	Наименование компонента	Состав компонента
1.	Исходный продукт (<i>предмет труда</i>)	Информация, данные, документы
2.	Процессы	Производственные и технологические информационные процессы
3.	Методы	Методы преобразования информации в различных <i>информационных процессах</i>
4.	Средства	Средства <i>информационного, лингвистического, программного, технического, технологического, математического, организационного, кадрового</i> и др. обеспечения
5.	Конечный продукт	Информационные продукты и услуги
6.	Субъекты	Информационные работники

7.	Объекты	Пользователи
8.	Условия реализации (среда)	Информационные системы и сети

(табл.1.1.1). Состав компонента ИТ.

Классификация информационных технологий

В настоящее время не существует единой классификации информационных технологий. Основными причинами этого являются стремительное увеличение количества видов информационных технологий и наличие *разнообразных подходов* к их классификации.

1. По масштабу сферы приложения

- 1.1. *Глобальные*
- 1.2. *Базовые*
- 1.3. *Конкретные*

2. По способу производства информационных продуктов и услуг

- 2.1. *Ручные*
- 2.2. *Механизированные*
- 2.3. *Автоматизированные (человеко-машинные)*

3. По степени новизны

- 3.1. *Традиционные*
- 3.2. *Новые*
- 3.3. *Новейшие*

4. По назначению

- 4.1. *Прикладные*
- 4.2. *Функциональные*
- 4.3. *Инструментальные (обеспечивающие)*
- 4.4. *Предметные*

5. По типу пользовательского интерфейса

5.1. **Wimp-ИТ** (*при его использовании на экране высвечивается окно, содержащее образы программ и меню действий. Для выбора одного из них используется указатель мыши*).

5.2. **Silk-ИТ** (при использовании этой информационной технологии на экране по речевой команде происходит перемещение от одних поисковых образов к другим по смысловым (семантическим) связям)

6. По способу организации работы вычислительной техники

6.1. Автономные

6.2. Сетевые

7. По форме представления информации

7.1. Традиционные

7.2. Гипертекстовые

7.3. Мультимедийные

8. По величине эффекта от использования ИТ в конкретной предметной области

8.1. Инжиниринговые

8.2. Реинжиниринговые

9. По степени распространения

9.1. Типовые

9.2. Авторские

10. По режиму взаимодействия пользователя с вычислительной техникой

10.1. Пакетные

10.2. Интерактивные

11. По типу используемых программных средств

11.1. HTML-технологии

11.2. FLASH-технологии

11.3. CASE- технологии

Этапы и тенденции развития информационной технологии

По признаку технического обеспечения можно выделить *пять этапов* в развитии *информационных технологий*.

1. «Ручная информационная технология» (до второй половины XIX в), инструментарий которой составляли *перо, чернильница, книга*. Коммуникации

осуществлялись *почтой*. Основная цель технологии - представление информации в нужной форме.

2. «*Механическая технология*» (с конца XIX в.), инструментарий которой составляли пишущая машинка, телефон, диктофон, оснащенная более совершенными средствами доставки почта. Основная цель технологии - представление информации в нужной форме более удобными средствами.

3. «*Электрическая технология*» (40 - 60-е гг. XX в.), инструментарий которой составляли большие ПК и соответствующее программное обеспечение, электрические пишущие машинки, ксероксы, портативные диктофоны. Акцент в информационной технологии начинает перемещаться с формы представления информации на формирование ее содержания.

4. «*Электронная технология*» (с начала 70-х гг.), основным инструментарием которой становятся большие ПК и создаваемые на их базе автоматизированные системы управления (АСУ) и информационно-поисковые системы (ИПС), оснащенные широким спектром базовых и специализированных программных комплексов. Центр тяжести технологии еще более смещается на формирование содержательной стороны информации для управленческой среды различных сфер общественной жизни, особенно на организацию аналитической работы.

5. «*Компьютерная (новая) технология*» (с середины 80-х гг.), основным инструментарием которой является персональный компьютер с широким спектром стандартных программных продуктов разного назначения. В связи с переходом на микропроцессорную базу существенным изменениям подвергаются и технические средства бытового, культурного и прочего назначений. Начинают широко использоваться в различных областях глобальные и локальные компьютерные сети.

6. В настоящее время - *ИКТ (информационно коммуникационные технологии)*.

Понятие информации, ее виды и свойства.

В любой науке основные понятия определить достаточно сложно. Так и понятие «информация» невозможно определить через другие, более «простые» понятия.

Существуют разные подходы к определению информации. Один из них отталкивается от происхождения слова «информация». Слово «информация» (от латинского *informatio* – разъяснение, изложение) в обычном понимании обозначает некоторые сведения об окружающем нас мире, которые мы используем для регулирования своего поведения. Поэтому можно определить информацию как определенным образом связанные сведения, данные, понятия, отраженные в нашем сознании и изменяющие наши представления о реальном мире.

Другой подход к определению информации отталкивается от схематичного представления процесса её передачи.

В нём можно выделить два фундаментальных элемента:

1. источник (*передатчик*);
2. приёмник (*потребитель*).

При их взаимодействии и возникает информация – некоторое сообщение, которое тем или иным способом уменьшает незнание потребителя (*приёмника*) о некотором объекте, факте или явлении. Основываясь на данном подходе, один из основоположников теории информации **Клод Шеннон** определил информацию следующим образом:

Информация - это сведения об объектах и явлениях окружающей среды, их параметрах, свойствах и состоянии, которые уменьшают имеющуюся о них степень неопределённости, неполноты знаний.

Итак, с термином “информация” связаны термины:

- *Сообщение* - информация, представленная в определенной форме (*речь, текст, изображение, цифровые данные, график, таблица*) и предназначенная для передачи.

- *Данные* - сведения, представленные в определенной знаковой системе и на определенном носителе для обеспечения возможностей их хранения, передачи, приема и обработки. Данные безотносительны к содержанию информации.

Действия, выполняемые с информацией, называются *информационными процессами*.

Информация обладает свойствами, наиболее важными из которых являются:

- ценность,
- достоверность,
- полнота,
- актуальность,
- доступность.

Ценность информации определяется тем, насколько она важна для достижения цели, стоящей перед ее получателем.

Достоверность - это определенный уровень соответствия, создаваемого с помощью полученной информации образа реальному объекту, процессу, явлению.

Полнота информации связана с тем, насколько много в ней сведений, позволяющих получателю информации достичь своей цели.

Актуальность информации определяется необходимостью ее немедленного использования получателем для достижения какой-либо цели.

Доступность - возможность получения информации в любой точке земного шара и в любое время (*кроме информации, доступ к которой ограничен законом*).

Характеристиками информации являются *ее содержание и форма представления*.

Содержание информации определяет, как и для чего, она будет использована.

Среди основных форм представления информации могут быть названы:

- *символьная (информация, представленная совокупностью букв, цифр, знаков и т.п.),*
- *графическая (различные виды изображений, анимация и т.п.),*
- *звуковая (фонетическая).*

Прежде чем говорить о содержании информации, поговорим о ее видах.

Выделяют различные *виды информации*. При этом для ее классификации по видам разработано много подходов, использующих разнообразные признаки и особенности информации.

Кодирование - это преобразование информации из одной ее формы представления в другую, наиболее удобную для её хранения, передачи или обработки.

Виды кодирования информации различают кодирование информации следующих видов: *кодирование текстовой информации; кодирование графической информации; кодирование числовой информации; кодирование звуковой информации.*

Кодирование текстовой информации.

Любой текст состоит из последовательности символов. Символами могут быть *буквы, цифры, знаки препинания, знаки математических действий, круглые и квадратные скобки* и т.д. Текстовая информация, как и любая другая, хранится в памяти компьютера в двоичном виде. Для этого каждому ставится в соответствии некоторое неотрицательное число, называемое кодом символа, и это число записывается в *память ПК* в двоичном виде. Конкретное соотношение между символами и их кодами называется системой кодировки. В персональных компьютерах обычно используется система кодировки **ASCII** (*American Standard Code for Informational Interchange* - *Американский стандартный код для информационного обмена*).

Кодирование числовой информации.

При кодировании чисел учитывается цель, с которой цифра была введена в систему: для арифметических вычислений или просто для вывода. Все данные, кодируемые в двоичной системе, шифруются с помощью единиц и нулей. Эти символы еще называют битами. Этот метод кодировки является наиболее популярным, ведь его легче всего организовать в технологическом плане: *присутствие сигнала - 1, отсутствие - 0*. У двоичного шифрования есть лишь один недостаток - это длина комбинаций из символов. Но с технической точки зрения легче орудовать кучей простых, однотипных компонентов, чем малым числом более сложных.

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

Для кодирования *действительных чисел* (это *целые, дробные числа*) используют 80 разрядное кодирование. При этом число предварительно преобразуется в *нормализованную форму*:

$$300000 = 0,3 * 10^6;$$

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

Единицы измерения информации. Информацию принято мерить *битами* и *байтами*. Самое маленькое значение это *1 бит* меньше нет, но, однако при измерении информации на носителях обычно не берут единицу ниже байта.

Байт - это набор из 8 битов.

1 символ=8битам=1байту.

Буква, цифра, знак препинания - это символы. Один пробел также является одним символом.

1 килобайт = 1024 байт

1 мегабайт = 1024 килобайт

1 гигабайт = 1024 мегабайт

1 терабайт = 1024 гигабайт

1 петабайт = 1024 терабайт и так далее.

Набор символов знаковой системы (*алфавит*) можно рассматривать как различные возможные состояния (*события*).

Тогда, если считать, что появление символов в сообщении равновероятно, количество возможных событий N можно вычислить как $N=2^i$

Количество информации в сообщении I можно подсчитать, умножив количество символов K на информационный вес одного символа i итак, мы имеем формулы, необходимые для определения количества информации в алфавитном подходе:

$N=2^i$	i	Информационный вес символа, бит.
	N	Мощность алфавита.
$I=K*i$	K	Количество символов в тексте.
	I	Информационный объем текста.

Пример:

Алфавит русского языка иногда оценивают в 32 буквы. Каков информационный вес одной буквы такого сокращенного русского алфавита?

Решение:

Дано:

$N=32$ По формуле $N=2^i$ находим
Найти: i -? $32=2^i, 2^5=2^i, i=5$

Ответ: 5



Контрольные вопросы

1. Определение информационных технологий
2. Классификация информационных технологий
3. Этапы и тенденции развития информационной технологии
4. Определите суть информационной технологии.
5. Как и для чего появилась информационная технология?
6. Полнота и достоверность информации.
7. Кодирование текстовой информации.
8. Кодирование числовой информации.
9. Каковы единицы измерения информации?

2-§. АРХИТЕКТУРА СОВРЕМЕННОГО ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА (ПК)

Ключевые слова: архитектура компьютера, процессор, память, внешние устройства, устройства ввода-вывода, операционное устройство, микропроцессор, кэш, интерфейс, адаптер, оперативная память




Архитектура персонального компьютера (ПК) включает в себя структуру, которая отражает состав ПК, и программное обеспечение.

Структура ПК – это набор его функциональных элементов (от основных логических узлов до простейших схем) и связей между ними. Архитектура определяет принципы действия, информационные связи и взаимное соединение основных логических узлов ПК, к которым относят процессор, оперативное запоминающее устройство, внешние запоминающие устройства и периферийные устройства. Основным принципом построения всех современных ПК является программное управление.

Аппаратное и программное обеспечение ПК

Персональный компьютер – двуединая система, состоящая из аппаратной части (технических устройств) и информационной части (программного обеспечения).

Современный персональный компьютер может быть реализован в вариантах:

-  настольный;
-  портативный;
-  карманный ПК.

Базовая конфигурация ПК – минимальный комплект аппаратных средств, достаточный для начала работы с компьютером:

Системный блок

Монитор

Клавиатура

Мышь



(рис.1.2.1). Базовая конфигурация ПК

Современные ПК характеризуются:

- небольшими размерами,
- возможностью для пользователя *работать с ПК лично*, без посредничества профессионального программиста,
- малым потреблением электрической энергии,
- удобством и комфортностью общения пользователя и ПК.

Современные ПК выполняют две основные функции:

- *обработка и хранение информации*
- *обмен информацией с внешними объектами.*

Выполнение этих функций осуществляется с помощью двух компонентов ПК: *программного обеспечения и аппаратного обеспечения.*

Аппаратное обеспечение компьютера – это совокупность всех устройств, предназначенных для приема, передачи, преобразования, хранения и выдачи информации.



(рис.1.2.2). Аппаратное обеспечение компьютера

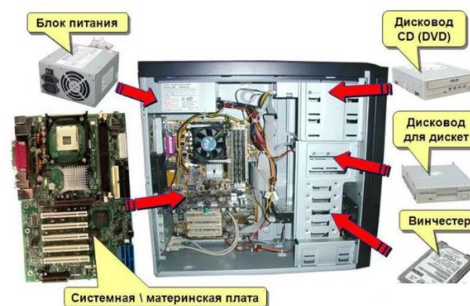
Аппаратное обеспечение современных ПК включает в себя следующее:

- системный блок,
- устройства ввода информации в ПК (*например, клавиатура*),
- устройства вывода информации из ПК (*например, монитор*).

Корпуса *системных блоков* бывают нескольких типов: вертикальный (*tower*), горизонтальный (*desktop*), моноблок (*системный блок и монитор в одном корпусе*).

Внутри *системного блока* располагаются:

Источник питания
Материнская (*системная*) плата
Процессор
Оперативная память
Жесткий диск
Привод CD-ROM, DVD-ROM,
DVD-RW
Звуковая карта
Сетевая карта и т.д.



(рис.1.2.3). *Внутреннее устройство системного блока*

Также в состав компьютерных систем входят и *внешние устройства*, называемые *периферийными*.

Периферийными называют все устройства компьютера, расположенные *вне материнской платы*. Часть устройств хоть и расположены вне материнской платы, но также, как и материнская плата находятся в системном блоке: *винчестер, дисководы, CD-ROM, звуковая карта, сетевая карта* и некоторые др.

По назначению периферийные устройства можно подразделить на:

- *устройства ввода данных; в устройства вывода данных;*
- *устройства хранения данных;*
- *устройства обмена данными.*

Теперь рассмотрим отдельные элементы более подробно.

Материнская плата - печатная плата, на которой монтируется чипсет и прочие компоненты компьютерной системы.

На материнской плате кроме чипсета располагаются разъёмы для подключения центрального процессора, *графической платы, звуковой платы, жёстких дисков, оперативной памяти* и других дополнительных периферийных устройств.

Все основные электронные схемы компьютера и необходимые дополнительные устройства включаются в материнскую плату, или подключаются к ней с помощью слотов расширения.

Центральный процессор - это «мозг» любого компьютера. Процессор производит все вычисления (*арифметические и логические операции*), взаимодействует с памятью и осуществляет управление всеми компонентами ПК. Таким образом, процессор включает в себя следующие части:

- арифметико-логическое устройство (*АЛУ*),
- устройство управления (*УУ*).
- внутренние регистры – ячейки памяти внутри кристалла процессора, предназначенные для хранения промежуточной информации.

Важнейшими характеристиками процессора, определяющими его производительность (*количество операций в единицу времени*) являются: тактовая частота, разрядность, объем адресуемой памяти.

Тактовая частота определяет скорость выполнения операций в процессоре. При повышении тактовой частоты увеличивается производительность процессора. Современные процессоры имеют тактовые частоты *400-4000 МГц* и более.

Разрядность обрабатываемых данных - количество бит информации, одновременно вводимой в процессор и выводимой из него. Чем больше разрядность, тем больше информации может обработать процессор в единицу времени. Разрядность современных процессоров – *32 и 64 бит*.

Объем адресуемой памяти (*адресное пространство*) - максимальное число ячеек основной памяти, которое может быть непосредственно адресовано процессором.

Мультипроцессор - это компьютерная система, которая содержит несколько процессоров и одно адресное пространство, видимое для всех процессоров. Он запускает одну копию ОС с одним набором таблиц, в том числе теми, которые следят, какие страницы памяти свободны. Содержат

несколько процессорных ядер в одном корпусе (*на одном или нескольких кристаллах*).

Внутренняя память - это память, расположенная на материнской плате. Внутреннюю память составляют два устройства: **ОЗУ** и **ПЗУ**.

ОЗУ (*оперативное запоминающее устройство*) предназначено для хранения текущих программ и текущей информации, т.е. программ и информации, с которыми в данный момент работает пользователь. В англоязычной литературе ОЗУ называют **RAM** (*random access memory - память случайного доступа*).

Основными характеристиками ОЗУ являются: *объем и время доступа*.

Объем ОЗУ (ед. измерения – Мбайт) - это общее количество ячеек памяти на всех кристаллах ОЗУ. В каждой ячейке может храниться либо «1» либо «0». Ячейки в кристаллах памяти объединены в блоки по 8 ячеек, и в каждый такой блок таким образом можно записать байт информации. От объема ОЗУ во многом зависит скорость работы компьютера: *чем больше объем ОЗУ, тем быстрее работает компьютер*.

Время доступа - время, за которое процессор может прочитать содержимое ячейки ОЗУ или записать в нее информацию. Чем меньше время доступа, тем быстрее общается процессор с ОЗУ и тем быстрее работает компьютер.

ОЗУ является энергозависимой памятью, т.е. при отключении питания оно «забывает» всю записанную в него информацию.

ПЗУ (*постоянное запоминающее устройство*) - хранит программу первоначальной загрузки компьютера, информацию о системной плате и расположенных на ней устройствах, информацию о подключенных устройствах внешней памяти, текущее время др.

Внешняя (периферийная) память - это память, расположенная вне материнской платы. На устройствах внешней памяти хранятся тексты программ, документы и другая информация. Эту память часто называют долговременной. Если необходимо работать с какой-то программой, то она

сначала копируется с устройств внешней памяти в оперативную память и затем запускается. Наиболее часто *внешняя память ПК* представлена накопителями на гибких магнитных дисках и накопителями на жестких дисках.

Накопитель на жёстких магнитных дисках, жёсткий диск, винчестер - энергонезависимое, перезаписываемое компьютерное запоминающее устройство. Является основным накопителем данных практически во всех современных компьютерах.

Графическая плата или *видеокарта* - устройство, преобразующее изображение, находящееся в памяти компьютера, в видеосигнал для монитора. Обычно видеокарта является платой расширения и вставляется в специальный разъем для видеокарт на материнской плате, но бывает и встроенной, иначе говоря, интегрированной.

Современные видеокарты не ограничиваются простым выводом изображения, они имеют встроенный графический микропроцессор, который может производить дополнительную обработку, разгружая от этих задач центральный процессор компьютера.

Звуковая плата или *звуковая карта* - позволяет работать со звуком на компьютере. В настоящее время звуковые карты бывают как встроенными в материнскую плату, так и отдельными платами расширения или как внешними устройствами.

Сетевая плата или *сетевая карта* - периферийное устройство, позволяющее компьютеру взаимодействовать с другими устройствами сети. Периферийные устройства для ввода информации в компьютер.

Клавиатура - устройство, с помощью которого осуществляется ввод данных и команд в ПК. Клавиатура снабжена специальным кабелем, посредством которого она подключается к специальному разъему на системном блоке.

Мышь - широко используемое в настоящее время устройство ввода информации. Позволяет быстро отметить какую-либо точку на экране

монитора. Работа с некоторыми программами без мыши практически невозможна.

Сканер - устройство ввода графической информации (*фотографий, изображений и др.*) в компьютер.

Существуют и другие устройства ввода информации в компьютер: цифровые фотоаппараты, манипуляторы «световое перо» и пр. Некоторые манипуляторы функционально аналогичны или дополняют мышь: трекбол, джойстик, виртуальный шлем и др.

Периферийные устройства для вывода информации из компьютера.

Монитор (дисплей) - устройство вывода *алфавитно-цифровой и графической* информации ПК. Монитор является основным техническим средством организации общения между пользователем и компьютером. Внешне напоминает телевизор.

Качество изображения монитора определяют следующие характеристики: *размер диагонали, разрешение, палитра, частота кадров.*

Изображение на экране монитора формируется с помощью точек (*пикселей*). Количество пикселей по горизонтали и вертикали и определяет разрешение экрана. Типичные значения: *800x600* для *14-дюймового монитора, 1600x1200* для *19-дюймового*. Чем больше размер диагонали экрана и выше разрешение, тем качественнее изображение, так как лучше прорисовываются мелкие детали.

Цветовую гамму (палитру) выводимого изображения определяет размер видеопамяти - чем больше ее объем, тем больше цветов и оттенков может вывести монитор. Для обычного пользователя, как правило, достаточно 16 тысяч цветов и оттенков.

Комфортность работы за монитором во многом определяет такая характеристика монитора, как максимальная частота смены кадров (*обычные значения: 75–100Гц, т.е. за секунду изображение на экране обновляется 75-100 раз*).

Принтер - устройство вывода, обеспечивающее печать выдаваемой компьютером информации. В качестве носителя чаще всего используется бумага.

Принтеры делятся на следующие типы: *матричные, струйные, лазерные.*

Наиболее простые принтеры - *матричные (дешевые, качество изображения низкое, уровень шума высокий).* Принцип печати таких принтеров следующий: печатающая головка содержит ряд тонких металлических *стержней (иглок).* Головка движется вдоль печатаемой строки, а стержни в нужный момент ударяют по бумаге через красящую ленту.

В *струйных принтерах* изображение формируется микрокаплями специальных чернил, выдуваемых на бумагу *с помощью сопел.* Такие принтеры обеспечивают высокое качество при печати на специальную бумагу, удобны и для цветной печати. Однако *струйные принтеры* дороже *матричных* и требуют тщательного ухода и обслуживания.

Лазерные принтеры обеспечивают самое высокое качество печати. В этих принтерах используется принцип ксерографии: изображение переносится на бумагу со специального барабана, к которому электрически притягиваются частички краски, только в отличие от ксерокса печатающий барабан электризуется *с помощью лазера* по командам компьютера.

Графопостроитель (плоттер) - устройство вывода, позволяющее получить высококачественные чертежи.

Модем (аббревиатура, составленная из слов модулятор-демодулятор) - устройство, применяющееся в системах связи и выполняющее функцию модуляции и демодуляции. Модулятор осуществляет модуляцию, то есть изменяет характеристики несущего сигнала в соответствии с изменениями входного информационного сигнала, демодулятор осуществляет обратный процесс. Частным случаем модема является широко применяемое периферийное устройство для компьютера, позволяющее ему связываться с другим компьютером, оборудованным модемом, через телефонную сеть (*телефонный модем*) или кабельную сеть (*кабельный модем*).

Подавляющее большинство компьютеров в своих основных чертах соответствует принципам **фон Неймана**, но схема устройства современных компьютеров несколько отличается от классической схемы. В частности, *арифметическо-логическое устройство* и *устройство управления*, как правило, объединены в центральный процессор. Многие быстродействующие компьютеры осуществляют параллельную обработку данных на нескольких процессорах.

Компьютерная информация хранится в электронном виде в различных запоминающих устройствах, которые называют компьютерной памятью. Для долговременного хранения информации используются постоянные носители компьютерной памяти, которые служат при вводе данных в компьютер и при выводе результатов его работы. Для хранения выполняемых в данный момент программ и промежуточных данных используется оперативная память компьютера, которая работает значительно быстрее постоянных носителей памяти.

Работа компьютера обеспечивается, с одной стороны, *аппаратными устройствами*, а с другой - *программами*.

Программное обеспечение

Программное обеспечение - наряду с *аппаратными средствами*, важнейшая составляющая информационных технологий, включающая компьютерные программы и данные, предназначенные для решения определённого круга задач и хранящиеся на машинных носителях. Программное обеспечение представляет собой либо данные для использования в других программах, либо алгоритм, реализованный в виде последовательности инструкций для процессора.

Компьютерные программы подразделяются на три категории:

Прикладные программы, которые непосредственно выполняют необходимые пользователю компьютера работы (*редактирование текстов, обработка информационных массивов, просмотр видео, пересылка*

сообщений). Пакеты прикладных программ - комплекс взаимосвязанных программ для решения определенного класса задач.

К ним относятся:

- пакеты прикладных программ, автоматизирующих бухгалтерский учет;
- пакеты прикладных программ финансовой деятельности;
- пакеты прикладных программ для управления персоналом (*кадровый учет*);
- пакеты прикладных программ управления производством;
- банковские информационные системы;

Системные программы, особую роль среди которых играет операционная система - программа, управляющая компьютером, запускающая другие программы и выполняющая сервисные функции при работе компьютера. Другие сервисные программы обычно выполняют различные вспомогательные функции - создают резервные копии используемой информации, проверяют работоспособность устройств компьютеров.

Инструментальные программы (системы программирования), которые помогают создавать новые программы для компьютера.

Сервисное программное обеспечение – программы и программные комплексы, которые расширяют возможности базового программного обеспечения и организуют более удобную среду работы пользователя.

Утилиты – программы, служащие для выполнения вспомогательных операций обработки данных или обслуживания компьютера, т.е. диагностики, тестирование аппаратных и программных средств, оптимизации использования дискового пространства, восстановление информации и т.д.

Системное программное обеспечение - это комплекс программ, которые обеспечивают эффективное управление компонентами вычислительной системы, такими как процессор, оперативная память, каналы ввода-вывода, сетевое и коммуникационное оборудование и т.п. Системное программное обеспечение реализует связь аппаратного и программного обеспечения,

выступая как "межслойный интерфейс" с одной стороны которого аппаратура, а с другой приложения пользователя. Кроме системного программного обеспечения принято выделять прикладное программное обеспечение, которое призвано решать прикладные задачи пользователя. В состав системного программного обеспечения входят: Операционные системы, среды программирования (*компиляторы, трансляторы, компоновщики, загрузчики, отладчики, текстовый редактор, библиотеки подпрограмм*), утилиты, системы управления файлами и системы управления базами данных.

К базисному программному обеспечению можно отнести **BIOS** - (*англ. Basic Input-Output System - базовая система ввода-вывода*) - программа, находящаяся в ПЗУ (*постоянном запоминающем устройстве*) персонального компьютера и исполняющаяся при включении питания. Главная функция BIOS - подготовить компьютер к тому, чтобы основное программное обеспечение (*в большинстве случаев это операционная система*), записанное на различных носителях либо доступное через сеть, могло стартовать и получить контроль над компьютером.



Контрольные вопросы

1. Аппаратное обеспечение ПК
2. Базовая конфигурация ПК
3. Современные ПК чем характеризуются?
4. Периферийные устройства
5. Программное обеспечение ПК
6. Системное программное обеспечение
7. Инструментальные программы
8. Сервисное программное обеспечение

3-§. ОПЕРАЦИОННЫЕ СИСТЕМЫ (ОС). ВИДЫ ОПЕРАЦИОННЫХ СИСТЕМ

Ключевые слова: *операционная система, приложение, пользователь, система, Windows, Mac OS, Linux, доступ, интерфейс*

Операционная система (ОС) - базовый комплекс компьютерных программ, обеспечивающий управление аппаратными средствами компьютера, работу с файлами, ввод и вывод данных, а также выполнение прикладных программ и утилит.

На сегодняшний день есть множество разновидностей операционных систем. Далеко не все пользуются большим спросом, но лишь некоторые из них. Для ПК мы рассмотрим основные ОС: *Windows, Mac OS, Linux*.



(рис.1.3.1). *Windows, Mac OS, Linux*

Принцип работы операционной системы

ОС представляет собой набор программ, которые *управляют работой компьютера*. Основные функции ОС:

- *Контролирует* загрузку процессора;
- *Управляет* ресурсами оперативной памяти, карт памяти или жестким диском;
- *Отвечает* за распределение задач и процессов;
- *Обновляет*, устанавливает и удаляет приложения и программы.

Microsoft Windows

Была создана в середине *1980-х годов*. Первой популярной операционной системой стала *Windows 95*. Ее главная особенность состояла в графическом интерфейсе. Благодаря этому пользователь мог управлять компьютером с

помощью курсора мыши. До этого нужно было использовать специальные команды, что было трудно доступным для понимания большинства людей.

На сегодняшний день – это самая популярная *операционная система* в мире. Процент пользователей составляет 90%. *Windows* устанавливается одним из трех вариантов:

- Home;
- Professional;
- Ultimate.

Самые успешные версии ОС Windows:

- Windows 10 (2015);
- Windows 8 (2012);
- Windows 7 (2009);
- Windows Vista (2007);
- Windows XP (2001).

Windows критикуют за излишнее использование *ресурсов компьютера*, нестабильную работу и постоянные надоедливые обновления. Тем не менее, *преимуществ данной ОС* больше, чем *недостатков*.

Преимущества

- Надежная поддержка *видео - аудиокарт*, принтеров и т.д.;
- Интуитивный интерфейс и легкодоступное освоение ОС;
- Простая установка программ и приложений;
- Огромный выбор программ, антивирусов, видеоредакторов, фоторедакторов, текстовых редакторов;
- Удобство в использовании.

Недостатки

- Необходимость постоянных обновлений для обеспечения безопасности;
- Не самая надежная защита от вирусов, которые могут стереть все данные или прекратить работу системы;
- Долгая загрузка ОС.

Секрет популярности *Windows* заключается в простом процессе освоении ОС самыми рядовыми пользователями всего за пару часов.

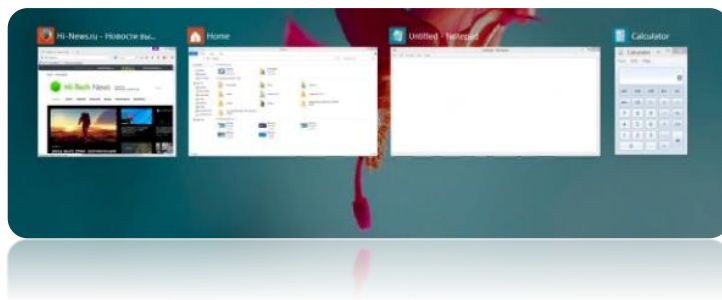
Windows 10 была выпущена после *Windows 8*, 29 июля 2015 года. Новое поколение ОС обладало большим количеством новых функций и возможностей интерфейса, которые подробно будут описаны ниже. Новое меню «*Пуск*» стало давать возможность пользователю в один клик просматривать списки часто используемых приложений и файлов, а также настраивать *приложения, программы, контакты и веб - сайты*. Теперь можно закрепить плитки в меню «*Пуск*», но при желании их можно убрать.



(рис.1.3.2). *Операционная Система Windows 10*

В *Windows 10* можно изменять размер меню «*Пуск*» с помощью курсора мыши или развернуть меню на весь экран. Недавно установленные приложения расположены в середине списка приложений меню «*Пуск*», а не отображаются и подсвечиваются при открытии диалога «*все приложения*», как это было в предыдущих *версиях Windows*. Приложения Магазина *Windows* теперь открываются в оконном режиме. С помощью новой кнопки в заголовке окна можно развернуть приложение на весь экран.

Возможность создания множества виртуальных рабочих столов с заданным набором запущенных приложений или открытых файлов и *удобное переключение* между ними. Центр уведомлений заменил *панель чудо-кнопок*. Теперь они доступны лишь в меню управления приложением в заголовке окна. Благодаря новому «*Центру уведомлений*» с кнопками быстрых настроек уведомления будут синхронизироваться с другими устройствами, работающими на *Windows 10*.



(рис.1.3.3). Рабочие столы

В *Панель задач* добавлены *панель поиска*, кнопки просмотра задач (*также рабочие столы*) и *Центра уведомлений*. Улучшенная функция *Snap* позволяет открывать сразу 4 приложения одновременно и подсказывает, какие ещё приложения запущены и как их можно разместить. Перерисованы некоторые значки. Также окна теперь имеют более современную анимацию *открытия и закрытия*. Изменена анимация при установке программ. В новой ОС рамки окон стали тоньше, также изменён их *интерфейс*.

Центр поддержки в *Windows 10* переименован в «*Безопасность и обслуживание*». Служба *Windows Hello* для входа в систему с помощью биометрических данных позволяет не вводить пароль на тех сайтах и приложениях, которые её поддерживают. Наряду с *Windows Hello*, Microsoft запускает систему, которая называется *Microsoft Passport*, предназначенную для замены пароля с помощью личных устройств, таких как смартфоны и носимые устройства, чтобы можно было пройти аутентификацию в *корпоративных системах* и *онлайн-контенте*.

К недостаткам Windows 10 можно отнести:

1. Установление на некоторых переносных ПК только со 2 - 3 й попытки.
2. Проблемы с региональными настройками. Многочисленные случаи показали, что при введении настроек своего региона ОС автоматически *устанавливает США*, с чем ничего нельзя сделать.
3. Отсутствие приложения *Media Center*, а также приложения для воспроизведения *DVD-дисков*. Аргументировали это тем, что такого рода носители в мире современных технологий устарели.

4. Проблемы с приватностью. Лицензионное соглашение W10 дает Microsoft право на сбор пользовательской персональной информации и статистики данных о местоположении, истории браузера и т.д.

Системные требования Windows 10

Следует учитывать тот факт, что данная ОС может использоваться для нескольких типов устройств - от *моноблоков* и заканчивая *мобильными телефонами* или *игровой приставкой Xbox One*. Если говорить об основной версии для компьютеров, то имеются следующие системные требования:

- *Процессор с частотой не менее 1 ГГц*
- *ОЗУ от 1 Гб (для 32х систем) и 2 Гб (для 64х систем)*
- *От 16 до 20 Гб свободного места на жестком диске*
- *Наличие DirectX 9 и выше*

Операционная система Windows 10 собрала в себе лучшее от седьмой и восьмой серии. Имеются большие возможности в плане индивидуальных настроек и красивая визуальная составляющая. Сразу после выхода ОС можно было назвать «сырой», но сейчас это уже исправлено. Все говорит в пользу того, что на протяжении следующих лет она войдет в число *наиболее популярных*.

Apple Mac OS

Линейка операционных систем, которые созданы компанией *Apple*. Это стандартная система для компьютеров Mac. По состоянию на 2014 год процент пользователей составлял от 9 до 10%.

Одна из главных причин недостатка популярности *Mac OS* – *высокие цены* на компьютеры Apple. Последней среди всех версий была выпущена Mac OS X. Ее предшественниками были:

- *Yosemite (2014);*
- *Mavericks (2013);*
- *Mountine Lion (2012);*
- *Lion (2011);*
- *Show Leopard (2009).*



(рис.1.3.4). Mac OS

Системы **Apple Mac OS** имеют сходства с *Linux*, но отличаются в том, что Mac OS имеет закрытый исходный программный код и является платной. Как и любая другая ОС, здесь тоже есть свои положительные и отрицательные стороны:

Преимущества

- *Закрытость системы* увеличивает быстродействие программ;
- *Эффективное взаимодействие* с памятью и другими процессами;
- *Надежность* в использовании;
- *Устойчивость* к вирусам.

Недостатки

- Платная основа;
- Устанавливается только на компьютеры *Apple*;
- Имеет *узкий спектр* использования (*предназначена больше для работы*).

Linux

В отличие от *Mac OS*, *Linux* имеет *открытый исходный код*. Это позволяет изменять и распространять ОС по всему миру. Например, Windows может распространять исключительно *компания Microsoft*. Операционная система была названа в честь ее создателя **Линуса Торвальдса**. Основа для Linux была заложена в 1991 году. Ее основные версии:

- Ubuntu;
- Mint;
- Fedora.



(рис.1.3.5). OS Linux

Каждая из версий имеет свой *графический дизайн*. По состоянию на 2014 год статистика пользователей Linux достигает около 2%. Большинство серверов работают на Linux из-за ее гибкости и легкости в настройках. Она состоит из пакетов, а главное сердце – ядро с установленной поддержкой драйверов.

Преимущества

- Бесплатное пользование;
- Программы и игры на Windows запускаются в Linux с помощью эмулятора;
- Возможность писать собственные программы при наличии знания языков программирования;
- Бесплатное программное обеспечение;
- Высокий уровень надежности и безопасности.

Недостатки

- Ограниченное количество игр и программ *с поддержкой Linux*;
- Ограниченная поддержка *аппаратного обеспечения*;
- *Низкая доступность* советов в случае проблем с программами или драйверами;
- Сложность в освоении.

Одним из главных факторов *низкой популярности Linux* заключается в сложности *освоения интерфейса* и *принципа работы* ОС для рядовых пользователей ПК.

Выбор ОС *целиком и полностью* ложится на плечи пользователя. Каждая система соответствует *жизненному стилю человека*, и, исходя из своих каждодневных задач, пользователь делает *выбор в пользу той или иной ОС*.



Контрольные вопросы

1. Дайте определение понятию "*операционная система*".
2. Приведите несколько способов классификации операционных систем.
3. Назовите требования к современным операционным системам и объясните, что они означают.
4. Этапы развития операционных систем
5. Когда *Windows 10* официально выпущена?
6. Какой нужен компьютер для *Windows 10*?
7. Системные требования *Windows 10*
8. Преимущества и недостатки ОС *Windows 10*
9. Самые успешные версии ОС Windows?

ПРАКТИЧЕСКИЕ РАБОТЫ

4-§. СЕРВИСНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, УТИЛИТЫ

Цель работы: *Научиться использовать специализированные программы для сервисного обслуживания компьютера и операционной системы.*

Краткая теоретическая информация:

Сервисные программы расширяют возможности ОС Windows по обслуживанию системы и обеспечивают удобство работы пользователя. К этой категории относят *системы технического обслуживания, программные оболочки ОС, а также программы-утилиты (служебные программы).*

Программы-утилиты (служебные программы) - это программы, служащие для выполнения вспомогательных операций обработки данных при обслуживании компьютеров (*диагностики, тестирования аппаратных и программных средств, оптимизации использования дискового пространства, восстановления разрушенной на магнитном диске информации, защиты от компьютерных вирусов и др.*).

Утилиты могут распространяться тремя способами:

1. Как самостоятельные программы;
2. Входить в состав многофункциональных пакетов сервисных утилит;
3. Входить в состав операционной системы (*в виде служебных программ*).

В ОС Windows входит ряд *служебных программ*:

1. **Проверка диска** – исправляет *физические и логические дефекты* на диске, помечает сбойные блоки, чтобы данные не записывались в них. Она позволяет избежать множество проблем, связанных с потерей информации. Программа обнаруживает и устраняет ошибки, связанные со структурой

файлов и каталогов, внутренней структурой сжатых дисков, с длинными именами файлов. К ошибкам файловой структуры относятся потери фрагментов файлов или наличие фрагментов старых файлов. Возникают они обычно при попытке разместить два файла на одном участке диска либо при отказе в освобождении места при удалении файла. Физические дефекты, связанные с поверхностью диска, выявляются в режиме полной проверки, которая занимает значительно больше времени, чем стандартная проверка. Зато позволит записи данных в такие участки диска, из которых невозможно будет считать.

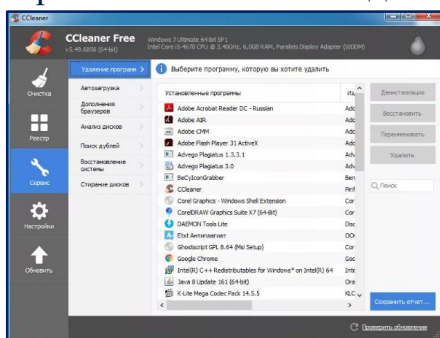
2. **Дефрагментация диска** – перераспределяет информацию на жестком диске, оптимизируя ее организацию. Когда в область данных незаполненного диска производится запись *первых файлов*, эти файлы помещаются в непрерывные последовательности кластеров. Затем, когда, в процессе работы происходит *удаление некоторых файлов* и запись на их место новых, файлы неизбежно разбиваются на фрагменты. Такая ситуация складывается вследствие интенсивной эксплуатации диска, когда часто *записываются и удаляются* большие программы или файлы данных. Если файлы записаны на диске не непрерывно, а фрагментами, то доступ к нему занимает больше времени, чем могло бы быть при записи его непрерывным потоком. Нельзя допускать чрезмерной фрагментации жесткого диска, которая замедляет чтение данных и ограничивает производительность ПК. При излишней фрагментации быстрее обычного изнашивается узел жесткого диска. После запуска программы, необходимо нажать кнопку дефрагментация диска в диалоговом окне. Запускается программа, которая упорядочивает расположение данных на диске и тем самым ускоряет доступ к файлам и загрузку программ. Дефрагментация жесткого диска обычно занимает от десятка минут до нескольких часов. Обязательным условием выполнения дефрагментации диска является наличие не менее 15% свободного дискового пространства.

3. **Очистка диска** – программа помогает очистить пространство на жестком диске. Программа очистки диска проверяет диск и выводит перечень *временных файлов, файлов Интернета, а также ненужных программных файлов*, удаление которых не приведет к негативным последствиям. Можно выбрать удаление некоторых или всех этих файлов.

4. **Уплотнение диска** – сжимает данные на жестком диске и дискетах. Сжатые диски, создаваемые с целью экономии места на реальных физических дисках, сами не являются настоящими дисковыми устройствами. Сжатый диск - это *логическая структура*, представляющая собой сжатый файл, который хранится на одном из обычных несжатых дисков. Этот файл называется **CVF-файлом** (*файл сжатого тома*) и располагается в корневом каталоге обычного диска. *CVF-файл* имеет атрибуты скрытого, системного и защищенного от записи файла. Суть уплотнения с помощью *DriveSpace* заключается в том, что создаваемый *CVF-файл* содержит объем информации больший, чем размер занимаемого им пространства на несущем диске. Например, *CVF-файл* может занимать *100 Мбайт* дискового пространства, в то время как объем размещенных в нем данных может составлять *200 Мбайт*.





Для запуска стандартных служебных программ необходимо выполнить *Пуск → Все программы → Стандартные → Служебные* или зайти *Мой компьютер → вызвать контекстное меню любого логического диска → пункт Свойства → вкладка Сервис*.

Программа-утилита CCleaner - бесплатная программа для оптимизации и очистки системы, а также инструмент обеспечения безопасности пользователя. *CCleaner* удаляет ненужные и неиспользуемые файлы, освобождает пространство на жестких дисках и позволяет запускать *Windows* быстрее.



(рис.1.4.1). - Окно программы *CCleaner*

Программа содержит четыре раздела, размещенные в левой части окна:

-  **Очистка** (для выполнения очистки системы от временных файлов),
-  **Реестр** (очищает реестр от устаревших ссылок),
-  **Сервис** (для удаления программ, настройка автозагрузки, поиска файлов, восстановления системы и стирания (форматирования) дисков),
-  **Настройки** (для настройки программы).

Практическое задание

1. Проверьте диск **D:/** на наличие физических и логических дефектов.
2. Выполните анализ диска **D:/** с помощью программы дефрагментации.
3. Выполните очистку диска **D:/** с помощью служебной программы.
4. Запустите *CCleaner*. Выполните анализ системы в разделе Очистка.
5. Выполните поиск проблем реестра в разделе Реестр.
6. Изучите раздел Сервис. Запишите, какие четыре метода работы с установленными программами предоставлены в разделе Удаление программ.
7. Сохраните отчет об установленных программах под своей фамилией в папке группы.
8. Результаты выполненного практического задания запишите в отчет.



Контрольные вопросы

1. Что такое программа-утилита?
2. Какие стандартные служебные программы входят в *OS Windows*?
3. Какие виды повреждений дисков Вам известны и как они возникают?
4. Каково обязательное условие для выполнения дефрагментации?
5. Какие инструменты (разделы) предлагает *CCleaner* и для чего они предназначены?
6. Какие критерии поиска файлов можно указать в программе-утилите *CCleaner*?

5-§. ПОДКЛЮЧЕНИЯ УСТРОЙСТВ К СИСТЕМНОМУ БЛОКУ

Цель работы: Изучение основных компонентов персонального компьютера и основных видов периферийного оборудования, способов их подключения, основных характеристик (*название, тип разъема, скорость передачи данных, дополнительные свойства*). Определение по внешнему виду типов разъемов и подключаемого к ним оборудования получение сведений по настройке пользовательского интерфейса периферийных устройств средствами операционной системы **Microsoft Windows**.

Краткая теоретическая информация:

Периферийные устройства (ПУ) - аппаратура, предназначенная для внешней обработки информации. Другими словами, это устройства, расположенные вне системного блока – внешние устройства.

Подключения устройств к системному блоку

Все *периферийные устройства* подключаются только к системному блоку. Для работы конкретного устройства в составе конкретного комплекта ПК необходимо иметь:

1. **Контроллер (адаптер)** – специальную плату, управляющую работой конкретного периферийного устройства. Например, *контроллер клавиатуры, мыши, адаптер монитора, портов* и т.п.

2. **Драйвер** – специальное программное обеспечение, управляющее работой конкретного периферийного устройства. Например, *драйвер клавиатуры, драйвер принтера* и т.п.

Для управления работой устройств в компьютерах используются электронные схемы – *контроллеры*. Различные устройства используют разные способы подключения к контроллерам:

✚ некоторые устройства (*дисковод для дискет, клавиатура и т.д.*) подключаются к имеющимся в составе компьютера стандартным контроллерам (*интегрированным или встроенным в материнскую плату*);

✚ некоторые устройства (*звуковые карты, многие факс-модемы и т.д.*) выполнены как электронные платы, т. е. смонтированы на одной плате со своим контроллером;

✚ некоторые устройства используют следующий способ подключения: в системный блок компьютера вставляется электронная плата (*контроллер*), управляющая работой устройства, а само устройство подсоединяется к этой плате кабелем;

✚ на сегодняшний день большинство внешних устройств подключаются к компьютеру через *USB-порт*.

Платы контроллеров вставляются в специальные разъемы (*слоты*) на материнской плате компьютера.

С помощью добавления и замены плат контроллеров пользователь может модифицировать компьютер, расширяя его возможности и настраивая его по своим потребностям. Например, пользователь может добавить в компьютер *факс-модем, звуковую карту, плату приема телепередач* и т.д.

Одним из видов контроллеров, которые присутствуют почти в каждом компьютере, является контроллер портов *ввода-вывода*. Часто этот контроллер интегрирован в состав материнской платы. Контроллер портов ввода-вывода соединяется кабелями с разъемами на задней стенке компьютера, через которые к компьютеру подключаются принтер, мышь и некоторые другие устройства.

Кроме контроллеров портов ввода-вывода в системном блоке присутствуют разъемы шины *USB* – универсальной последовательной шины, к которой можно подключить *клавиатуру, мышь, принтер, модем, дисковод компакт-дисков, сканер* и т.д. Основное требование возможность подключения к данной шине устройства. Особенность шины *USB* –

возможность подключения к ней устройств во время работы компьютера (*не выключая его*).

В отличие от внутренних компонентов, для установки периферийных устройств не нужно открывать корпус.



(рис1.5.1). Внешние разъемы системного блока

Периферийные устройства подключаются к разъемам на внешней части корпуса с помощью проводов или беспроводной связи. Исторически периферийные устройства разработаны в расчете на определенный тип портов. Например, в конструкции принтеров для персональных компьютеров предусмотрено подключение к параллельному порту, который передает от компьютера к принтеру данные в определенном формате.

Разработанный интерфейс универсальной последовательной шины (*USB*) сильно упростил использование проводных периферийных устройств. *USB-устройства* не требуют сложных процедур конфигурации. Они просто подключаются к соответствующему порту (*при наличии нужного драйвера*). Кроме того, все чаще появляются устройства, которые подключаются к узлу с использованием беспроводной технологии.

Установка периферийного устройства выполняется в несколько этапов. Порядок и тип этих шагов зависит от типа физического подключения и от того, относится ли устройство к типу автоматически настраивающихся (*PnP*). Предусмотрены следующие шаги:

- ✚ подсоединение периферийного устройства к узлу с помощью соответствующего кабеля или беспроводного соединения;
- ✚ подключение устройства к источнику питания;
- ✚ установка соответствующего драйвера.

Некоторые устаревшие устройства, так называемые "*обычные устройства*", не предусматривают самонастройки. Драйверы таких устройств устанавливаются после того, как устройство подключается к компьютеру и включается питание.

Драйверы самонастраивающихся *USB-устройств* в системе уже имеются. В таком случае при подключении и включении операционная система распознает устройство и устанавливает соответствующий драйвер.

Организация электропитания устройств компьютера

Все устройства, имеющиеся внутри системного блока (*как внутренние, так и внешние*) питаются через блок питания системного блока.

Монитор поддерживает две схемы подключения:

- через системный блок (*в этом случае при включении/выключении системного блока одновременно включается/выключается монитор*);
- через отдельную розетку (*тогда при включении системного блока монитор нужно включить через отдельную кнопку*).

Все остальные периферийные устройства, расположенные в отдельных корпусах, имеют отдельные кабели питания, подключаемые к собственным розеткам.

Несмотря на высокую надежность и безопасность, компьютерное оборудование должно быть заземлено.

Если в составе комплекта компьютерного оборудования нет UPS, то желательно использовать хотя бы обычный стабилизатор напряжения или специальный сетевой фильтр.

Разъемы для подключения электропитания и внешних устройств **Расположение разъемов**

Обычно разъемы для подключения электропитания и внешних устройств находятся на задней стенке системного блока компьютера. В портативных компьютерах эти разъемы могут находиться и с боковых сторон системного блока.

Подключение электропитания и внешних устройств к компьютеру выполняется с помощью специальных проводов (*кабелей*). Для защиты от ошибок разъемы для вставки этих кабелей сделаны разными, так что кабель, как правило, просто не вставится в неподходящее гнездо.

Разъемы для электропитания

Обычно на задней стенке системного блока компьютера имеется разъем для подсоединения к электросети и разъем для подачи электропитания на монитор.

Разъемы для подключения внешних устройств

Разъемы для подключения внешних устройств достаточно унифицированы. Распространены разъемы со штырьками и с гнездами.

Порядок вставки кабелей

Вставлять и вынимать кабели из разъемов можно только при выключенном компьютере, в противном случае компьютер и устройства могут быть испорчены.

Вставлять кабели надо аккуратно, чтобы не погнуть и не поломать содержащиеся в разъемах штырьки.

Некоторые кабели (*например, для монитора или принтера*) закрепляются с помощью винтов, эти винты надо завернуть рукой или отверткой (*только не слишком туго*), чтобы кабель не выпадал из разъема во время работы компьютера.

Периферийные устройства

К внешним или периферийным устройствам персонального компьютера относится:

1. Устройствами, которые предназначены для ввода информации в компьютер для обработки, и устройствами, для вывода информации из него.

2. Есть еще и устройства для хранения информации вне системного блока (*внешние накопители*).

Клавиатура и мышь

Это основные устройства для *ввода информации*, необходимость в них очевидна и не обсуждается, пока сенсорные мониторы не вытеснили обычные:



(рис.1.5.2). Клавиатура и мышь

Подключаться клавиатура и мышь к компьютеру может с помощью двух видов разъемов:

- **PS/2**
- **USB.**



(рис.1.5.3). Разъемы на задней панели системного блока

Клавиатуру и мышь с разъемом **PS/2** можно подключать или отключать только при выключенном компьютере. Если они выполнены с **USB** - разъемами, то их можно подключать или отключать в любой момент, даже если компьютер работает.

Акустические системы (колонки)

Акустические системы преобразуют электрический сигнал, полученный со звуковой карты компьютера в звуковые колебания (*т.е. в звук*) и относятся к устройствам вывода информации. На сегодняшний день акустические

системы можно назвать обязательными устройствами ПК. Чтобы слушать музыку, смотреть фильмы, слушать аудиокниги, а также изучать IT-уроки, без «колонок» не обойтись.



(рис.1.5.4). Акустические системы (колонки)

Наушники

Наушники можно считать разновидностью акустических систем, только предназначены они для одного пользователя ПК. Фактически, наушники – это миниатюрные акустические системы для индивидуального использования.



(рис.1.5.5). Наушники

Подключаются наушники к компьютеру так же, как и акустические системы.

Микрофон

Микрофон может понадобиться в двух случаях:

- ✚ Общение через Интернет (*например, через Skype*);
- ✚ Запись звука с целью дальнейшего хранения и обработки на ПК.

Как можно догадаться, микрофон является устройством *ввода информации*.



(рис.1.5.6). Микрофон

Веб-камера (Web-камера)

Веб-камера может понадобиться, если Вы планируете часто общаться через Интернет, и при этом хотелось бы передавать не только голос, но и своё изображение (*следует учитывать, что скорость Интернета должна быть достаточной для передачи видеосигнала*).



(рис.1.5.7). Веб-камера на ноутбуке (встроенная)

Веб-камера может иметь встроенный микрофон, что освобождает Вас от покупки «лишнего» устройства.

USB-флэш-накопитель (флэшка)

Флэш-накопители относятся к устройствам хранения данных, т.е. предназначены для *хранения и переноса информации* между компьютерами.



(рис.1.5.8). USB-флэш-накопитель (флэшка)

*Из названия можно догадаться, что подключается USB-флэш-накопитель с помощью **USB-разъема**.*

Обычно нельзя подключить к компьютерной системе новое устройство, не установив или не настроив соответствующее программное обеспечение. К счастью, операционная система *Windows* обычно делает это автоматически.

Несмотря на автоматическое распознавание, необходимо внимательно изучить инструкции по установке, которые поставляются с каждым устройством. Иногда перед установкой нужно устанавливать специальное программное обеспечение. Как об этом узнать? Прочитать инструкцию!

Способность Windows автоматически распознавать новые устройства называется **plug-and-play** (*подключай и играй*).

Программа, управляющая работой устройства, называется драйвером.

Порядок выполнения работы

Задание: Осуществить подключение периферийного оборудования, используя различные интерфейсы: жесткий диск, привод на компакт-дисках, флоппи-дисковод, плату видеоадаптера, сетевую карту, звуковую карту, монитор, принтер, сканер.

Изучение разъемов для подключения электропитания и внешних устройств

Посмотрите на обратную сторону системного блока с подключенными кабелями. Поочередно вынимая кабель конкретного устройства зарисуйте разъемы системного блока. Результаты изучения разъемов занесите в таблицу.

Устройство

Схема разъема, к которому подключено

- | | |
|---------------------|-------------------|
| 1. Электропитание | 5. Монитор |
| 2. Клавиатура | 6. Сетевой кабель |
| 3. Мышь | 7. USB порт |
| 4. Питание монитора | |

Подключение и настройка принтера.

Подключите принтер к системному блоку.

Произведите установку программного обеспечения принтера - *драйвера*:

1. Запустите команду *Пуск – Панель управления – Принтеры и факсы*
2. В открывшемся окне Принтеры и факсы выберите команду *Файл - установить принтер*
3. Далее действуйте по шагам мастера установки.

Таким образом, принтер установлен.

Результат можно посмотреть в окне Принтеры и факсы (*Пуск – Панель управления – Принтеры и факсы*). Ваш принтер появился в списке принтеров.

Использование свойств принтера при печати

Откройте текстовый редактор (*Пуск – Все программы – Microsoft Office - Microsoft Word 2013*)

Запустите команду *Печать* главного меню окна.

Изучите все возможности печати документов (*какие изменения можно производить при печати*).

Результат зафиксируйте в отчете.



Контрольные вопросы

1. *Расскажите*, как осуществляется питание различных устройств ПК.
2. *Посчитайте* количество необходимых розеток электропитания, если в состав ПК входят сканер и принтер.
3. *В чем состоит* особенность электропитания мониторов?
4. *Какие устройства* используются в ПК для стабилизации напряжения в сети?
5. *Какое устройство* в составе ПК позволяет стандартно завершить работу аппаратуры при внезапном отключении электропитания?
6. *Как осуществляется* подключение электропитания и внешних устройств в компьютере?

ЛАБОРАТОРНЫЕ РАБОТЫ

6-§. ТЕСТИРОВАНИЕ КОМПЬЮТЕРА (ПРОЦЕССОР, ВИДЕОКАРТА, HDD, RAM)

Цель работы: *Познакомиться с программным обеспечением для тестирования компьютера. Научиться определять характеристики и работоспособность устройств ПК для оптимизации системы*

Краткая теоретическая информация:

Видеокарта

Видеокарта - это устройство, выводящее на экран все действия и процессы, совершаемые на компьютере. При помощи видеокарты данные, передаваемые компьютером, преобразовываются в видеосигнал и передаются на монитор.

Существуют два основных типа видеокарт: *интегрированные* и *дискретные*.

Интегрированные (встроенные) являются неотъемлемой частью *центрального процессора* или *материнской платы*, так как встроены в них. Наличие интегрированного видео снижает стоимость и энергопотребление компьютера, однако такие видеокарты обладают ограниченной производительностью. У них зачастую отсутствует собственная видеопамять, поэтому они используют **ОЗУ компьютера**. Кроме этого, у интегрированной видеокарты нет отдельной системы охлаждения, что позволяет сэкономить пространство внутри *системного блока* или *ноутбука*. Такой тип видеокарт в основном используется в *офисных* и *простых домашних компьютерах*, где не требуется высокая мощность и производительность.

Дискретная видеокарта - это отдельная плата, устанавливаемая в специальный слот (*PCI Expressx16*) и содержащая в себе всё необходимое для

полноценной работы. Это позволяет увеличить компьютеру производительность и использовать его для игр с высоким уровнем графики или работы с мощными графическими приложениями. У дискретных видеокарт имеется собственная видеопамять. Их главными недостатками являются дороговизна и потребление большого количества энергии, что играет важную роль для ноутбуков.



(рис.1.6.1). Дискретная видеокарта для настольного компьютера



(рис.1.6.2). Дискретная видеокарта для ноутбука

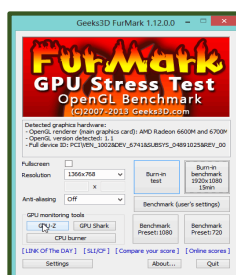
Многие считают, что чем дороже видеокарта (и чем больше у нее памяти) - тем она производительнее! На самом деле это далеко не всегда так. Вообще, чтобы оценить производительность своей видеокарты - необходимо провести спец. тест (как правило, на него потребуется 10-15 минут).

Для этого существует определенное количество программ, которые могут провести полное тестирование видеокарты и выдать заключение о ее работоспособности.

FurMark

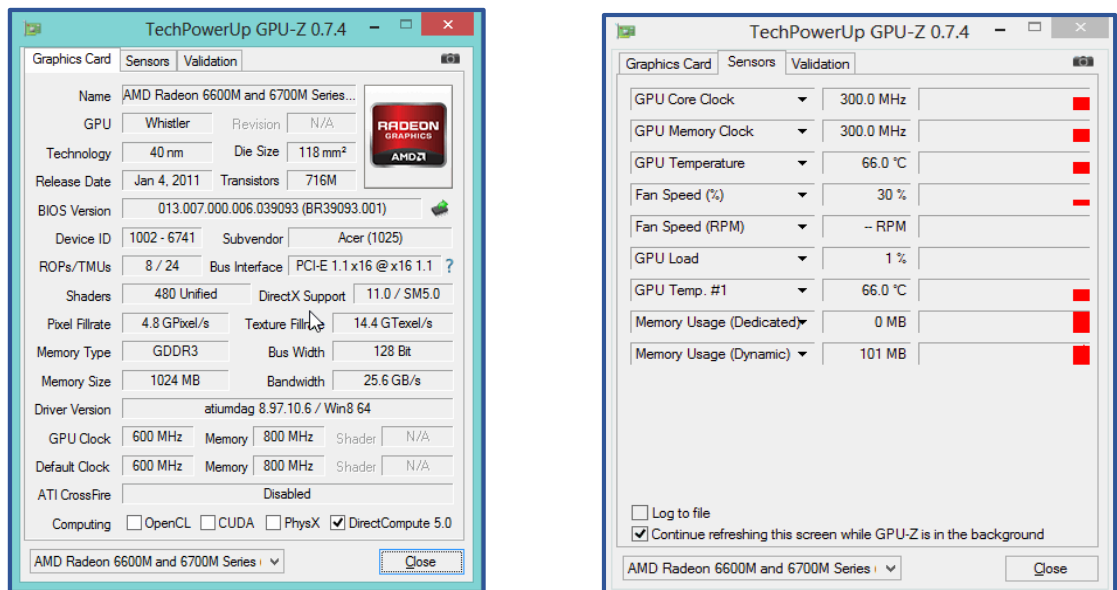
Поддерживает все современные ОС Windows и Vista. Кроме этого, позволяет реально оценить производительность вашей видеокарты.

После установки и запуска программы должно появиться следующее окно:



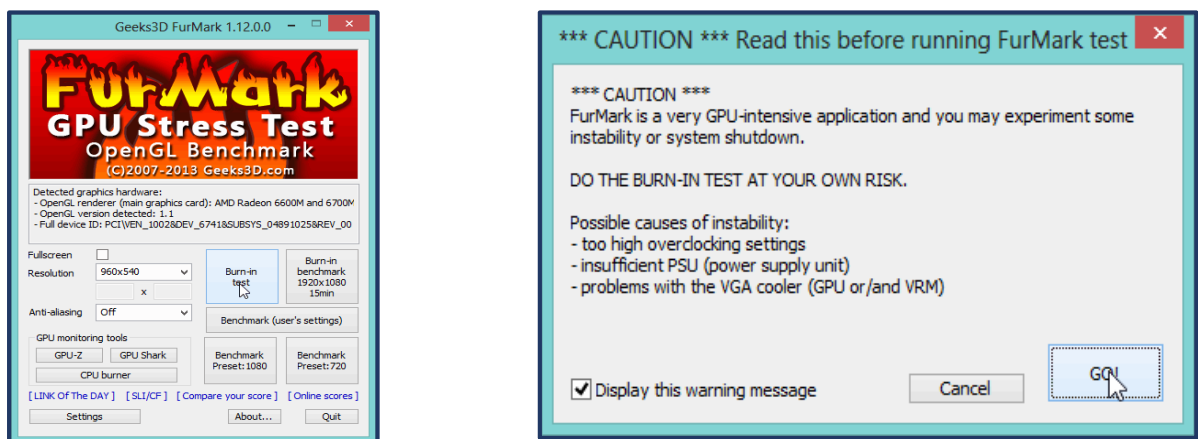
(рис.1.6.3). Окно программы FurMark.

Для просмотра сведений о параметрах видеокарты нужно нажать на кнопку CPU-Z. Здесь можно узнать модель видеокарты, ее дату выпуска, версию БИОСа, DirectX, память, частоты процессора и пр. Рядом же есть вкладка «Sensors»: она показывает нагрузку на устройство в данный момент времени и температуру нагрева устройства. Кстати, эту вкладку можно не закрывать во время теста.



(рис.1.6.4). CPU-Z, вкладка «Sensors»

Для начала тестирования видеокарты нажмите в главном окне на кнопку «Burn in test», далее на кнопку «GO».



(рис.1.6.5). Окно тестирования

Ждите около 15 мин. В это время *загрузка вашей видеокарты* будет на *максимуме!*

Итоги теста

Если по истечении 15 мин. компьютер не перезагружался, не зависал - то можете считать, что видеокарта прошла тест.

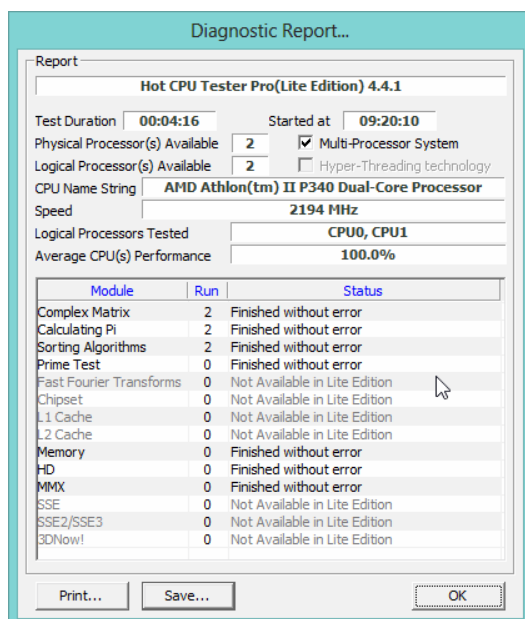
Важно еще обратить внимание на температуру процессора видеокарты (*можно посмотреть во вкладке Sensor*). Температура не должна подняться выше 80⁰-С. Если выше - есть риск, что видеокарта может начать вести себя нестабильно.

Процессор

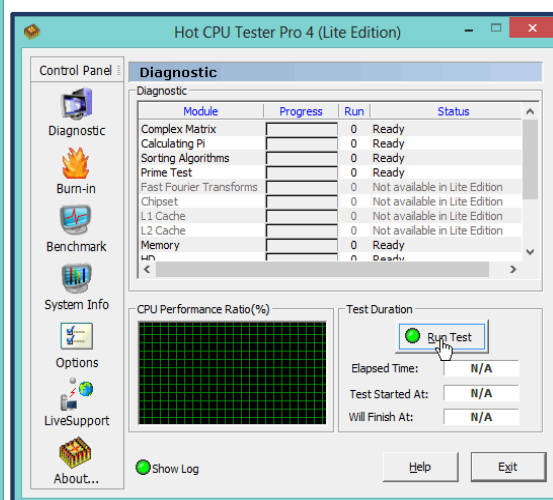
Неплохой утилитой для *тестирования процессора* является *7Byte Hot CPU Tester*. При первом запуске утилиты появится следующее окно. Для *начала тестирования* можно сразу же нажать кнопку **Run Test**. Кстати, перед этим лучше закрыть все *посторонние программы, игры* и пр., т.к. при тестировании процессор будет нагружен и все приложения начнут существенно тормозить.

После проведения тестирования предоставится отчет, который, кстати, можно вывести даже на печать.

В большинстве случаев, особенно, если тестируется новый компьютер, одного факта - что во время тестирования не было сбоев - будет достаточно, чтобы признать процессор нормальным для работы.



(рис.1.6.6). Окно диагностики



(рис.1.6.7). Окно отчета

Оперативная память (Ram)

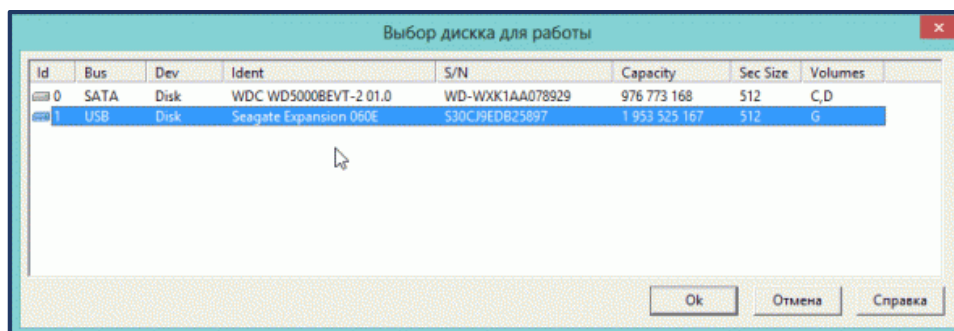
Одна из лучших утилит для тестирования оперативной памяти является **Memtest**.

1. Скачайте утилиту *Memtest*.
2. Создайте загрузочный диск *CD/DVD* или *флешку*.
3. Загрузитесь с него и проверьте память.
4. **Жесткий диск (HDD)**

Для тестирования жестких дисков есть множество утилит.

PC3000DiskAnalyzer - бесплатная свободно распространяемая утилита для проверки работоспособности жестких дисков. К тому же, утилита поддерживает все самые популярные носители, среди которых: *HDD, SATA, SCSI, SSD, External USB HDD/Flash*.

После запуска, утилита предлагает выбрать жесткий диск, с которым будете работать.



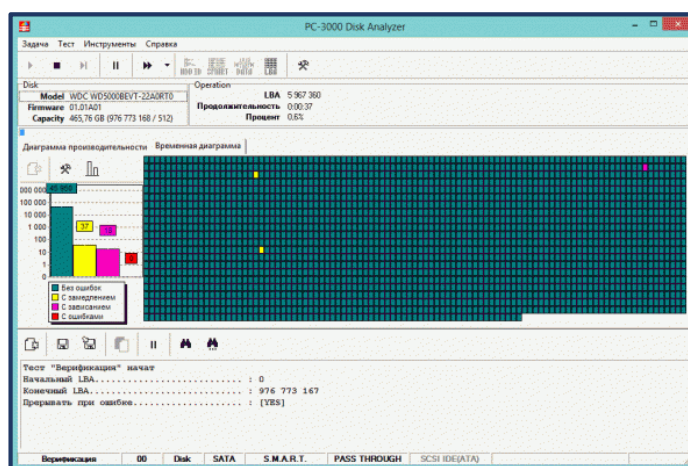
(рис.1.6.8). Окно выбора дисков

Далее появится основное окно программы. Для начала тестирования нажмите кнопку **F9**, либо «*тест/запуск*».

Далее будет предложен один из вариантов тестирования:

Выберите «*верификацию*», этого достаточно, чтобы проверить скорость работы жесткого диска, проверить сектора, какие отвечают быстро, а какие уже выдают ошибки.

На такой диаграмме хорошо видно, что ошибок практически нет, есть совсем небольшое количество секторов, отвечающих с замедлением (*это не страшно, даже на новых дисках есть такое явление*).



(рис.1.6.9). Диаграмма секторов

Монитор (на битые пиксели)

Чтобы картинка на мониторе была качественной и передавала ее в полной мере - на нем не должно быть битых пикселей.

Битый - это значит, что в этой точке не будет отображаться никакой из цветов. Не всегда их можно заметить на той или иной картинке, т.е. нужно на мониторе последовательно сменять цвета и смотреть: если есть битые пиксели вы их должны заметить, когда начнете менять цвета.

Проводить такую процедуру лучше при помощи *специальных утилит*. Например, очень удобная **IsMyLcdOK** (для 32 и 64 битных систем). Устанавливать ее не нужно, работает сразу же после запуска.


```

IsMyLcdOK 2.44
IsMyLcdOK, это небольшая программа, но эффективное решение.
Для теста используйте клавиши:
<<
[1] = Проверка белого           // [2] = Проверка черного
[3] = Проверка красного         // [4] = Проверка зеленого
[5] = Проверка синего           // [6] = Проверка голубого
[7] = Проверка фиолетового     // [8] = Проверка желтого
[9] = Горизонтальный градиент  // [0] = Вертикальный градиент
[F7] или [V] = Вертикальные линии // [F8] или [H] = Горизонтальные линии
>>
[F2] = Проверка BitBlt MB/сек. // [F3] = Окрашенные прямоугольники
[F4] = Окрашенные полосы      // [F5] = Проверка долговечности

[Любая другая клавиша] = Следующий тест

[ESC] = Выход / [F1] = Этот текст / [L] = Language

[ENTER] = Домашняя страница (Nenad Hrg)!
[П] = Перевести

Я надеюсь, что "Вы ничего не нашли" ;) !
(c) Nenad Hrg / SoftwareOK.com

```

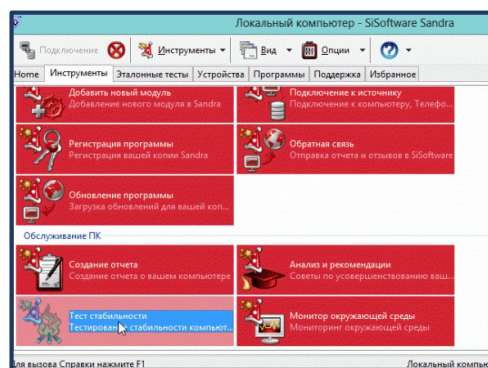
(рис.1.6.10). Утилита IsMyLcdOK

Нажимаете последовательно цифре на клавиатуре и монитор будет закрашиваться в разные цвета. Внимательно наблюдайте за точками на мониторе.

Общий тест компьютера

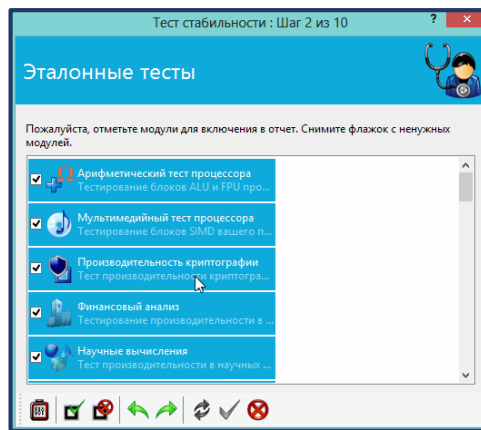
Нельзя не отметить и еще одну *утилиту*, которая может протестировать компьютер сразу по десяткам параметров. **SiSoftware Sandra Lite** *бесплатная утилита*, которая предоставит сотни параметров и сведений о вашей системе и сможет провести тестирование десятка устройств.

Для начала тестирования зайдите во вкладку «инструменты» и запустите «тест стабильности».



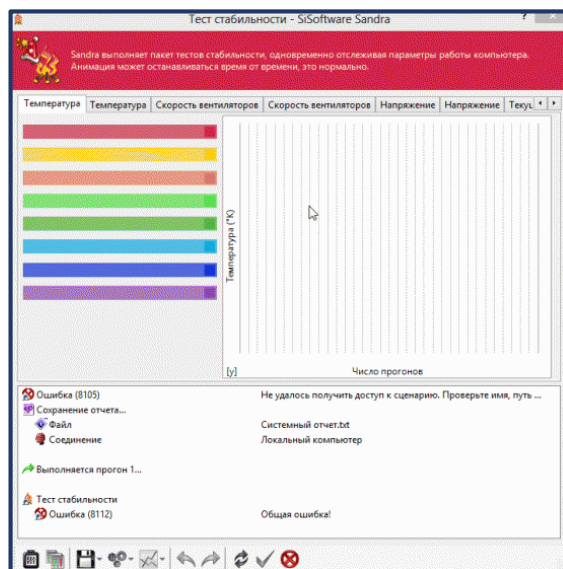
(рис.1.6.11). Утилита SiSoftware

Напротив, нужных проверок ставьте галочки. Проверить можно: *процессор, оптические приводы, флешки, скорость передачи на телефон/кпк, оперативную память* и т.д. Причем, для того же процессора десятков различных тестов, начиная от производительности криптографии и заканчивая арифметическими вычислениями.



(рис.1.6.12). Окно тестирования

После пошаговых настроек и выбора места, куда вы сохраните файл-отчет о проведенном тесте, программа начнет работу.



(рис.1.6.13). Окно тест на стабильности

На этом *тестирование компьютера* завершено.



Контрольные вопросы и задания

1. Определите объём оперативной памяти, занимаемый операционной системой и время загрузки
2. Выполните оптимизацию функций системы
3. Повторите тестирование вашего компьютера
4. Сделайте вывод о проделанной работе
5. Осуществите поиск бесплатной программы тестирования, установите программу на домашнем ПК и примените её для тестирования.

6. Оформите отчет о проделанной работе

7-§. НАСТРОЙКА КОМПЬЮТЕРА С ПОМОЩЬЮ BIOS SETUP

Цель работы: Развитие профессиональных навыков работы с базовой системой ввода-вывода (*BIOS*)

Краткая теоретическая информация:

SETUP – программа установки параметров *BIOS*, первоначальной настройки компьютера. Запускается нажатием специальной клавиши, которая зависит от производителя и версии *BIOS*.

BIOS SETUP - вспомогательная программа, записанная в постоянном запоминающем устройстве **ПЗУ**, хранящая информацию о текущей конфигурации компьютера и предоставляющая пользователю средства для настройки параметров системы. Эти параметры хранятся в *энергонезависимой памяти CMOS* и сохраняются при *отключении компьютера*. При включении компьютера, система конфигурируется в соответствии с параметрами, записанными в **CMOS** памяти.

Данная программа позволяет произвести первоначальную настройку аппаратных средств компьютера, например, *задать порядок загрузки, указать параметры памяти* и т.д.

Через простое меню пользователь может сделать настройки следующих устройств и функций:

- ✓ *Жесткие диски, приводы и периферийные устройства.*
- ✓ *Последовательность загрузки операционной системы.*
- ✓ *Защита системы от несанкционированного доступа.*
- ✓ *Режим энергосбережения и др.*

Вызов программы **SETUP** производится в момент начальной загрузки компьютера при появлении на экране монитора надписи «*Press «DEL» to enter SETUP»* («Нажмите клавишу «DEL» для входа в SETUP»). После нажатия клавиши «DEL» на экране появляется главное меню. Программа BIOS постоянно совершенствуется, поэтому описание данной версии может отличаться от установленной на *Вашем компьютере*.

Типичными являются следующие *пункты меню*:

<i>CMOS SETUP UTILITY</i>	<i>LOAD SETUP (BIOS) DEFAULTS</i>
<i>STANDART CMOS SETUP</i>	<i>SUPERVISOR PASSWORD</i>
<i>BIOS FEATURES SETUP</i>	<i>USER PASSWORD</i>
<i>CHIPSET FEATURES SETUP</i>	<i>IDE HDD AUTO DETECTION</i>
<i>POWER MANAGEMENT SETUP</i>	<i>SAVE & EXIT SETUP</i>
<i>PNP AND PCI SETUP</i>	<i>EXIT WITHOUT SAVING</i>

Самотестирование при включении:

При включении компьютера или после нажатия кнопки сброса («*RESET*» на *передней панели ПК*) программа *BIOS* осуществляет проверку основных ресурсов системной платы:

- ✓ центрального процессора
- ✓ оперативной памяти
- ✓ приводов жестких и гибких дисков
- ✓ клавиатуры и др.

Чтобы войти в *BIOS Setup*, нужно во время первоначального *тестирования компьютера* нажать определенную клавишу или их комбинацию. Наиболее часто используется *Delete*, *F1* или *F2*; есть и другие варианты. Узнать, за какой же клавишей закреплен вход в *BIOS Setup*, можно из инструкции к системной плате или из подсказки, которая появляется во время прохождения процедуры **POST** и имеет, например, такой вид:

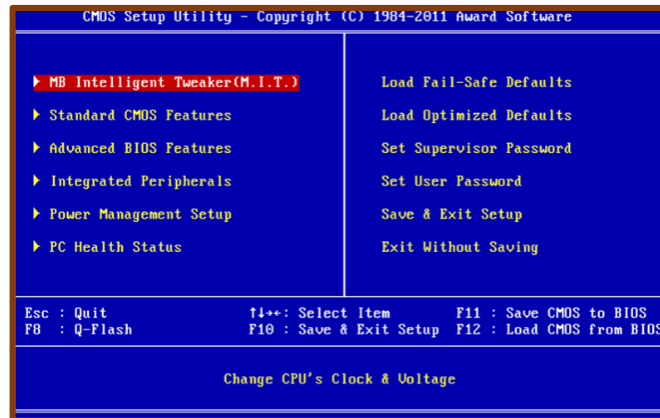
Press DEL to enter SETUP

Если инструкции к плате нет, а экранная подсказка отсутствует, можно последовательно попробовать наиболее известные варианты:

 *Delete*;

- + Одну из функциональных клавиш: *F1, F2, F3, F10, F11, F12*;
- + *Esc*;
- + *Ctrl+Shift+S* или *Ctrl+Alt+S*;
- + *Ctrl+Alt+Esc* или *Ctrl+Alt+Delete*.

Интерфейс *BIOS Setup Utility*



(рис.1.7.1). Интерфейс *BIOS Setup Utility*

Для выхода из *BIOS Setup* есть два способа:

- + с отменой всех внесенных изменений;
- + с сохранением всех внесенных изменений.

Некорректные установки BIOS могут привести к повреждению компьютера.

Содержание и порядок выполнения лабораторной работы.

Примечание: Так как изменение настроек в **CMOS** может привести к выходу компьютера из строя. Проводите все работы *на эмуляторе BIOS*. Выполните следующие действия:

Работа выполняется под руководством преподавателя.

1. Включите системный блок;
2. При появлении информации на экране нажмите клавишу *Delete* - произойдет запуск программы *SETUP* и откроется основное меню;
3. С помощью клавиш управления курсором выберите пункт меню **STANDARD CMOS SETUP** (*Стандартные настройки микросхемы CMOS*);

4. В открывшемся окне проверьте установку *системных часов, системного календаря, количество и объем жестких дисков*;
5. Вернитесь в предыдущее меню с помощью клавиши *Esc*;
6. Выберите пункт **BIOS FEATURES SETUP** (*Настройки параметров BIOS*);
7. В открывшемся окне проверьте:
 - ✓ с *какого диска* начинается запуск компьютера. Последовательность запуска задается в пункте *BOOT SEQUENCE*;
 - ✓ состояние защиты компьютера от несанкционированного доступа: задается пункт *Security Option. Setup* - отключено (*для режима настройки*), *System* - включено (*для штатной работы системы*);
8. Вернитесь в предыдущее меню нажатием клавиши *Esc*;
9. Выберите пункт *IDE HDD AUTO DETECT* (*Автоматическое определение жестких дисков*). Обратите внимание на порядок тестирования дисков;
10. Завершите работу *с программой SETUP* без сохранения результатов изменения. Для этого нажмите клавишу *Esc* и при получении запроса подтвердите выход без сохранения изменений нажатием клавиши **Y** (*Yes - Да*).

Заполните таблицу:

Элемент конфигурации	Маркировка, тип	Дополнительные характеристики	Значение
BIOS			
Процессор		тип наличие сопроцессора тактовая частота	
Оперативная память		тип объем	
Жесткий диск		количество объем	
Дисководы гибких дисков		количество тип	
Порты ввода-вывода		количество: параллельные последовательные	



Контрольные вопросы и задания

1. Как расшифровывается ПЗУ?
2. Что такое BIOS?
3. Каково основное назначение программ пакета BIOS?
4. Для чего предназначена программа SETUP?
5. Какова дополнительная задача программы SETUP?
6. Как вызывается программа SETUP?
7. Какие клавиши необходимо использовать для наблюдения сообщений, поступающих от компьютера в процессе запуска?
8. Какую процедуру запускает процедура инициализации?
9. В какой таблице можно просмотреть данные, определяющие состав компьютерной системы и ее настроек?
10. Как вернуться в предыдущее меню программы SETUP?
11. В каком меню программы SETUP можно просмотреть последовательность запуска компьютера?

8-§. ПРОГРАММЫ-ОБОЛОЧКИ И АРХИВАТОРЫ

Цель работы: Научиться архивировать файлы с разными значениями сжатия.

Краткая теоретическая информация:

Системные программы (СП) служат для управления ресурсами ПК - процессором, памятью, вводом-выводом. Это программы общего пользования, предназначенные для эффективной работы прикладных программ.

Класс *СП вспомогательного назначения* – **утилиты** (лат. *Utilitas - польза*). Они расширяют и дополняют возможности ОС, либо решают самостоятельные задачи. Часть утилит входит в состав ОС, а другая часть функционирует автономно.

Деление утилит по объектам:

✚ программы контроля, тестирования и диагностики функциональных блоков ПК;

✚ программы-драйверы - Системные программы для управления внешними устройствами. Драйверы стандартных устройств образуют в совокупности базовую систему ввода-вывода (*BIOS*);

✚ программы-упаковщики (*архиваторы*) **WinZip** и **WinRar** позволяют записывать информацию на дисках более плотно, а также объединять копии нескольких файлов в один архивный файл. Это удобно применять для переноса файлов с одного ПК на другой при помощи дискет и сети **Internet**;

✚ *антивирусные программы* предназначены для предотвращения заражения вирусами и ликвидации последствий заражения;

✚ программы *оптимизации* и контроля качества дискового пространства;

✚ программы *восстановления информации*, форматирования, защиты данных;

✚ *коммуникационные программы*, организующие обмен информацией между ПК;

✚ программы *управления памятью*, обеспечивающие гибкое использование ОП;

✚ программы для записи *CD-ROM, CD-R* и многие другие.

Архиватор - программа, предназначенная для упаковки без потерь одного и более файлов в единый **файл-архив** или в **серию архивов** для удобства переноса и/или хранения данных. Распаковка архивов выполняется с

помощью того же архиватора либо посредством сторонних совместимых утилит.

Необходимость архивации связана с *резервным копированием информации* на диски и дискеты с целью сохранения программного обеспечения компьютера и защиты его от порчи и уничтожения (*умышленного, случайного или под действием компьютерного вируса*). Чтобы уменьшить потери информации, следует иметь резервные копии всех программ и файлов.

Программы-упаковщики (архиваторы) позволяют за счет специальных методов сжатия информации создавать копии файлов меньшего размера и объединять копии нескольких файлов в один архивный файл. Это даёт возможность на дисках или дискетах разместить больше информации, то есть повысить плотность хранения информации на единицу объёма носителя (*диска*).

Наиболее высокоэффективными являются архиваторы *RAR, ACE, AIN, ARJ*.

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 направлять в *формате ZIP*. Можно также направить самораспаковывающийся архив. Такой архив является немного большим, но может быть извлечен без *внешних программ*. Другое преимущество *ZIP* – **скорость**. *Архив ZIP* обычно создается быстрее, чем *RAR*.

Архив в формате RAR

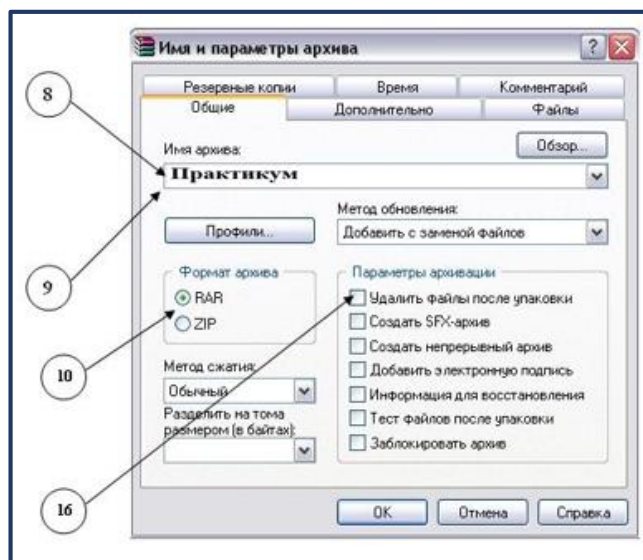
Формат RAR в большинстве случаев обеспечивает *значительно лучшее сжатие, чем ZIP*. Кроме того, *формат RAR* обеспечивает поддержку *многотомных архивов*, имеет средства восстановления *поврежденных файлов*, архивирует файлы практически *неограниченных размеров*. Необходимо отметить, что при работе в файловой *системе FAT32* архивы могут достигать

только 4 гигабайт. Работа с большими размерами архива поддерживается только в файловой системе NTFS.

Технология выполнения работы:

1. *Создайте* в своей рабочей папке (*папке с вашей группой*) следующие папки: *папку со своей фамилией, в ней папки Архивы*.
2. *Запустите* программу **WinRar**.
3. *Откройте* папку с исходным материалом для практической работы **Практикум**. В этой папке хранятся *три типа файлов .doc, .bmp, .exe*.

Ответьте на вопрос письменно в тетради, какого типа документы имеют вышеперечисленные расширения?



(рис.1.8.1). Окно параметров архивации

4. *Скопируйте* в папку Архивы файлы из папки Практикум.
 5. *Заархивируйте графический файл* и сравните размеры обоих файлов.
- Для этого выполните следующие действия:
6. Щелчком правой кнопки мыши выделите файл типа *.bmp*
 7. Щелкните на кнопке «*Добавить в архив...*», появится диалоговое окно, уточняющее параметры архивации.
 8. *По умолчанию* архивный файл имеет имя исходного файла.
 9. Если необходимо задать иное имя архива, то введите его в поле ввода имени.
 10. Выберите формат архивного файла, *например, RAR*.

11. Остальные параметры оставьте без изменения.
12. Щелкните по кнопке **ОК**.
13. Сравните размеры исходного файла и архивного. Данные внесите в *таблицу 1*.

14. Заархивируйте файл типа *.doc* и сравните размеры обоих файлов. Данные внесите в *таблицу (1.18.1)*.

15. Заархивируйте файл типа *.exe* и сравните размеры обоих файлов. Данные внесите в *таблицу (1.18.1)*.

16. Удалите исходные файлы.

17. Заархивируйте файлы в формате *архива ZIP*. Заполните *таблицу (1.18.1)* полученными данными.

Внимание! Возможно упаковывать файлы с их последующим удалением, если был выбран такой метод.

Формат архива	Имя файла и его расширение	Исходный размер	Размер после архивации

(табл.1.8.1). Таблица для заполнения

При создании нового архива нужно задать параметры архивирования. Прежде всего, необходимо задать имя архивного файла и место его сохранения на диске. Далее, нужно выбрать формат архивации *RAR* или *ZIP* (*формат ZIP более широко распространен, а метод RAR обеспечивает больше возможностей и более сильное сжатие*).

В обоих форматах поддерживаются шесть *методов архивации*:

- | | |
|---------------|-----------------|
| ✓ Без сжатия, | ✓ Обычный, |
| ✓ Скоростной, | ✓ Хороший |
| ✓ Быстрый, | ✓ Максимальный. |

Архив может быть **непрерывным** (позволяет добиться максимальной степени сжатия) и **самораспаковывающимся** (*SFX*, от англ. *Self-Extracting*). Для разархивации такого архива не нужна специальная программа, достаточно запустить файл архива на выполнение, так как он является исполняемым файлом и имеет расширение *.exe*.

Для архивации звуковых и графических файлов может использоваться дополнительный специальный метод **мультимедиа сжатие**, при котором может быть достигнута на 30% более высокая степень сжатия, чем при обычном сжатии.

Разархивирование файлов

Выполнить эту операцию можно двойным кликом по файлу архива, либо через контекстное меню - *Извлечь файлы*.

Антивирусы

Антивирусные программы - это программы, основной задачей которых является защита именно от вирусов, или точнее, от вредоносных программ.



(рис.1.8.2). Антивирусные программы

Методы и принципы защиты теоретически не имеют особого значения, главное, чтобы они были направлены на борьбу с вредоносными программами. Но на практике дело обстоит несколько иначе: практически любая

антивирусная программа объединяет в разных пропорциях все *технологии* и *методы защиты* от вирусов, созданные к сегодняшнему дню.

Из всех методов *антивирусной защиты* можно выделить две основные группы:

✚ **Сигнатурные методы** - точные методы обнаружения вирусов, основанные на *сравнении файла* с известными образцами вирусов

✚ **Эвристические методы** - приблизительные методы обнаружения, которые позволяют с определенной вероятностью *предположить*, что файл заражен.



Антивирус Касперского - *антивирусное программное обеспечение, разрабатываемое лабораторией Касперского.*

Признаки заражения ПК вирусом. Желательно не допускать появление вирусов в ПК, но при заражении компьютера вирусом *очень важно его обнаружить.*

Основные признаки появления вируса в ПК.

- 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 и другие признаки.

Антивирус Касперского 7.0

Программа состоит из *следующих компонентов:*

✚ **Файловый Антивирус** - компонент, контролирующий файловую систему компьютера. Он проверяет все открываемые, запускаемые и сохраняемые файлы на компьютере.

✚ *Почтовый Антивирус* – компонент проверки всех входящих и исходящих почтовых сообщений компьютера.

✚ *Веб-Антивирус* – компонент, который перехватывает и блокирует выполнение скрипта, расположенного на «*веб - сайте*», если он представляет угрозу

✚ *Проактивная защита* - компонент, который позволяет обнаружить *новую вредоносную программу* еще до того, как она успеет нанести вред. Таким образом, компьютер защищен не только от уже *известных вирусов*, но и от *новых*, еще не исследованных.

Антивирус Касперского 7.0 – это классическая защита компьютера от вирусов, *троянских* и *шпионских программ*, а также от любого *вредоносного ПО*.

Основные функции:

Три степени защиты от известных и новых *интернет-угроз*:

1. Проверка по базам сигнатур
2. Эвристический анализатор
3. Поведенческий блокиратор

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
- ✓ Обнаружение всех видов руткитов
- ✓ Автоматическое обновление баз

Kaspersky Lab

1. *Запустить Kaspersky Scanner*, во вкладке «*Объекты*» выбрать диск *A:*, в правой области действий выбрать «*Спросить пользователя*», и запустите проверку значком ►.

2. *Выбрать «Статистика»* и записать отчет о проверке

3. Во вкладке «*Объекты*» выбрать диск **C:**, папку **Мои Документы**, в правой области действий выбрать «*Лечить*», и запустите проверку значком ►
Если такая папка отсутствует в перечне объектов тестирования – *Добавить*.

4. Выбрать «*Статистика*» и записать *отчет* о проверке.



Контрольные вопросы и задания

1. Что такое *архивирование*?
2. Для чего нужно *резервное копирование*?
3. Что такое *разархивация*?
4. Что такое *архивация* и *архиваторы*?
5. Какие функции выполняют *архиваторы*?
6. Сформулируйте *цель* и *задачи* архивации.
7. В чём назначение *антивирусных программ*?
8. Какие методы обновления *антивирусных баз* вы знаете?
9. Как настроить обновление *антивирусных баз* без выхода в *Интернет*?
10. Как настроить *почтовый* и *Веб-антивирус*?

ГЛАВА 2. СОВРЕМЕННЫЕ ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

МОДУЛЬ 2

9-§. АЛГОРИТМ И ЕГО СВОЙСТВА

Ключевые слова: *алгоритм, свойства алгоритма, дискретность, понятность, определённость, результативность, массовость, исполнитель, характеристики исполнителя, круг решаемых задач, среда, режим работы, система команд, формальное исполнение алгоритма.*

Слово «алгоритм» происходит от имени хорезмского учёного **Абу Абдуллах Мухаммеда ибн Муса аль-Хорезми**. Около 825 года он написал сочинение, в котором впервые дал описание придуманной в Индии позиционной десятичной системы счисления. Аль Хорезми сформулировал правила вычислений в новой системе и, вероятно, впервые использовал **0** для обозначения пропущенной позиции в записи числа (*её индийское название арабы перевели как as-sifr или просто sifr, отсюда такие слова, как «цифра» и «шифр»*).

Приблизительно в это же время индийские цифры начали применять и другие арабские учёные. В первой половине XII века книга аль Хорезми в латинском переводе проникла в Европу. Переводчик дал ей название **Algoritmi de numero Indorum** («Алгоритмы о счёте индийском»). Латинизированное имя среднеазиатского учёного было вынесено в заглавие книги, и сегодня считается, что слово «алгоритм» попало в европейские языки именно благодаря этому сочинению.

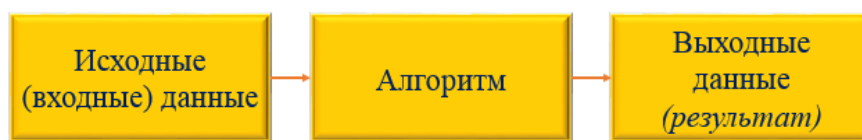
Алгоритм - это последовательность команд, предназначенная исполнителю, в результате выполнения которой он должен решить поставленную задачу.

Алгоритм должен описываться на *формальном языке*, исключающем неоднозначность толкования.

Исполнитель алгоритма - человек или *техническое устройство*, которые понимают команды алгоритма и умеют правильно их выполнять. Совокупность команд, которые исполнитель может понять и выполнить, называется *системой команд исполнителя*.

Исполнитель *должен уметь* выполнять все команды, составляющие алгоритм. Множество возможных команд конечно и изначально строго задано. Действия, которые выполняет исполнитель по этим командам называются элементарными.

Для выполнения алгоритма исполнителю недостаточно только самого алгоритма. *Выполнить алгоритм* - значит применить его к решению конкретной задачи, т.е. выполнить запланированные действия по отношению к определенным входным данным. Поэтому исполнителю необходимо иметь *исходные (входные) данные* - те, что задаются до начала алгоритма.



(рис.2.9.1). Выполнения алгоритма

В результате выполнения алгоритма исполнитель должен получить искомый результат - *выходные данные*, которые исполнитель выдает как результат выполненной работы. В процессе работы исполнитель может создавать и использовать данные, не являющиеся выходными, - *промежуточные данные*.

Запись алгоритма на формальном языке называется программой.

Алгоритм обладает следующими свойствами:

1. **Дискретность.** Процесс решения задачи должен быть разбит на последовательность отдельных *шагов-команд*, которые выполняются одна за другой. Только после завершения *одной команды* начинается выполнение *следующей*.

2. **Понятность.** Алгоритм должен содержать только те команды, которые *известны исполнителю*.

3. **Детерминированность.** Каждый шаг и переход от шага к шагу должны быть точно определены, чтобы его мог выполнить любой другой человек или *механическое устройство*. У исполнителя нет возможности принимать самостоятельное решение (*алгоритм исполняется формально*).

4. **Конечность.** Обычно предполагают, что алгоритм заканчивает работу за *конечное число шагов*. Результат работы алгоритма также должен быть получен за конечное время. Можно расширить понятие алгоритма до понятия процесса, который по различным каналам получает данные, выводит данные и потенциально может не заканчивать свою работу.

5. **Массовость.** Алгоритм должен решать не одну *частную задачу*, а всех задач *данного типа*.

Способы представления алгоритмов:

✚ **Словесная запись** (*на естественном языке*). Алгоритм записывается в виде последовательности пронумерованных команд, каждая из которых представляет собой произвольное изложение действия;

✚ **Блок–схема** (*графическое изображение*). Алгоритм представляется с помощью специальных значков (*геометрических фигур*) - блоков;

✚ **Формальные алгоритмические языки.** Для записи алгоритма используется специальная система обозначений (*искусственный язык, называемый алгоритмическим*);

✚ **Псевдокод.** Запись алгоритма *на основе синтеза* алгоритмического и обычного языков. Базовые структуры алгоритма записываются строго с помощью элементов некоторого базового алгоритмического языка.

Словесная запись алгоритма

Произвольное изложение этапов алгоритма на естественном языке имеет свои недостатки. Словесные описания строго не формализуемы, поэтому может быть нарушено свойство определенности алгоритма: исполнитель может неточно понять описание этапа алгоритма. Словесная запись

достаточно многословна. Сложные задачи трудно представить в словесной форме.

Пример:

При *словесно-формульном способе* алгоритм записывается в виде текста с формулами по пунктам, определяющим последовательность действий.

Пусть, например, необходимо найти значение следующего выражения: $y=2a-(x+6)$.

Словесно-формульным способом алгоритм решения этой задачи может быть записан в следующем виде:

1. Ввести значения, a и x .
2. Сложить x и 6 .
3. Умножить, a на 2 .
4. Вычесть из $2a$ сумму $(x+6)$.
5. Вывести y как результат вычисления выражения.

Формальные исполнители алгоритма

Формальный исполнитель - это исполнитель, который выполняет все команды алгоритма *строго в предписанной последовательности*, не вникая в его смысл, не внося ничего в алгоритм и ничего не отбрасывая. Обычно под формальным исполнителем понимают технические устройства, автоматы, роботов и т.п. *Компьютер* можно считать *формальным исполнителем*.

Программы на языке произвольного формального исполнителя могут состоять только из элементарных команд, которые входят в его систему (*которые исполнитель «понимает»*).



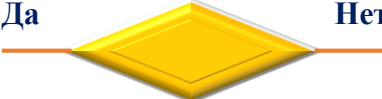
Исполнитель может иметь свою среду (*например, систему координат, клеточное поле и др.*).

Среда исполнителя - это совокупность объектов, над которыми он может выполнять определенные действия (*команды*), и связей между этими объектами. Алгоритмы в этой среде выполняются исполнителем по шагам.

Блок–схема

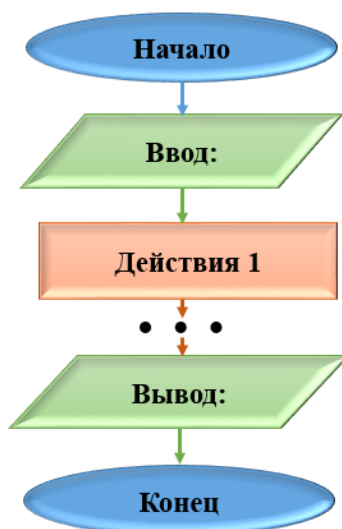
Блок–схема - наглядный способ *представления алгоритма*. Блок–схема отображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий. Определенному типу действия соответствует определенная *геометрическая фигура* блока. *Линии, соединяющие блоки*, определяют очередность выполнения действий. По умолчанию блоки соединяются *сверху вниз и слева направо*. Если последовательность выполнения блоков должна быть иной, используются направленные линии (*стрелки*).

Основные элементы блок–схемы алгоритма:

Внешний вид блока	Назначение блока
	Блок <i>начала/конца</i> алгоритма. Используется в начале и конце блок-схемы алгоритма.
	Блок <i>ввода/вывода</i> . Служит для организации ввода исходных данных и вывода результирующих данных.
	Функциональный блок (<i>процесс</i>). Служит для указания действия (<i>шага</i>) алгоритма.
Да  Нет	Альтернативный блок (<i>условный блок, условие</i>). Служит для указания выбора одного из двух возможных действий.

(табл.2.9.1). Элементы блок–схемы

Общий вид блок–схемы алгоритма:



(рис.2.9.2). Общий вид блок–схемы

Алгоритмические языки

Алгоритмический язык - это искусственный язык (*система обозначений*), предназначенный для записи алгоритмов. Он позволяет представить алгоритм в виде текста, составленного по определенным правилам с использованием *специальных служебных слов*. Количество таких слов ограничено. Каждое служебное слово имеет точно определенный смысл, назначение и способ применения. При записи алгоритма служебные слова выделяют *полужирным шрифтом* или *подчеркиванием*.

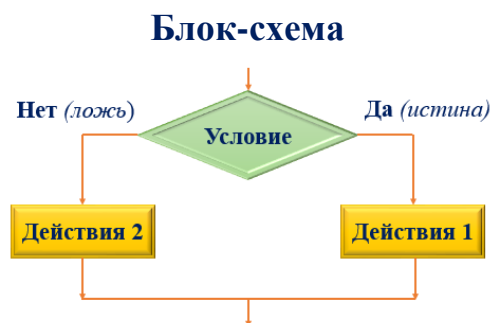
В алгоритмическом языке используются формальные конструкции, но нет строгих синтаксических правил для записи команд. Различные алгоритмические языки различаются набором служебных слов и формой записи основных конструкций.

Псевдокод

Псевдокод занимает промежуточное положение между *естественным языком* и *языками программирования*. Благодаря этому алгоритмический язык позволяет легче освоить *основы программирования*.

Пример: Вывести значение наибольшего из двух чисел.

- Псевдокод**
1. **Ввод** двух чисел a, b .
 2. **ЕСЛИ** $a > b$, **ТО** «выводим a »,
ИНАЧЕ «выводим b ».
 3. **Конец**.



(рис.2.9.3). Полная форма ветвления

Классификация алгоритмов

Алгоритмы в зависимости от порядка действий исполнителя подразделяются следующим образом:

Линейный алгоритм - набор команд (*указаний*), выполняемых последовательно во времени *друг за другом*.

Разветвляющийся алгоритм - алгоритм, содержащий хотя бы одно условие, в результате проверки которого может осуществляться *разделение на несколько параллельных ветвей алгоритма*.

Циклический алгоритм - алгоритм, предусматривающий *многократное повторение* одного и того же действия (*одних и тех же операций*) над новыми исходными данными. К циклическим алгоритмам сводится большинство методов вычислений, перебора вариантов.

Пример: Алгоритм поиска *вещественных корней* квадратного уравнения:

$$ax^2+bx+c=0, a\neq 0.$$

1. Задать значения коэффициентов a , b и c .
2. Вычислить дискриминант $D=b^2-4ac$.
3. Проверить условие существования вещественных корней $D\geq 0$. Если $D\geq 0$ выполнено, то перейти к шагу 5. Если не выполнено (*т.е. $D<0$*), то перейти к шагу 4.
4. Дать ответ - «*вещественных корней нет*». Перейти к шагу 8.
5. Проверить условие существования двух различных корней $D>0$. Если условие выполнено, то перейти к шагу 7, иначе - перейти к шагу 6.
6. Вычислить корень уравнения $x = -b/2a$. Дать ответ - «*уравнение имеет один корень x* ». Перейти к шагу 8.
7. Вычислить корни уравнения $x_1 = \frac{-b+\sqrt{D}}{2a}$, $x_2 = \frac{-b-\sqrt{D}}{2a}$. Дать ответ - «*уравнение имеет два корня x_1, x_2* ».
8. Закончить вычисления.



Контрольные вопросы и задания

1. Что такое алгоритм?
2. Как можно представить алгоритм?
3. Как выглядит блок проверки логического условия в блок-схемах?
4. Какие виды алгоритмов вы знаете?
5. Какой алгоритм называется линейным?

6. Какой алгоритм называется разветвляющимся?
7. Какой алгоритм называется циклическим?
8. Разработать алгоритм вычисления гипотенузы прямоугольного треугольника по известным значениям длин его катетов a и b .
9. Разработать алгоритм вычисления суммы натуральных чисел от 1 до 100 .

10-§. ЯЗЫКИ ПРОГРАММИРОВАНИЯ. ЯЗЫК ПРОГРАММИРОВАНИЯ PASCAL

Ключевые слова: язык программирования, Pascal ABC, программа, синтаксис, семантика, тестирование, отладка, выходные данные, переменная, оператор, идентификатор, зарезервированное слово, тип данных, арифметическое выражение, операторные скобки, модуль CRT, begin

Язык программирования – искусственный (формальный) язык,
предназначенный для записи компьютерных программ.

Язык программирования определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно -ПК) под её управлением.

Со времени создания первых программируемых машин человечество придумало более восьми тысяч языков программирования. Каждый год их число увеличивается. Некоторыми языками умеет пользоваться только небольшое число их собственных разработчиков, другие становятся известны миллионам людей.

Язык программирования предназначен для написания компьютерных программ, которые представляют собой набор правил, позволяющих компьютеру выполнить тот или иной вычислительный процесс, организовать управление различными объектами, и т.п. Язык программирования отличается от естественных языков тем, что предназначен для управления ПК, в то время как естественные языки используются, прежде всего, для общения людей

между собой. Большинство языков программирования использует специальные конструкции для определения и манипулирования структурами данных и управления процессом вычислений.

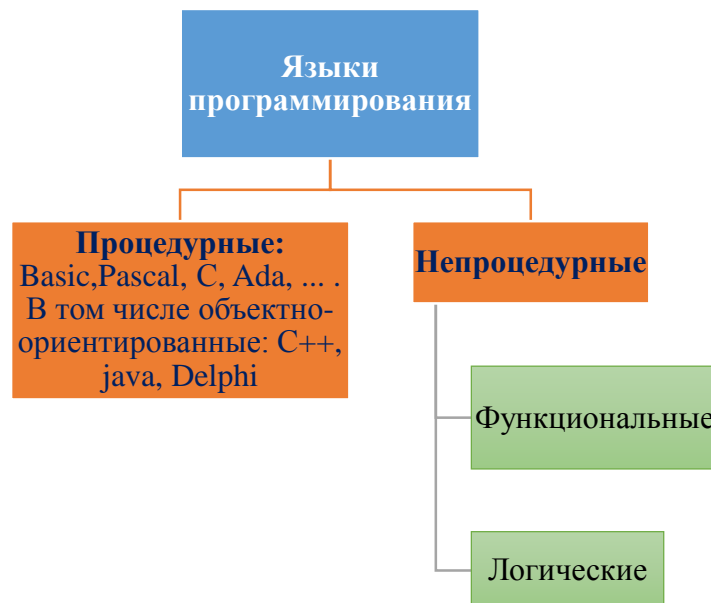
Классификация языков программирования

Языки программирования можно разделить на два класса:

- ✚ процедурные;
- ✚ непроцедурные.

Процедурные (императивные) языки - это языки операторного типа. Описание алгоритма на этом языке имеет вид последовательности операторов. Характерным для процедурного языка является наличие оператора присваивания (*Pascal, Си, Ада, ...*). Программа, написанная на императивном языке, очень похожа на *приказы*, выражаемые повелительным наклонением в естественных языках, то есть это *последовательность команд*, которые должен выполнить компьютер. Программируя в императивном стиле, пользователь должен объяснить компьютеру, как нужно решать задачу.

Непроцедурные (декларативные) языки - это языки, при использовании которых в программе в явном виде указывается, какими свойствами должен обладать результат, но не говорится, каким способом он должен быть получен.



(струк.2.10.1). Классификация языков программирования

Структура языка программирования

Между существующими языками программирования есть принципиальные расхождения в концепции построения языков, особенно это справедливо для более ранних языков, но все эти языки потому и называются языками программирования, что они с точки зрения внутренней системы построения имеют одинаковое формальное строение.



(рис.2.10.1). Структура языка программирования

Любой язык программирования состоит из предложений (*операторов*). Предложения (*как и слова*) определены над неким алфавитом S . Синтаксис языка описывает множество предложений над алфавитом S , которые внешне представляют правильно сформированные программы.

Синтаксис языка - это правила *получения слов и предложений* этого языка. Синтаксис схематически описывается с помощью определенных грамматических правил.

Знание формального языка (*алфавита + синтаксиса*) хотя и достаточно для установления синтаксической корректности программы, однако недостаточно для понимания ее назначения и способа действий. Значение и способ действия программы на языке программирования уточняются путем задания семантики.

Семантика языка - это правила интерпретации слов формального языка, т.е. установления значения отдельных языковых элементов.

Для определения формальных языков, в том числе для языков программирования, используют *БНФ (формы Бэкуса - Наура)* и синтаксические диаграммы. Это два взаимозаменяемых способа описания.

Язык программирования Pascal

Pascal - язык профессионального программирования, который назван в честь французского математика и философа **Блеза Паскаля (1623-1662)** и

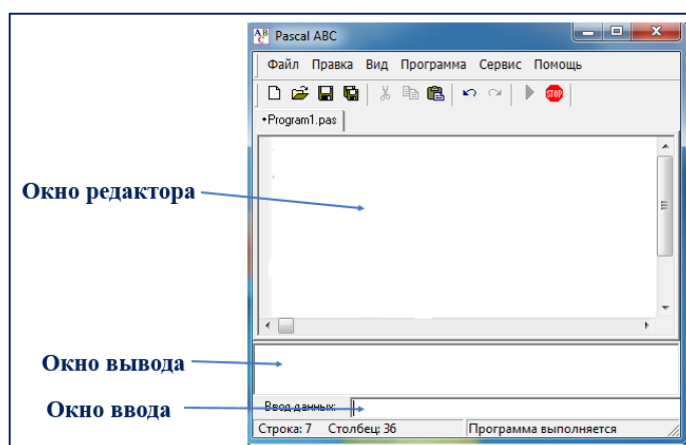
разработан в 1968-1971 гг. **Никлаусом Виртом**. Первоначально был разработан для обучения, но вскоре стал использоваться для разработки программных средств в *профессиональном программировании*.

Pascal популярен по *следующим причинам*:

1. Прост для обучения.
2. Отражает *фундаментальные идеи алгоритмов* в легко воспринимаемой форме, что предоставляет исполнителю средства, помогающие *проектировать программы*.
3. Позволяет четко реализовать идеи *структурного программирования* и *структурной организации данных*.
4. Использование простых и гибких структур управления: *ветвлений, циклов*.
5. Надежность *разрабатываемых программ*.

Система программирования Pascal ABC

Система программирования **Pascal ABC** ориентирована на *школьников и студентов младших курсов*. Это в достаточно простая и дружественная среда программирования. Во время работы используются три основных **окна**: *окно редактора кода, окно ввода и окно вывода*.



(рис.2.10.2). Интерфейс программы

Окно редактора кода. Большую часть рабочей области, её верхнюю часть занимает *окно редактора кода*. В него вводится *исходный текст программы*.

Горячие клавиши, которые можно использовать при работе с *текстом программы*:

F2, Ctrl-S - сохранить файл.

F3, Ctrl-O - загрузить файл.

F12 - сохранить файл под новым именем.

Ctrl-Shift-S - сохранить все открытые файлы.

Ctrl-Tab, Ctrl-Shift-Tab - перейти к следующему/предыдущему окну редактора.


Ctrl-Shift-I - увеличить отступ выделенного блока.

Ctrl-Shift-U - уменьшить отступ выделенного блока.

Окно вывода

Под окном редактора расположено окно вывода. Оно предназначено для вывода данных процедурами *write* и *writeln*, а также для вывода сообщений об ошибках и предупреждений во время работы программы.

Окно вывода может быть скрыто. Клавиша *F5* показывает/скрывает окно вывода. Для скрытия окна вывода используется также клавиша **Esc**.

Окно вывода обязательно открывается при любом выводе в него. Для очистки окна вывода следует нажать комбинацию клавиш *Ctrl-Del* или кнопку .

Окно ввода

Окно ввода открывается при выполнении процедур *read* и *readln* в ходе работы программы. Ввод данных в окно ввода сопровождается эхо-выводом в окно вывода. После нажатия клавиши **Enter** данные из окна ввода попадают в соответствующие переменные, окно ввода закрывается, и программа продолжает работать дальше.

Запуск и остановка программы

Для запуска программы в текущем окне редактора следует нажать клавишу **F9** или кнопку  панели инструментов.

Программа вначале компилируется во внутреннее представление, после чего, если не найдены ошибки, программа начинает выполняться. При выполнении программы кнопка запуска программы становится неактивной,

кнопка останова программы, *наоборот, активной* и в строке статуса отображается информация «*Программа выполняется*».

Выполнение программы можно в любой момент прервать нажатием комбинации клавиш **Ctrl-F2** или кнопки . При этом в окне вывода появится сообщение: *Программа прервана пользователем*.

К достоинствам системы *Pascal ABC* следует отнести то, что она позволяет:

- программировать на языке *Pascal*, подобном **Delphi**, включая *объектно-ориентированные расширения*;
- работать с *графикой*;
- создавать *событийные приложения*.

Структура программы Программа на языке **Pascal ABC** имеет следующий вид:

```
Program имя программы;  
{раздел подключения модулей}  
{раздел описаний}  
begin  
{операторы}  
end.
```

- Первая строка называется *заголовком программы* и не является *обязательной*.
- Раздел подключения модулей начинается со служебного слова **uses**, за которым следует *список имен модулей*, перечисляемых *через запятую*.
- Раздел описаний может включать разделы описания *переменных, констант, типов, процедур и функций*, которые следуют друг за другом в произвольном порядке.
- Раздел *подключения модулей* и *раздел описаний* могут отсутствовать.
- Операторы *отделяются* один от другого символом "*точка с запятой*".

Словарь языка «*Pascal*»

Любой естественный язык (*русский, английский и т.д.*) состоит из четырех основных элементов: *символов, слов, словосочетаний и предложений*.

Символ языка – это элементарные знаки, используемые для составления любых текстов. Набор таких символов называют алфавитом языка.

Алфавит языка «Pascal» включает в себя следующие символов:

1. латинские буквы (*заглавные и прописные*).

2. цифры от 0 до 9.

3. знаки арифметических операций:

4. Знаки отношений:

> - больше;

<= - меньше или равно;

>= - больше или равно;

<> - не равно.

< - меньше;

Разделители:

. - для разделения *целой и дробной части числа*, обозначение конца программы, отделение полей в записи;

, - для разделения *элементов* в списке;

;- для разделения *операторов*;

: - для отделения *переменной* или *константы* от типа и метки от оператора;

‘ ‘ – выделение *символа* или *строковой константы*;

() – выделение *выражений*, списков параметров;

[] – выделение *индексов массивов, элементов множеств*;

{ } – *скобки комментариев*.

Операторы вывода - write и writeln выводят заданную информацию на консоль.

Оператор *writeln* после этого еще и переводит курсор на следующую строку. Операторы могут иметь *произвольное число аргументов*, разделенных запятыми. В качестве аргументов могут выступать буквальное константы (*например, строки символов*) и переменные стандартных типов (*в т.ч. и целочисленные*).

Пример: *write('Текстовая строка');* *writeln(a,b);*

Форматы вывода. В операторах *write* и *writeln* можно указывать формат, определяющий число позиций, используемых для вывода целого числа на экран, например: *writeln('Ответ',a:5)*; Это означает, что для вывода значения, хранящегося в переменной *a* следует *зарезервировать 5 символов*, т.е. если число будет не пятизначным, то при выводе оно будет *дополнено слева пробелами*. Если число «не помещается» в формат, он будет автоматически «растянут», так чтобы число «влезло».

Оператор ввода - read, readln. Оператор *read* при чтении данных с консоли используется крайне редко, в основном используется оператор *readln*. *Параметры оператора* – только переменные и только стандартных типов, возможно использование оператора вообще без аргументов для создания ситуации ожидания нажатия клавиши **Enter**. По оператору *readln* программа приостанавливает свою работу, ожидая ввода данных от пользователя. Пользователь набирает желаемые значения переменных, разделяя их пробелами (*или Enter*) и нажимает **Enter**. Оператор интерпретирует введенные символы, переводя их во внутреннее представление *соответствующих переменных*. Если это не удастся (*например, вместо целого числа введены буквы*), программа аварийно завершает свою работу.

Пример: *readln(a,b);*

Недопустимо: readln(a+3);

Для того чтобы пользователь знал, что «хочет» ввести программа, полезно перед оператором *readln* выводить подсказку с помощью оператора *write* или *writeln*.

Например: *write('Введите количество студентов в группе ');*

readln(n);

Обратите внимание на пробел перед вторым апострофом. Он сделан для того, чтобы ввод пользователя *отделялся от подсказки*.

Идентификаторы и служебные слова

Идентификаторы служат в качестве *имен программ, модулей, процедур, функций, типов, переменных и констант*. Идентификатором считается любая

последовательность латинских букв или цифр, начинающаяся с буквы. Буквой считается также символ подчеркивания "_".

Например, $a1$, $_h$, $b123$ - идентификаторы, а $1a$, $\phi 2$ – нет.

Служебные слова служат для оформления конструкций языка и не могут быть использованы в качестве имен.

Идентификатор (или имя) – служит для обозначения различных сущностей в программе. Идентификатор не должен совпадать с ключевыми словами. В языке *Pascal* большие и малые буквы не различаются (в других языках программирования могут различаться).

Переменная – это величина, которая может менять свое значение. Переменная всегда должна быть явно описана как принадлежащая какому-либо типу данных.

Тип данных – множество значений, которые может принимать объект (чаще всего это переменная) данного типа. Кроме множества значений тип данных задает множество допустимых операций.

Например, целочисленный *min integer* определяет диапазон целых чисел от $-2\ 147\ 483\ 648$ до $+2\ 147\ 483\ 647$, в целочисленных выражениях можно применять операции сложения, вычитания и т.п.

Оператор – инструкция языка *Pascal*. Операторы отделяются точкой с запятой друг от друга. Операторы можно объединить в составной оператор, заключив их в операторные скобки *begin ... end*.

Оператор присваивания. Оператор присваивания обозначается двумя символами $:=$, между которыми нет пробела. Слева от знака оператора должна стоять переменная, а справа – выражение. Суть оператора заключается в вычислении выражения и присвоения получившегося значения переменной.

Примеры: $a:=2$; $a:=2+3$; $a:=b*2+1$;

Неправильные примеры: $a+1:=a$; $2:=b+3$;

Очень часто используется оператор вида $a:=a+1$; Он не содержит ошибки. Вначале берется старое значение переменной, к нему прибавляется

единица, и результат записывается в опять в переменную *a*. Таким образом, этот оператор увеличивает значение переменной *a* на единицу.

Существует два способа задать переменной значение: *оператор присваивания* и оператор ввода (*read* или *readln*).

Арифметическое выражение – целочисленное арифметическое выражение состоит из переменных, констант, знаков операций, скобок и вызовов функций. Правила построения выражения очень похожи на те, которые употребляются в математике, за исключением деления, для которого существуют две операции, обозначаемые ключевыми словами *div* и *mod*.

Надо запомнить, что знак операции умножения * никогда не опускается в отличие от алгебры. Операции имеют обычный приоритет: умножение и деление выполняется раньше, чем сложение и вычитание.

Вычисление функций выполняется еще раньше. В остальном операции выполняются слева направо. Для изменения порядка операций используются круглые скобки. Уровень вложенности скобок не ограничен.

Бинарные¹ арифметические операции над типом integer

Операция	Обозначение	Пример
сложение	+	a+3
вычитание	-	c-d
умножение	*	2*3
деление нацело	<i>div</i>	a <i>div</i> 2
остаток от деления	<i>mod</i>	a <i>mod</i> 2

Деление на цело: $5 \text{ div } 2=2$, остаток от деления нацело: $5 \text{ mod } 2=1$.

Кроме бинарных операций в Паскале используются унарные² операции – и + (в последнем особого смысла нет).

Например, в выражении $-2*(-a-b)$ минус перед переменной *a* является унарной операцией, а минус перед двойкой может рассматриваться и как унарная операция и как запись отрицательной константы.

Целочисленные типы языка ABC Pascal

¹ Бинарные операции – имеющие два операнда, располагающиеся слева и справа от оператора

² Унарная операция – имеющая только один операнд.

Тип	Размер в байтах	Диапазон
integer	4	-2147483648 ... 2147483647
byte	1	0 ... 255
word	2	0 ... 65 535

Встроенные функции - в языке Pascal предусмотрено большое число различных встроенных функций. Функции можно применять в выражениях, *аргумент(ы) функции* всегда заключаются в скобки, например: $x := x - \text{abs}(x)$. Функции требуют аргументы определенных типов (*например, целого типа*) и возвращают значение также определенного типа (*может не совпадать с типом аргумента*).

Использование модуля CRT. Язык Pascal имеет специальную библиотеку работы с экраном в *текстовом режиме*. Она называется CRT (*от английского Cathode Ray Tube – электронно-лучевая трубка*). Эта библиотека содержит *набор подпрограмм*, позволяющих задавать цвет выводимых букв, цвет фона, *устанавливать курсор* в желаемую позицию. В текстовом режиме считается, что окно содержит *25 строк по 80 колонок*. В каждой позиции может быть один символ (*буква, цифра, специальные знаки*). Графические объекты (*прямые, окружности и т.п.*) в текстовом окне *не допустимы*. Модуль CRT эмулирует текстовый терминал первых персональных компьютеров. Подключение библиотеки к программе осуществляется предложением **Uses**:

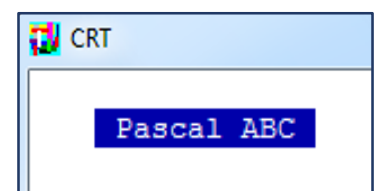
```
Program {имя программы};
```

```
Uses CRT; {Это предложение вставляется сразу после Program}.
```

Далее могут идти описания переменных, затем *begin* и *тело программы*.

Рассмотрим использование *подпрограмм этой библиотеки* на примере *создания заставки программы*.

```
uses CRT; {Подключить модуль CRT}
begin
ClrScr; {Очистить экран}
TextColor(White); {Установить белый цвет
букв}
TextBackGround(Blue); {Установить синий
цвет фона}
```



```
GotoXY(36,13); {Поставить курсор в 36
колонку, 13 строку}
write('Pascal ABC'); {Вывести текст}
ReadKey; {Ожидать нажатия любой клавиши}
end.
```

Наберите текст этой программы и выполните её. Разберем действия отдельных процедур:

ClrScr - очищает экран или текущее окно (*от англ. Clear Screen*), закрашивая его текущим цветом фона, установленным процедурой *TextBackground*. По умолчанию - цвет экрана *белый*.

TextColor(цвет) - устанавливает *текущий цвет букв*, который будет использоваться операторами *write* и *writeln*. По умолчанию используется *светло-серый цвет букв*.

TextBackground(цвет) - устанавливает *текущий цвет фона*, который будет использоваться операторами *write* и *writeln*.

GotoXY(колонка, строка) - переводит курсор в указанную позицию; вывод, осуществляемый последующим оператором *write* или *writeln* будет начинаться с этой позиции. Обычно экран в текстовом режиме содержит *80 колонок и 25 строк*.

ReadKey – ожидание нажатия на любую клавишу.

Рассмотрим еще несколько полезных подпрограмм.

Window(x1, y1, x2, y2) - создание окна вывода. В случае задания окна, все операторы *write* и *writeln* выводят только в него, процедура *ClrScr* будет очищать только это окно.

WhereX, WhereY - две функции, позволяющие узнать координаты местонахождения курсора.

Пример использования: *GotoXY(WhereX+10, WhereY+2)*.



Контрольные вопросы и задания

1. Общие сведения о языке «*Pascal*»
2. Словарь языка «*Pascal*»

3. Типы величин языка «*Pascal*»
4. Числовые величины
5. Логические величины
6. Операторы вывода - *write* и *writeln*
7. Оператор ввода - *read*, *readln*
8. Идентификаторы и служебные слова языка Pascal.
9. Встроенные функции в языке Pascal
10. Использование модуля CRT.

ПРАКТИЧЕСКИЕ РАБОТЫ

11-§. СРЕДА ПРОГРАММИРОВАНИЯ PASCAL ABC

Цель работы: *Научиться* применять основные *операторы (команды)* и редактировать программы с линейной структурой; сформировать навыки работы в среде программирования (*на примере Pascal ABC*); изучить на практике, как составляются программы с линейной структурой.

Краткая теоретическая информация:

Структура программы на Pascal ABC

Основная структура программы

Правила языка *Pascal* предусматривают единую для всех программ форму основной структуры:

Program {Имя программы};

{Раздел описаний}

Begin

{Тело программы}

End.

Здесь слова **Program**, **Begin** и **End** являются *служебными*.

Арифметический оператор присваивания:

{числовая переменная} := {арифметическое выражение}

Чтобы при каждом запуске программы экран был очищен от предыдущих расчетов, необходимо подключить модуль **Crt**, из которого использовать в программе процедуру очистки экрана **Clrscr**. Кроме того, в этом модуле находится функция задержки результатов на экране **Readkey**.

Арифметическое выражение может содержать *числовые константы* и *переменные*, *знаки арифметических операций*, *круглые скобки*. Кроме того, в арифметических выражениях могут присутствовать *функции*.



(рис.2.11.1). Структура программы с линейной структурой

Служебные (зарезервированные) слова языка Pascal:

Absolute	Exports	Library	Set
Assembler	External	Mod	Shl
And	Far	Name	Shr
Array	File	Nil	String
Asm	For	Near	Then
Assembler	Forward	Not	To
Begin	Function	Object	Type
Case	Goto	Of	Unit
Const	If	Or	Until
Constructor	Implementation	Packed	Uses
Destructor	In	Private	Var
Div	Index	Procedure	Virtual
Do	Inherited	Program	While
Downto	Inline	Public	With
Else	Interface	Record	Xor
End	Interrupt	Repeat	Export
Label		Resident	

(табл.2.11.1). Служебные слова языка Pascal

Стандартные функции языка Pascal ABC

В таблице 1 приведены стандартные функции языка Pascal ABC, которые можно использовать в программах.

В математике	В Паскале
$ x $	abs(x)
x^2	sqr(x)
\sqrt{x}	sqrt(x)
e^x	exp(x)
$\ln x$	ln(x)
$\sin x$	sin(x)

cosx	cos(x)
arctgx	arctan(x)
Frac(x)	Дробная часть числа
Int(x)	Целая часть числа

(табл.2.11.2). Стандартные функции языка *Pascal*

Функции преобразования типов

ord (x)	Возвращает порядковый номер аргумента и, таким образом, преобразует величину порядкового типа в величину целого типа.
round (x)	Округляет вещественное число до ближайшего целого.
trunc (x)	Выдает целую часть вещественного числа, отбрасывая дробную.

(табл.2.11.3). Функции преобразования типов

Целочисленные арифметические операции

Целочисленная арифметика языка **Pascal** основана на использовании пяти стандартных целых типов: *Byte*, *Word*, *ShortInt*, *Integer*, *LongInt*.

Со всеми целыми типами связан набор из пяти основных арифметических операций: +, -, *, **div**, **mod**.

$A \bmod B = 0$	Условие делимости целого A на целое B.
$A \bmod 2 = 0$	Условие чётности целого A.
$A \bmod 10$	Значение последней цифры в десятичной записи целого A (например, $1234 \bmod 10 = 4$).
$A \div 10$	Число, десятичная запись которого получится отбрасыванием последней цифры в десятичной записи целого числа A (например, $1234 \div 10 = 123$).
$A \bmod p$	Значение последней цифры в p-ичной записи целого A (например, при $p=2$, $A \bmod 2$ - последняя цифра записи A в двоичной системе счисления);
$A \div p$	Число, p-ичная запись которого получится отбрасыванием последней цифры в p-ичной записи целого числа A (например, $5_{10} = 101_2$; $5_{10} \div 2 = 2_{10} = 10_2$; т.е. из 101_2 получили 10_2);

(табл.2.11.4). Целочисленные арифметические операции

Операции отношения

Выражение	Операция	Результат
A=B	Равно	True, если A= B
A<>B	Не равно	True, если A <>B
A>B	Больше	True, если A >B

$A < B$	Меньше	True, если $A < B$
$A >= B$	Больше или равно	True, если $A >= B$
$A <= B$	Меньше или равно	True, если $A <= B$

(табл.2.11.5). Операции отношения

Правила записи выражений на языке Pascal

1. Выражения записываются в строчку.
2. Допускается применение круглых скобок
3. Не допускается подряд два знака операций
4. Выражения просматривается слева направо, сравниваются две последовательные операции, и выполняется та, чей приоритет выше.
5. Часть выражения, заключенная в скобки, выполняется в первую очередь (т.е. скобки изменяют приоритет).

Примеры записи выражений:

Функция	Запись на языке Pascal
$y = \frac{10ab}{\sqrt{x}(e^{2kx} + 3x^2)} - \log_k(x + 2)$	$y=10*a*b/(sqrt(x)*(exp(2*k*x)+3*sqr(x)))-ln(x+2)/ln(k)$
$Y=x^{12} + \text{sincos}(x+1)+1$	$y=\exp(12*\ln(x))+\sin(\cos(x+1))+1$
$z = \sqrt[3]{x+1} + \text{tg}x + 3,55$	$z=\exp(1/3*\ln(x+1))+\sin(x)/\cos(x)+3.55$

(табл.2.11.6). Примеры

В среде программирования **Pascal ABC** наберите программы и выполните их для данных вариантов.

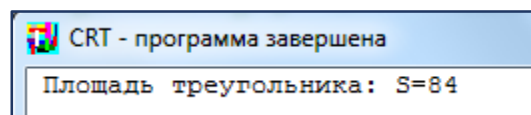
Даны длины сторон треугольника. Вычислить его площадь.

```

Program Geron;
uses crt;
Var a, b, c:integer; p, S:real;
{p - полупериметр, S - площадь
треугольника}
begin
a:=13;
b:=14;
c:=15;
p:=(a+b+c)/2;
s:=sqrt(p*(p-a)*(p-b)*(p-c));
writeln(' Площадь треугольника:
S=', S)

```

Выполнить программу для случаев, если: $a=13$, $b=14$, $c=15$;

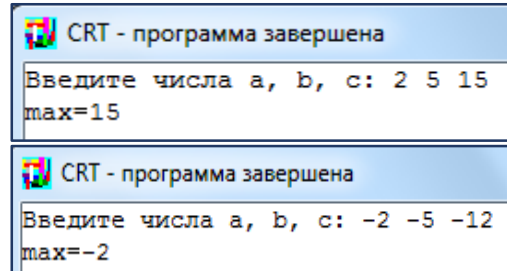


end.

Найти максимальное из трех чисел a, b, c.

```
uses crt;
Var a, b, c, max:real;
begin
write('Введите числа a, b, c:
');
readln(a, b, c);
if a>b then max:=a else max:=b;
if c>max then max:=c;
writeln ('max=', max)
end.
```

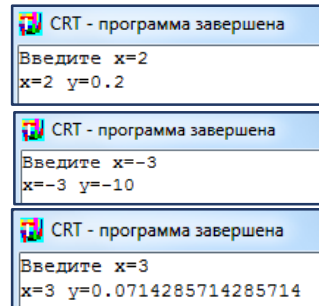
Выполните программу для: a=2, b=5, c=11; a=-1, b=-5, c=-11.



При заданном значении вычислить значение функции $y=x^2+4x-7$.

```
uses crt;
Var x, y:real;
begin
write ('Введите x=');
readln (x);
y:=sqr(x)+4*x-7;
if x>=2 then y:=1/y;
writeln ('x=',x,' y=',y)
end.
```

Выполните программу для случаев, когда $x>2$, $x<2$, $x=2$.



Результаты выполнения задания *покажите преподавателю!*

Задачи для самостоятельного решения

1. Даны координаты диагонали прямоугольника. Найти его площадь.
2. Дан цилиндр с заданным радиусом основания r и высотой h . Найти объем цилиндра.
3. Дано четырехзначное число, найти в нем число тысяч, десятков и единиц.
4. Даны два ненулевых числа. Найти их сумму, разность, произведение и частное.
5. Ввести трехзначное число a . Поменять крайние цифры числа местами.
6. Ввести двузначное число a . Поменять цифры числа местами.

7. Найти площадь кольца по заданным внешнему и внутреннему радиусам.

8. Дана масса в килограммах. Найти число полных центнеров в ней.

9. Дано расстояние в сантиметрах. Найти число полных метров в нем.

10. Составьте программу нахождения значения выражения $(a - (d + 21) * 2) * (c - 7 * k)$, где значения переменных a , d , c и k вводятся с клавиатуры.

11. Даны две целые переменные A , B . Составить фрагмент программы, после исполнения которого, значения переменных поменялись бы местами (новое значение A равно старому значению B и наоборот).

12. Мальчик купил в магазине n порций мороженого по цене 1200 сум и k плиток шоколада по цене 3800 сум. Сколько всего потратил мальчик.

13. Из поселка в город выехал мотоциклист со скоростью 60 км/ч. В то же время ему на встречу выехал велосипедист со скоростью 15 км/ч. Они встретились через 2 часа. Составьте программу, которая выводит расстояние от города до поселка.

14. Задан радиус окружности. Найти площадь и длину окружности.

15. Определить время падения камня на поверхности земли с высоты.



Контрольные вопросы

1. Что такое среда программирования?
2. Опишите среду программирования Pascal ABC.
3. Что означает понятие "зарезервированное слово"?
4. Найдите в следующем списке зарезервированные слова:

X, Program, Y, Summa, MyMoney, Произведение, Feruza, begin, end, if, repeat, Read.

5. Какая структура программы правильна:

a) Program MyProgram;	b) Program MyFirst;
Begin	Begin
WriteLn('Привет!');	X:=Y+100;
End.	End.

6. Для чего используется слово Uses?

7. Как осуществить запуск программы?
8. Как выглядит *оператор вывода*? Для чего он служит? Какое существует различие между выводом сообщения и выводом значения переменной? Можно ли с помощью *оператора вывода* выполнить два действия? Если - да, то какие?
9. Что собой представляет *оператор ввода*? Для чего он служит? Как можно его использовать, чтобы осуществить задержку экрана?
10. Как происходит присваивание переменной какого-либо значения? Как называется этот оператор?
11. Для чего необходим в программе раздел описаний?
12. Какие стандартные типы вам известны?
13. Какие типы относятся к порядковым?
14. В чем различия между данными действительного и целого типов?

12-§. ОПЕРАТОРЫ ЯЗЫКА PASCAL

Цели работы:

- **Формирование** технологии программирования циклических алгоритмов с использованием операторов цикла;
- **Развитие** логического и абстрактного мышления;

Краткая теоретическая информация:

Оператором называется предложение языка программирования, задающее полное описание некоторого действия, которое необходимо выполнить.

Отличием оператора, по отношению к другим элементам, является то, что под ним всегда подразумевается какое-то действие. В языке Pascal операторы состоят из *служебных слов*. Операторы, используемые в программе, отделяются между собой и от других элементов программы

символом (;). Все операторы языка Pascal можно условно разбить на две группы:

1. простые;
2. структурированные.

Простые операторы – это операторы, не содержащие в себе других операторов. К ним относятся:

- ✚ оператор присвоения (`:=`);
- ✚ оператор процедуры;
- ✚ оператор безусловного перехода (*goto*).

Структурированные операторы – это операторы, которые содержат в себе другие операторы. К ним относятся:

- ✚ составной оператор;
- ✚ операторы условий (*if, case*);
- ✚ операторы цикла (*for, while, repeat*);
- ✚ оператор присоединения (*with*).

Простые операторы.

Оператор процедуры

Оператор процедуры служит для вызова процедуры.

Формат: [имя_процедуры] (список параметров вызова);

Оператор процедуры состоит из идентификатора процедуры, непосредственно за которым в скобках расположен список параметров вызова. В Паскале имеются процедуры без параметров. В этом случае, при вызове, список параметров отсутствует. Выполнение оператора процедуры приводит к активизации действий, описанных в ее теле. В языке Pascal имеется два вида процедур:

- Стандартные, которые описаны в самом языке и являются принадлежностью языка;
- Процедуры пользователя, которые создает пользователь.

Для вызова стандартных процедур необходимо подключение в разделе *uses* имени модуля (*библиотеки*), где описана данная процедура. Ряд процедур, находящихся в модуле *system* всегда подключается к программе автоматически и их подключение в разделе *uses* не нужно. Стандартные процедуры языка Pascal - *read, write, rewrite, close, reset*.

read (*[файловая_переменная]*, *[список_ввода]*)
read (*x, y*)

Процедуры пользователя (*нестандартные*) должны быть созданы перед их использованием в программе и находятся либо в разделе описания самой программы, либо в отдельные программные единицы модуля. Если процедура находится в модуле, то имя этого модуля необходимо упомянуть в приложении *uses*.

Оператор безусловного перехода GOTO

Формат: *goto* [*метка*];

GOTO – зарезервированное слово в языке Pascal. [*метка* - это произвольный идентификатор, который позволяет пометить некий оператор программы и в дальнейшем сослаться на него. В языке Pascal допускается в качестве меток использовать целое число без знаков. Метка располагается перед помеченным оператором и отделяется от него (:). Один оператор можно помечать несколькими метками. Они так же отделяются друг от друга (:). Перед тем как использовать метку в разделе оператора ее необходимо описать в разделе *label* (*раздел описания*).

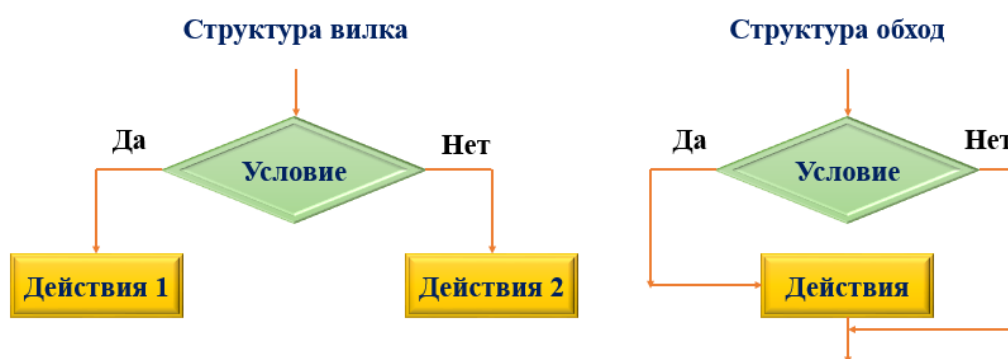
Действие *goto* передает управление соответствующему помеченному оператору. При использовании меток нужно руководствоваться следующими правилами:

- метка должна быть описана в разделе описаний и все метки должны быть использованы;
- если в качестве меток используются целые числа, их не объявляют.

Структурированные операторы

Операторы условия If

Условный оператор используется в программе для реализации алгоритмической структуры - *ветвления*. В данной структуре вычислительный процесс может продолжаться по одному из возможных направлений. Выбор направления обычно осуществляется проверкой какого-либо условия. Существует два вида структуры ветвления: структура *вилка* и *обход*.



(рис.2.12.1). Структура вилка и обход.

В языке Pascal условный оператор **if** это средство организации ветвящегося вычислительного процесса.

Формат: IF [логическое_выражение] then [оператор_1]; else [оператор_2];

IF, then, else – служебные слова. [оператор 1], [оператор 2] – обыкновенные операции языка Pascal. Часть *Else* является необязательной (может отсутствовать).

Если часть *else* отсутствует, то оператор *if* имеет не полную форму:

if [логическое_выражение] then [оператор];

В этом случае, если результат *Истина (true)*, то выполняется [оператор], если *Ложь (false)*, то управление передается оператору, следующему за оператором *if*.

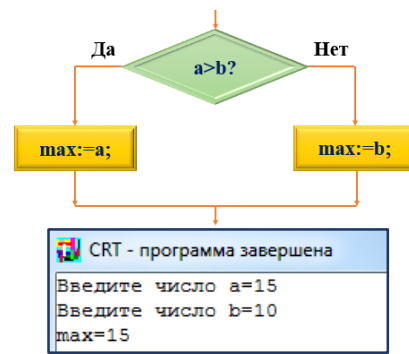
Пример:

Имеются 2 числа А и В. Найти максимальное число.

```

uses crt;
var a, b, max: integer;
begin
write ('Введите число a=');
readln(a);
write ('Введите число b=');
readln(b);
If a>b then max:=a else
max:=b;
Writeln ('max=',max);
end.

```



Составной оператор

Составной оператор – это последовательность произвольных операций в программе, заключенная в так называемые операторные скобки (*begin-end*).

Формат: *begin [операторы]; end;*

Составные операторы позволяют представлять группу операторов, как один оператор.

Оператор выбора CASE

Предназначен для реализации множественных ветвлений, поскольку оператор **IF** может реализовать всего два направления вычислительного процесса, использовать его для реализации множественных ветвлений не всегда удобно. Множественное ветвление реализуется оператором **CASE**.

Формат:

```

CASE [ключ выбора] OF
  [константа_выбора_1]: [оператор_1];
  [константа_выбора_2]: [оператор_2];
  ...
  [константа_выбора_N]: [оператор_N];
ELSE [оператор];
End;

```

CASE, OF, ELSE, END – служебные слова. *[ключ_выбора]* – это параметр одного из порядковых типов. *[константы_выбора]* – константы того же типа, что и ключ выбора, реализующие выбор. *[оператор_1(N)]* – обыкновенный оператор. *ELSE* может отсутствовать.

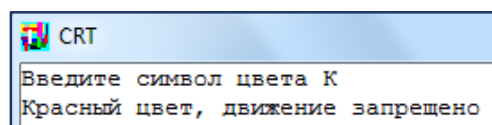
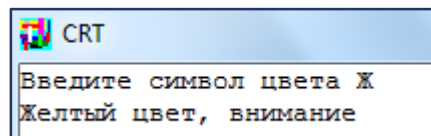
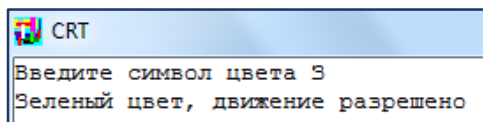
В операторе *CASE* нет явной проверки условия, характерного для оператора **IF**. В тоже время в неявном виде операция сравнения выполняется.

CASE вносит диссонанс в программу на языке Pascal, поскольку данные оператор завершается служебным словом End, которому нет парного Begin.

Пример:

Составить алгоритм и программу задачи, моделирующей работу светофора. При вводе символа первой буквы цветов светофора, программа должна выводить сообщение о соответствующем цвете и действиях.

```
uses crt;
var cvet:char;
begin
write ('Введите символ цвета ');
readln(cvet);
case cvet of
'З': writeln ('Зеленый цвет, движение разрешено');
'Ж':      writeln ('Желтый      цвет,
внимание');
'К': writeln ('Красный цвет,
движение запрещено');
else      writeln ('Светофор      не
работает');
end;
readln
end.
```



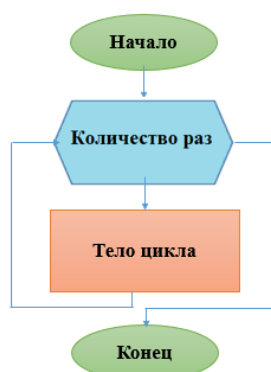
Операторы цикла

Циклической алгоритмической структурой считается такая структура, в которой некоторые действия выполняются несколько раз.

Оператор FOR

Цикл со счетчиком или, как он еще называется - цикл с параметром в Паскале служит для организации повторений с заранее известным их количеством, т.е. когда точно известно, сколько раз должны выполняться операторы тела цикла.

Блок-схема, соответствующая циклу For:

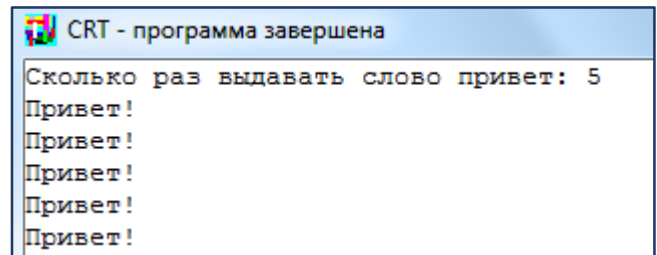


(рис.2.12.2). Блок-схема. Цикл for

Пример:

Вывести пять раз слово «Привет!»

```
uses crt;  
var i, a: integer;  
begin  
write('Сколько раз выдавать  
слово привет: ');  
read(a);  
for i:=1 to a do  
writeln('Привет!');  
end.
```



Прямое направление цикла (от меньшего значения к большему):

При движении от меньшего значения к большему, например, от 1 до 10

for счетчик:=значение **to** конечное_значение **do** тело_цикла;

Обратное направление (от большего к меньшему):

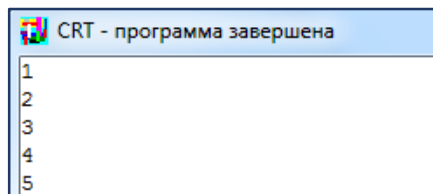
При движении от большего значения к меньшему, например, от 10 до 1

for счетчик:=значение **downto** конечное_значение **do** тело_цикла;

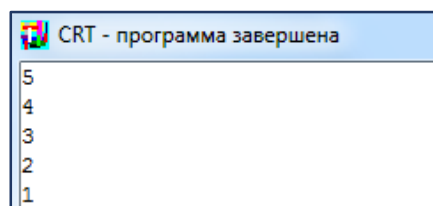
Пример:

Вывод чисел по убыванию и возрастанию

```
uses crt;  
var i:integer;  
begin  
for i:=1 to 5 do  
writeln(i);  
end.
```

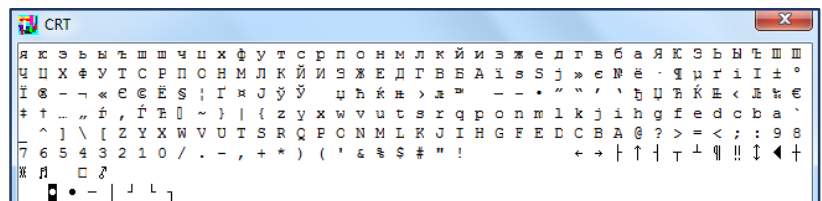


```
uses crt;  
var i:integer;  
begin  
for i:=5 downto 1  
do  
writeln(i);  
end.
```



Вывести на экран символы в обратном порядке.

```
uses crt;  
var i:integer;  
begin  
for i:=255  
downto 0 do  
write(chr(i), '  
' );  
readln  
end.
```



Операторы цикла while и repeat

Используются тогда, когда заранее не известно общее количество итераций (*повторений вычислений*) цикла, а завершение вычислений зависит от некоего условия. Если условие ставится вначале цикла (*на входе*), то используется оператор **While**, если на выходе – **Repeat**.

А теперь расшифруем сказанное: сначала *while*, а потом ниже *repeat*.

Оператор цикла **while** имеет такую структуру:

while условие do

оператор

Это надо понимать так: пока истинно условие, стоящее между **while** и **do**, выполняется оператор после **do**, называемый телом цикла. Перед каждым заходом в цикл проверяется условие: если оно истинно, то выполняется оператор, если ложно, то автоматически осуществляется выход из цикла.

Если оператор в цикле состоит из нескольких операторов, то поместить их нужно в операторные скобки *begin - end* (*сравните оператор цикла for*). Не стоит также забывать, что сами операторы разделяются оператором "точка с запятой" (*но перед закрывающим END в конце ставить её не обязательно*).

while условие do

begin

оператор 1;

оператор 2;

оператор 3;

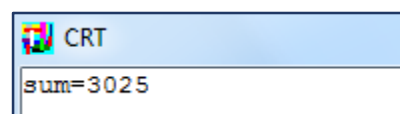
...

оператор N

end;

Вычислите сумму кубов целых чисел от 1 до 10

```
uses crt;  
var b, sum:integer;  
begin  
  b:=1;  
  sum:=0; //начальная сумма  
  while b<=10 do begin  
    sum:=sum+b*b*b;  
    inc(b) end;
```



```
writeln ('sum=', sum) ; //ВЫВОДИМ  
результат  
readln  
end.
```

Цикл с предусловием в Паскале – WHILE

Оператор цикла с предусловием выполняет действия заранее неизвестное число раз. Выход из цикла осуществляется, если некоторое логическое выражение или его результат окажется ложным. Так как верность логического выражения проверяется в начале, тело цикла может не выполниться ни одного разу.

Структура цикла с предусловием:

WHILE <условие> DO

begin

<тело цикла>

end;

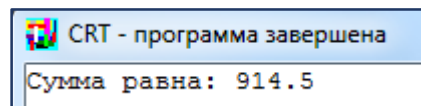
Порядок выполнения цикла:

Пока условие истинно, выполняется тело цикла. Как только условие становится ложно, выполнение цикла прекращается.

Пример:

Задача: вычислить сумму ряда $1+1.5+2+2.5+3+3.5+ \dots + 30$.

```
uses crt;  
var sum:real; n:real;  
begin  
sum:=0; n:=1;  
while n <= 30 do  
begin  
sum:=sum+n; n:=n+0.5;  
end;  
writeln ('Сумма равна: ', sum) ;  
end.
```



Цикл с постусловием - Repeat - until.

Этот оператор аналогичен оператору цикла с предусловием, но отличается от него тем, что проверка условия производится после выполнения тела (*действий*) цикла. Это обеспечивает его выполнение хотя бы один раз в отличие от ранее разобранных циклов.

Следует обратить внимание на то, что данный оператор цикла предполагает наличие нескольких операторов в теле цикла, то есть можно

выполнять несколько действий, поэтому служебные слова `Begin` и `End` не нужны.

Последовательность операторов, входящих в тело цикла, выполняется один раз, после чего проверяется соблюдение условия, записанного следом за служебным словом `Until`. Если условие не соблюдается, цикл завершается. В противном случае - тело цикла повторяется ещё раз, после чего снова проверяется соблюдение условия.

Формат записи, структура цикла:

REPEAT

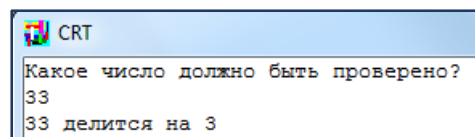
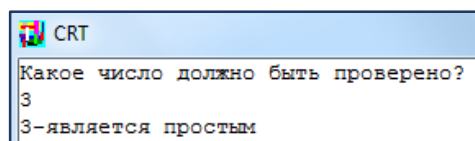
<тело цикла>

UNTIL <условие>;

Пример:

Определить, является ли введенное число простым.

```
uses crt;  
var i, {возможный делитель}  
Number:integer;  
{исследуемое число}  
begin  
  ClrScr;  
  writeln('Какое число должно быть  
  проверено? ');  
  read(Number);  
  i:= 1;  
  repeat  
    i:= i+1;  
  until Number mod i = 0;  
  if Number=i then  
    writeln (Number,'-является  
    простым') else  
    writeln (Number,' делится на ',i);  
  readln;  
end.
```



Контрольные вопросы

1. Основные элементы программирования.
2. Основные характеристики программы.
3. Операторы языка программирования Паскаль. Оператор присваивания. Формат, примеры.

4. Оператор безусловного перехода. Формат, примеры. Основные правила использования
5. Условные операторы. Оператор *if*. Формат, описание. Основные правила использования. Особенности использования вложенного оператора *if*.
6. Примеры использования оператора *if*.
7. Оператор выбора *case*. Формат, описание.
8. Правила использования оператора *case*. Примеры использования.
9. Оператор повтора *while*. Формат записи, описание работы цикла.
10. Пример использования оператора *while*. Описание работы программы.

13-§. ГРАФИЧЕСКИЕ ВОЗМОЖНОСТИ ЯЗЫКА PASCAL ABC

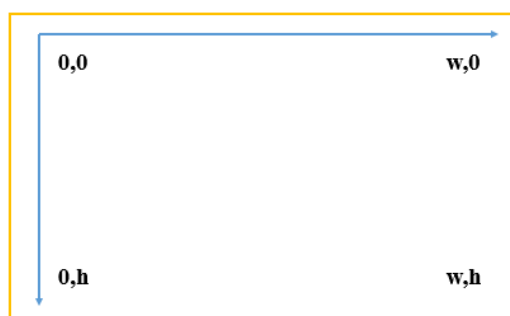
Цели работы: *Ввести понятия: «графический редактор», «координатная плоскость». Изучить графическое правило, графические процедуры для построения геометрических фигур. Формировать умения и навыки инициализации экрана для вывода графических объектов. Выработать навыки составления программ, реализующих вывод графических объектов на экран.*

Краткая теоретическая информация:

После запуска *Pascal ABC*, по умолчанию, запускается текстовый режим. Для работы с графикой служит отдельное графическое окно. Чтобы его открыть, необходимо подключить модуль *GraphABC*. В этом модуле содержится набор процедур и функций, предназначенных для работы с графическим экраном, а также некоторые встроенные константы и переменные, которые могут быть использованы в программах с графикой.

С их помощью можно создавать разнообразные графические изображения и сопровождать их текстовыми надписями. Подключение осуществляется в разделе описаний.

Изображение на экране монитора формируется из точек (*пикселей*) определенного цвета. Положение каждой точки задается ее координатами. В качестве экранных координат используют порядковые номера пикселей по горизонтали и вертикали. Они могут принимать только целочисленные значения. Началом отсчета является левый верхний угол экрана. Значения координаты x (*абсциссы*) отсчитывается слева направо, а значения y (*ординаты*) - сверху вниз. В среде *PascalABC* рисование осуществляется в графическом окне. Начало системы координат находится в левом верхнем углу окна, ось абсцисс направлена вправо, а ось ординат – вниз. (рис.1).



(рис.2.13.1). Экранные координаты

Координатная плоскость экрана

Началом координат $(0,0)$ считается левый верхний угол экрана. Координаты x (*номера столбцов или позиций*) возрастают справа налево, а Модуль *GraphABC* содержит константы, типы, процедуры, функции и классы для рисования в графическом окне.

ClearWindow - очищает графическое окно белым цветом.

SetWindowWidth (w: integer) - устанавливает ширину графического окна.

SetWindowHeight (h: integer) - устанавливает высоту графического окна.

CloseWindow - закрывает графическое окно.

Процедуры для построения геометрических объектов.

SetPixel (x, y, color: integer) - закрашивает один пиксел с координатами (x, y) цветом *color*.

Line (x1, y1, x2, y2: integer) - рисует отрезок с началом в точке $(x1, y1)$ и концом в точке $(x2, y2)$.

Rectangle(x1, y1, x2, y2: integer) - рисует прямоугольник, заданный координатами противоположных вершин (x1, y1) и (x2, y2).

Circle(x,y,r) – окружность радиусом r , с центром в точке (x, y)

Установочные процедуры.

SetPenColor (color: integer) устанавливает цвет пера, задаваемый параметром color.

SetPenWidth (w: integer) - устанавливает ширину пера, равную w пикселям.

SetPenStyle (ps: integer) - устанавливает стиль пера, задаваемый параметром ps.

Стили пера задаются константами:

SetBrushColor(color) - устанавливает цвет кисти. (*заливка кистью распространяется на замкнутый контур*).

SetBrushStyle(bs: integer) - устанавливает стиль кисти, задаваемый параметром bs.

FloodFill (x,y,color: integer) - заливает область одного цвета цветом color, начиная с точки (x, y).

FillRect (x1,y1,x2,y2: integer) - заливает прямоугольник.

TextOut(x, y, 'строка'); - выводит строку текста в позицию (x, y). Точка (x, y) задает верхний левый угол прямоугольника, который будет содержать текст.

Модуль *GraphABC* содержит константы и функции для работы с цветами.

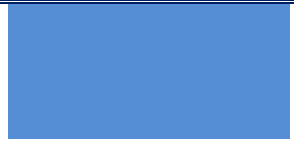
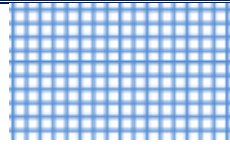




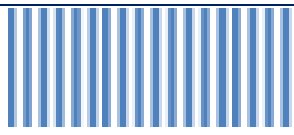

Стандартные цвета

Значение	Имя	Виды цветов. Назначение
0	clBlack	Черный
1	clBlue	Синий
2	clGreen	Зеленый
3	clSkyBlue	Голубой
4	clRed	Красный
5	clPurple	Фиолетовый
6	clBrown	Коричневый
7	clMedGray	Серый
8	clDkGray	Темно-серый
9	clNavy	Темно-синий

10	clLime	Ярко-зеленый
11	clAqua	Бирюзовый
12	clFuchsia	Сиреневый
13	clMaroon	Темно-красный
14	clYellow	Желтый
15	clWhite	Белый

(табл.2.13.1). Стандартные цвета

Стили кисти

bsSolid		bsCross	
bsClear		bsDiagCross	
bsHorizontal		bsBDiagonal	
bsVertical		bsFDiagonal	

(табл.2.13.2). Стили кисти

Режимы пера задаются константами:

1. **pmCopy** – обычный режим; при рисовании цвет поверхности заменяется цветом пера.

2. **pmNot** – режим инвертирования; при рисовании цвет поверхности становится негативным, а цвет пера при этом игнорируется.

Пример:

Составить программу, которая рисует на экране графические примитивы: отрезок, прямоугольник, круг, эллипс.

```

program grafika;
uses graphABC; {подключение
модуля graphABC}
begin
setWindowSize(600,240);
//размеры окна
line(80,40,500,40); //отрезок
setPenWidth(5); //толщина пера
setPenColor(clBrown);
setBrushColor(clYellow);
//цвет пера и кисти

```



```

rectangle(40,80,200,160);
//прямоугольник
setPenColor(clRed);
setBrushColor(clAqua);
circle(300,120,40); //круг
setPenColor(clBlue);
setBrushColor(clRed);
ellipse(400,80,540,160);
//эллипс setFontSize(15);
setFontStyle(fsBold); //размер
и начертание шрифта
setFontColor(clBrown);
setBrushColor(clWhite);
textOut(120,180,'Графические
примитивы'); //вывод текста
end.

```

Пример:

Создать рисунок (домик) используя заготовку чертежа.

```
uses GraphABC;
```

```
begin
```

```
line(100,200,200,100);
```

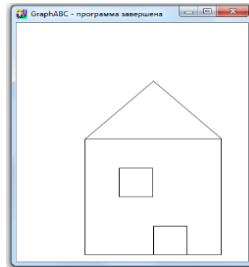
```
line(200,100,300,200);
```

```
rectangle(100,200,300,400);
```

```
rectangle(150,250,200,300);
```

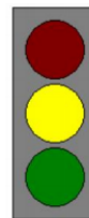
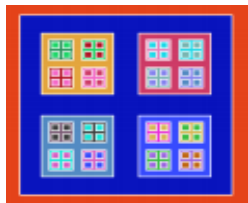
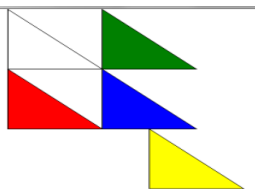
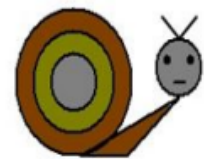
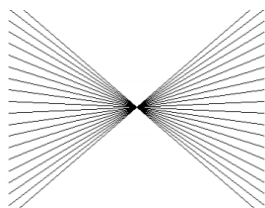
```
rectangle(200,350,250,400);
```

```
end.
```



Задания к практической работе

1. Составьте программу, которая выводит на экран одно из приведенных изображений. Для закрашивания можно использовать различные стили линий



2. Составить программу, выводящую на экран геометрические фигуры: квадрат, прямоугольник, треугольник.

3. Создайте программы, используя следующие наборы команд:

- a. SetBrushColor(clLime); circle (300,350,150)
- b. SetPenColor (clBlack); rectangle(100,100,200,200);
- c. SetBrushColor(clNavy); Line (90,90,150,150);



Контрольные вопросы

1. В чём отличия *графических координат* разрешения экрана монитора и *значений координатной плоскости*?
2. Как переключить программу **Pascal ABC** из *символьного режима* в *графический*?
3. С помощью каких процедур можно получить высвечивание точки на экране, *рисования линий, прямоугольников, окружностей*?
4. Посмотрите *образец программы*, которая рисует «дом». Скажите, пожалуйста, из каких разделов состоит *структура данной программы*?
5. Какая процедура позволяет *закрашивать фигуры*?
6. Какие цвета вы использовали? (*Назовите на английском языке*)
7. Количество параметров в *процедуре Line*?
8. Сколько параметров имеет *процедура Circle*?

ЛАБОРАТОРНЫЕ РАБОТЫ

14-§. ВВОД И ОТЛАДКА ЛИНЕЙНОЙ ПРОГРАММЫ

Цель работы: Приобрести практические навыки в составлении, отладке линейных программ и в работе со средой программирования.

Краткая теоретическая информация:

Линейной называется программа, все операторы которой выполняются последовательно, в том порядке, в котором они записаны. Это самый простой вид программ.

Операторы предназначены для описания действий, которые будут выполнены при реализации алгоритма. В Паскале операторы разделяются на две группы: *простые* и *структурные*.

Операторы отделяются друг от друга в программе символом «;». Точка с запятой не является частью оператора, это **разделитель** операторов.

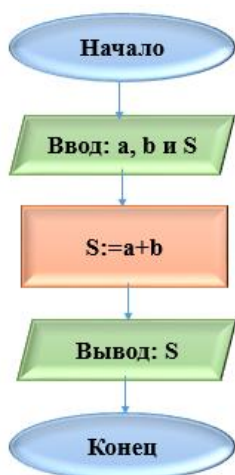
Несмотря на то, что создание программы происходит в несколько этапов, наиболее важным и трудоемким является этап отладки и тестирования программы. Именно на этом этапе устраняются все логические и синтетические ошибки в создаваемой программе.

Отладка – это деятельность, направленная на обнаружение и исправление ошибок в программе.

Пример 1

Составить блок-схему и программу для вычисления суммы *a* и *b*.

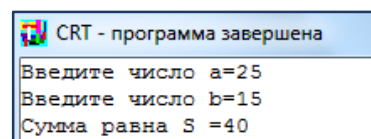
Блок-схема



Программа

```
uses Crt;  
var a, b, S: Integer;  
begin  
  Clrscr;  
  write('Введите число  
a=');  
  Readln(a);  
  write('Введите  
число b=');  
  readln(b);  
  S:=a + b;  
  writeln('Сумма равна  
S =',S);  
  readkey  
end.
```

Результат



Описание программы:

1. *Program* - служебное слово начала программы
2. *Summa* - имя программы (набирать на латинском регистре)

3. *Uses* - подключение дополнительных *модулей* или *библиотек* к программе

4. *Crt* – очистка экрана от *предыдущих расчетов*

5. *Var* - начало раздела *описания переменных*

6. *a, b, S: integer;* - выделение в оперативной памяти места для размещения переменных *a, b, S* - *целого типа*

7. *Begin* - начало операторной части *программы*

8. *Clrscr;* - процедура *очистки экрана*

9. *Write('введи число a=');* - вывод на экран *комментария*

10. *Readln(a);* - ввод значения переменной *a* с клавиатуры в оперативную память

11. *Write('введи число b=');* - вывод на экран *комментария*

12. *Readln(b);* - ввод значения переменной *b* с клавиатуры в оперативную память

13. *S:=a+b;* - вычисление значения *S* и занесения его в оперативную память

14. *Writeln('Сумма равна S=',S);* - вывод на экран *комментария S=* и значение *S*

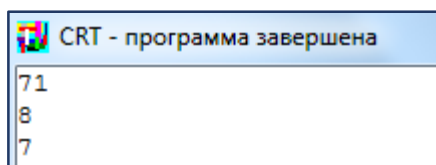
15. *Readkey;* - задержка *расчетов*

16. *End.* - конец *программы*

Пример 2

Вывести на экран сумму и произведение цифр двузначного числа.

```
uses crt;  
var ab, b, s, p, a:integer;  
begin  
  readln(ab); //ввод двузначного  
  числа  
  a:=ab div 10; //вывод первой  
  цифры числа  
  b:=ab mod 10; //вывод второй  
  цифры числа  
  s:=a+b; //сумма цифр числа  
  p:=a*b; //произведение цифр  
  числа  
  writeln(s); //вывод суммы  
  writeln(p); //вывод  
  произведения  
end.
```

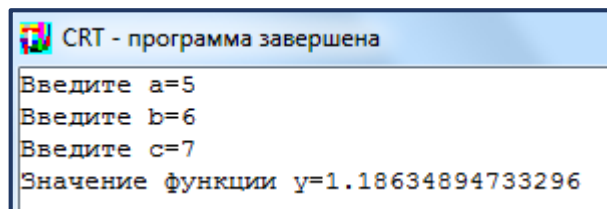


Пример 3

Вычислить значение функции $y = \arctan \sqrt{\frac{ax^2+b}{c}} + \ln \frac{ax^2+b}{c} + d$; при $x=1,5$; $d=2,5$.

Примечание: Очевидно, что функцию Y целесообразно вычислять в такой последовательности: предварительно введя исходные данные a, b, c и присвоив значение переменных x и d , вначале найдем значение выражения, которое обозначим переменной z , и далее определим выражение Y . Используя общепринятые символы блоков, изобразим схему разрабатываемого алгоритма и составим программу при значении аргументов $x=1.5, d=2.5$ и произвольных значениях переменных a, b, c .

```
uses crt;  
var a, b, c:integer; x, d, y,  
z:real;  
begin  
write('Введите a=');readln(a);  
write('Введите b=');readln(b);  
write('Введите c=');readln(c);  
x:=1.5; d:=2.5;  
z:=(a*x*x+b)/c;  
y:=arctan(sqrt(z)+ln(z));  
writeln('Значение функции  
y=',y);  
end.
```



Задачи для самостоятельного решения

1. Найти периметр и площадь прямоугольного треугольника, если даны длины его катетов a и b .
2. Вывести на экран три введенных с клавиатуры числа в порядке, обратном их вводу.
3. Дано натуральное число меньше 256. Сформировать число, представляющее собой его квадрат.
4. Сформировать число, представляющее собой реверсную (обратную в порядке следования разрядов) запись заданного трехзначного числа. Например, для числа 341 таким будет 143.
5. Обменять значения двух переменных, используя третью (буферную) переменную.
6. Дан радиус шара. Найти его объем.
7. Вывести на экран действительное число в различных форматах представления.
8. Дана длина ребра куба a . Найти объем куба $V=a^3$ и площадь его поверхности $S=6a^2$.

9. Составить структурную схему алгоритма и проект программы вычисления функции, один параметр ввести с клавиатуры, а другой задать как константу, все вычисляемые значения вывести на экран.

№	Функция	Значения параметров	№	Функция	Значения параметров
	$y = \sin^4(a^2 + b^2)$ $a = \sqrt{b + t};$ $t = b^2 + k^3$	b=6 k=3,4		$y = \cos^3 x + a $ $x = e^b;$ $b = a + \sqrt{a + p^2}$	a=5,5 p=4
	$y = \lg^3 p + x$ $p = x(a + b^3);$ $x = e^{2a} + \sqrt{b}$	a=6 b=4,3		$y = \ln^5 x + a^2 $ $x = r^2 \sqrt{ a + b };$ $a = t + b^3$	t=3 b=4,2
	$y = \sqrt[3]{x^2 + c^2}$ $x = e^{mk};$ $c = \cos^2 m + k^2$	k=1 m=1,8		$y = \cos x^5 - b \sin^2 x$ $x = a^3 + \sqrt{t + b};$ $a = t^2 b$	t=6,2 b=1,8



Контрольные вопросы и задания

1. Почему при *записи формул* на Паскале их "вытягивают" в линию?
2. Почему на Паскале аргумент функции всегда записывают в скобках?

(Например, $\ln(5)$, а не $\ln 5$)

3. Укажите символы, буквы, составные символы:

^, Y, <, +, *, R, k, \$, !, ф.

4. Что в списке можно рассматривать как идентификаторы:

ФИО, ФИО, 33719, X, Y, >=, \$, &, Summa, _Rezult

5. Какие идентификаторы удобнее использовать и почему:

klass1, Klass_1, summadohoda, SummaDohoda, nomerdoma, Nomer_Doma

6. Найдите ошибки в программах:

a) **Program**
Ошибки1;
begin
summa:=6+8;
end;

b) **Program**
Ошибки2;
begin
var
X:integer;
X:=6+8;
write(X);
end;

7. С чего начинается программа? С чего начинается основная программа? Чем заканчивается программа?

8. Как описываются переменные?

9. Что такое процедура и чем она отличается от процедуры?

15-§. ВВОД И ОТЛАДКА ПРОГРАММ С ВЕТВЛЕНИЕМ

Цель работы: Закрепить знания о *разветвляющихся алгоритмах* и умения применять их при решении поставленных задач.

Краткая теоретическая информация:

Форма организации действий, при которой в зависимости от выполнения некоторого условия совершается одна или другая последовательность действий, называется ветвлением.

Алгоритм называется *разветвляющимся*, если порядок выполнения шагов алгоритма изменяется в зависимости от заданных условий

Оператор выбора case

Оператор *case* позволяет сделать выбор между несколькими вариантами:

```
case <переключатель> of  
    <список_констант>:<один_оператор>;  
    [<список_констант>:<один_оператор>;]  
    [<список_констант>:<один_оператор>;]  
    [else <один_оператор>;]  
end;
```

Замечание: Обратите внимание, что после *else* двоеточие не ставится.

Существуют дополнительные правила, относящиеся к структуре этого оператора:

1. *Переключатель* должен относиться только к *порядковому типу* данных, но не к типу *longint*.

2. *Переключатель* может быть переменной или выражением.
3. Список *констант* может задаваться как явным перечислением, так и интервалом или их объединением.
4. Повторение *констант* не допускается.
5. *Тип переключателя* и типы всех констант должны быть совместимыми.

Выполнение оператора *case* происходит следующим образом:

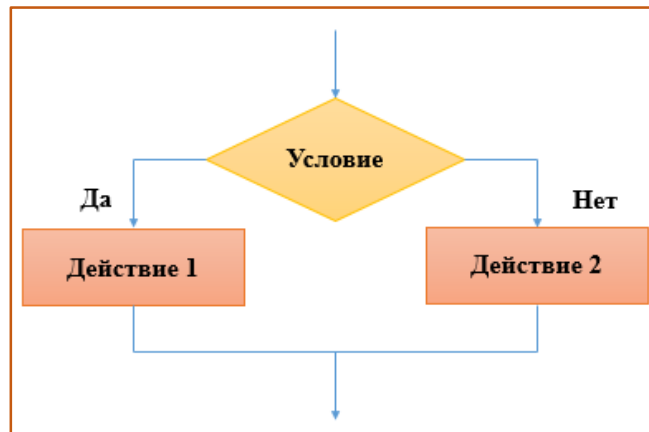
1. вычисляется значение переключателя;
2. полученный результат проверяется на принадлежность к тому или иному списку констант;
3. если такой список найден, то дальнейшие проверки уже не производятся, а выполняется оператор, соответствующий выбранной ветви, после чего управление передается оператору, следующему за ключевым словом *end*, которое закрывает всю конструкцию *case*.
4. если подходящего списка констант нет, то выполняется оператор, стоящий за *ключевым словом else*. Если *else-ветви* нет, то не выполняется ничего.

Отладку программы выполняют для поиска логических ошибок в программе.

Отладка осуществляется с помощью *пошагового выполнения программы* и *просмотра* текущих значений переменных.

Полная команда ветвления

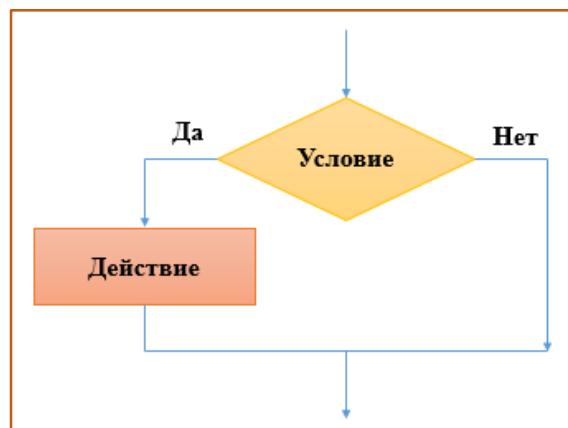
if <i>условие</i>	then <i>действие 1</i>	else <i>действие 2</i>
если	то	иначе



(рис.2.15.1). Полная форма ветвления

Неполная команда ветвления

if условие **then** действие
если **то**

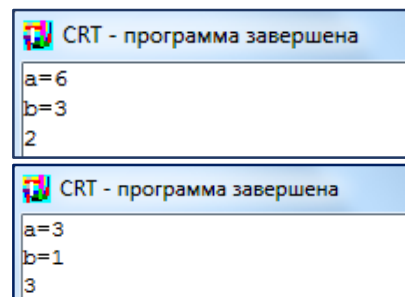


(рис.2.15.2). Неполная форма ветвления

Пример 1

Ввести два числа. Если сумма этих чисел четная, найти произведение, в противном случае, найти частное этих чисел.

```
uses crt;
var a,b:integer;
begin
write('a=');
readln(a);
write('b=');
readln(b);
if (a+b) mod 2=0 then
writeln(a*b)
else
writeln(a/b)
end.
```



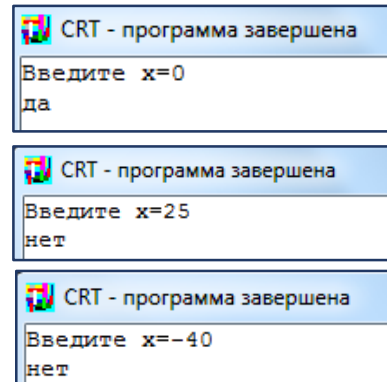
Пример 2

Проверить, принадлежит ли число, введенное с клавиатуры, интервалу (-5;3).


```

uses crt;
var a,b,x:integer;
begin clrscr;
  write('Введите x=');
  readln(x);
  if (x>=-5) and (x<=3) then
    writeln('да') else
    writeln('нет');
  end.

```



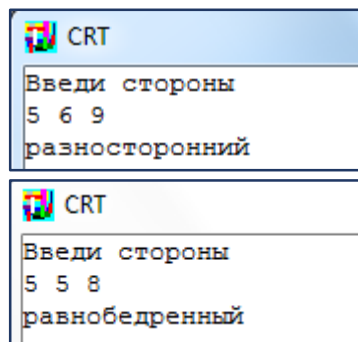
Пример 3

Известны стороны треугольника, определить равны ли его стороны *равнобедренный* или *разносторонний*.

```

uses crt;
var a,b,c:integer;
begin clrscr;
  writeln('Введи стороны ');
  readln(a,b,c);
  if (a=b) or (a=c) or (c=b) then
    writeln('равнобедренный') else
    writeln('разносторонний');
  readln
  end.

```



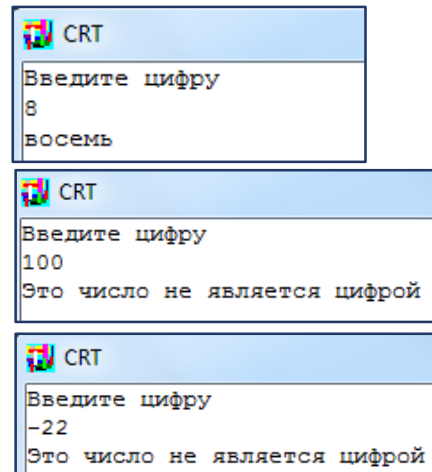
Пример 4

Написать программу преобразования цифр в слова.

```

uses crt;
var a, b, c:integer;
begin
  writeln('Введите цифру');
  readln(a);
  if (a<0) or (a>9)
    then writeln ('Это число не является цифрой')
  else
    case a of
      0 : writeln ('ноль');
      1 : writeln ('один');
      2 : writeln ('два');
      3 : writeln ('три');
      4 : writeln ('четыре');
      5 : writeln ('пять');
      6 : writeln ('шесть');
      7 : writeln ('семь');
      8 : writeln ('восемь');
      9 : writeln ('девять');
    end;
  readln
end.

```



Задачи для самостоятельного решения

1. Найти наибольшее из трех данных чисел.
2. Запишите условный оператор, в котором значение переменной вычисляется по формуле: $a + b$, если a - нечетное и $a*b$, если a - четное.
3. Вывести на экран номер четверти, которой принадлежит точка с координатами (x, y) , при условии, что x и y отличны от 0.
4. Если целое число m делится нацело на целое число n , то вывести на экран частное от деления, в противном случае - сообщение « m на n нацело не делится».
5. По номеру дня недели вывести его название.
6. В зависимости от того введена ли открытая скобка или закрытая, напечатать "открытая круглая скобка" или "закрытая фигурная скобка". (Учитывать круглые, квадратные, фигурные скобки).
7. В зависимости от введенного символа L, S, V программа должна вычислять длину окружности; площадь круга; объем цилиндра.
8. По введенному числу от 0 до 15 вывести название цвета, соответствующего этому коду.
9. Определить, является ли введенная буква русского алфавита гласной.
10. Написать программу, которая бы предлагала меню для вывода графических объектов, и рисовала бы соответствующий выбору.
11. Напишите программу, которая по введенному номеру месяца високосного или невисокосного года, выводит количество дней в месяце.



Контрольные вопросы и задания

1. Укажите общий вид условного оператора
2. Можно ли в условном операторе использовать более одного оператора?
3. Как запишется условный оператор для поиска большего из двух чисел?

4. Какая фигура в *блок-схеме* используется для записи проверки условия?
5. Для чего предназначен *оператор выбора*?
6. К какому виду алгоритма относится *условный оператор*?
7. В чём разница между *условным* оператором и оператором *варианта*?
8. *Сколько операторов* можно написать после возможного значения переменной?
9. Может ли сразу несколько констант *фигурировать* в качестве возможного значения переменной?

16-§. СОСТАВЛЕНИЕ ПРОГРАММ С ИСПОЛЬЗОВАНИЕМ ЦИКЛОВ

Цель работы: *Формировать навыки составления программ с циклической структурой, используя инструкции for, repeat, while;*

Краткая теоретическая информация:

Цикл – *это последовательность операторов, которая может выполняться более одного раза.*

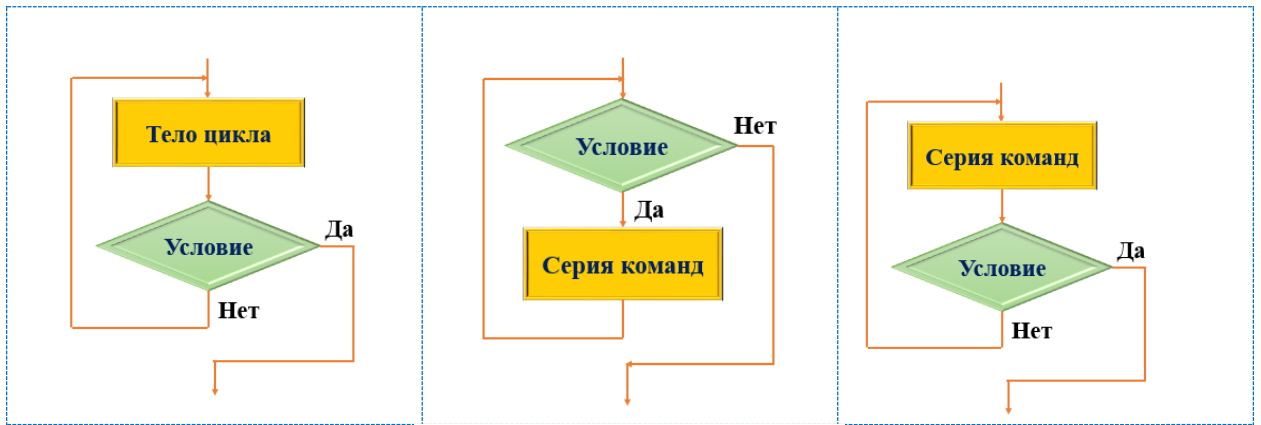
В языке *Pascal*, как и в большинстве языков программирования, существует три типа циклических конструкций (*практическая работа №4*).

Любой цикл состоит из *тела* и *заголовка*.

Тело цикла - это набор повторяющихся операторов, а *условие* - это логическое выражение, в зависимости от результата которого и происходит *повторное выполнение цикла*.

Схема выполнения операторов имеет вид:

Цикл for	Цикл while	Цикл repeat



(рис.2.16.1). Схема операторов

While, или цикл с предусловием

While - это цикл, в котором условие стоит *перед телом*. Причем тело цикла выполняется тогда и только тогда, когда условие *true*; как только условие становится *false*, выполнение цикла прекращается.

Данный цикл подходит только для одного оператора, если же вы хотите использовать несколько операторов в своем коде, вам следует заключить их в операторные скобки - *begin* и *end*.

Repeat, или цикл с постусловием.

Repeat - полная противоположность *while*. Repeat - это цикл, в котором условие стоит после тела. Причем оно выполняется тогда и только тогда, когда результат условия *false* как только логическое выражение становится *true*, выполнение цикла прекращается.

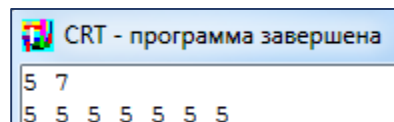
For, или цикл с параметром.

For - это цикл, в котором тело выполняется *заданное количество раз*.

Пример 1

Даны целые числа **K** и **N** ($N > 0$). Вывести **N** раз число **K**.

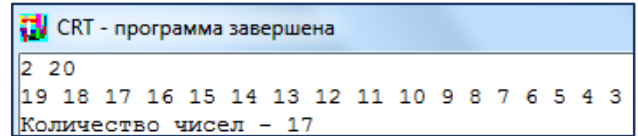
```
uses crt;
var K, N, i: integer;
begin read(K, N);
  for i := 1 to N do
    write(K, ' ');
    {Пишем K через пробел}
  end.
```



Пример 2

Даны два целых числа A и B ($A < B$). Вывести в порядке убывания все целые числа, расположенные между A и B (не включая числа A и B), а также количество N этих чисел.

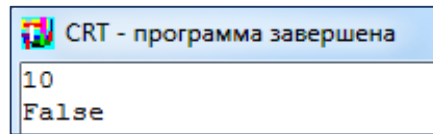
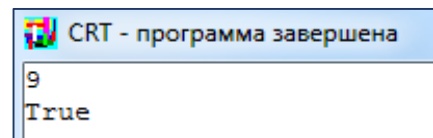
```
uses crt;
var A, B, i, count:integer;
begin read(A, B);
for i := B - 1 downto A + 1 do
write(i, ' ');
{выписываем числа от большего
к меньшему}
count:=(B - 1) - (A + 1) + 1;
{считаем количество чисел}
writeln;
write('Количество чисел - ',
count);
end.
```



Пример 3

Дано целое число N ($N > 0$). Если оно является степенью числа 3, то вывести True, если не является - вывести False.

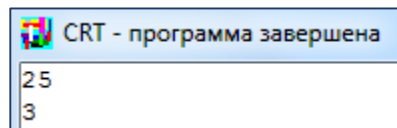
```
uses crt;
var N: integer;
begin
readln(N);
while N mod 3 = 0 do N := N div
3;
{Пока остаток от деления на три
равен нулю, делим N нацело}
writeln(N = 1);
{логическое выражение}
end.
```



Пример 4

Дано натуральное число N . Определить K - количество делителей этого числа.

```
uses crt;
var n, d:longint;
k:integer;
begin
readln(n);
if n=1 then k:=1 else
k:=2; d:=2;
repeat
if n mod d=0 then
inc(k);
d:=d+1;
until d> n div 2;
writeln(k);
end.
```



Тесты к заданию:

№	Ввод	Вывод
	45	6
	121	3
	19	2

Задачи для самостоятельного решения

1. Составьте программу вычисления суммы:
 - a. *четных чисел*;
 - b. *нечетных чисел* в диапазоне от **1** до **100**
2. Как на паскале написать программу для отображения четных и нечетных чисел от **1** до **10**
3. Найти сумму четных цифр числа.
4. Найти произведение четных чисел от **30** до **50**.
5. Выведите (*через пробел*) все четные числа от **a** до **b** (*включительно*).
6. Найти сумму четных чисел от **1** до **n**.
7. Заданы числа **x**, **y**, **z**. Найти сумму четных чисел и произведение нечетных.
8. Найти сумму четных чисел натурального ряда, начиная с числа **n** до **m**.
9. Даны два целых числа **a** и **b** ($a < b$). Вывести в порядке возрастания все целые числа, расположенные между **a** и **b** (*включая сами числа a и b*), а также количество **n** этих чисел.
10. Даны два целых числа **a** и **b** ($a < b$). Найти сумму всех целых чисел от **a** до **b** включительно.
11. Даны два целых числа **a** и **b** ($a < b$). Найти произведение всех целых чисел от **a** до **b** включительно.
12. Даны два целых числа **a** и **b** ($a < b$). Найти сумму квадратов всех целых чисел от **a** до **b** включительно.
13. Дано целое число **n** ($n > 0$). Найти сумму $1 + 1/2 + 1/3 + \dots + 1/n$ (*вещественное число*).
14. Дано целое число **n** ($n > 0$). Найти сумму $n^2 + (n+1)^2 + (n+2)^2 + \dots + (2n)^2$ (*целое число*).
15. Дано целое число **n** ($n > 0$). Найти произведение $1.1 \cdot 1.2 \cdot 1.3 \dots (n \text{ сомножителей})$.

1. Дано целое число n ($n > 0$). Найти значение выражения $1.1 - 1.2 + 1.3 - \dots$ (n слагаемых, знаки чередуются). Условный оператор не использовать.



Контрольные вопросы и задания

1. Что называется, циклом?
2. К какому типу данных должен быть отнесен счетчик цикла *for*?
3. Какие виды циклов существуют в *Pascal*?
4. Дайте сравнительную характеристику циклов *for - to* и *for - downto*.
5. Что представляет собой цикл *while*?
6. Как осуществляется накапливание суммы или произведения?
7. Как вычисляются математические функции?
8. Как определяется *четность* и *нечетность* элементов?
9. Вычислить сумму всех четных чисел до 50.

ГЛАВА 3. ИСПОЛЬЗОВАНИЕ ПРИКЛАДНЫХ ПРОГРАММ

МОДУЛЬ 3

17-§. РЕДАКТОРЫ ЭЛЕКТРОННЫХ ТАБЛИЦ. ИХ ВОЗМОЖНОСТИ

Ключевые слова: *электронные таблицы, табличный процессор, столбец, строка, ячейка, диапазон ячеек, лист, книга, пользовательский интерфейс, лента, формулы, содержимое ячеек, числовые значения, формулы, прогрессия, форматирование данных.*

Современные *технологии обработки информации* часто приводят к тому, что возникает необходимость *представления данных в виде таблиц*. Для табличных расчетов характерны относительно *простые формулы*, по которым производятся вычисления, и *большие объемы исходных данных*. Такого рода расчеты принято относить к разряду *рутинных работ*, для их выполнения следует использовать *компьютер*.

Электронные таблицы (или табличные процессоры) – это прикладные программы, предназначенные для проведения табличных расчетов.

Основное свойство ЭТ – мгновенный пересчет формул при изменении значений, входящих в них операндов. Благодаря *этому свойству*, таблица представляет собой *удобный инструмент* для организации *численного эксперимента*:

1. подбор параметров;
2. прогноз поведения *моделируемой системы*;
3. анализ зависимостей;
4. планирование.

Дополнительные удобства для моделирования дает возможность графического представления данных (*диаграммы*); Использование электронной таблицы в качестве *базы данных*. Конечно, по сравнению с СУБД электронные таблицы имеют меньшие возможности в этой области. Однако

некоторые операции манипулирования данными, свойственные реляционным СУБД, в них реализованы. Это *поиск информации* по заданным условиям и *сортировка информации*.

Области применения электронных таблиц

1. *бухгалтерский и банковский учет;*
2. *планирование распределение ресурсов;*
3. *проектно-сметные работы;*
4. *инженерно-технические расчеты;*
5. *обработка больших массивов информации;*
6. *исследование динамических процессов.*

Основные возможности электронных таблиц

1. *анализ и моделирование* на основе выполнения вычислений и обработки данных;
2. *оформление таблиц, отчетов;*
3. *форматирование* содержащихся в таблице данных;
4. *построение диаграмм* требуемого вида;
5. *создание и ведение баз данных* с возможностью выбора записей по заданному критерию и сортировки по любому параметру;
6. *перенесение (вставка)* в таблицу информации из документов, созданных в других приложениях, работающих в среде *Windows*;
7. *печать итогового документа целиком или частично;*
8. *работа в Internet (поиск данных и публикация информации)* с помощью инструментария ЭТ.

Основные элементы электронных таблиц:

- | | |
|-------------------------------|------------------------------|
| 1. <i>Столбец,</i> | 4. <i>Заголовки строк,</i> |
| 2. <i>Заголовки столбцов,</i> | 5. <i>Неактивная ячейка,</i> |
| 3. <i>Строка,</i> | 6. <i>Активная ячейка.</i> |

Популярные табличные процессоры:

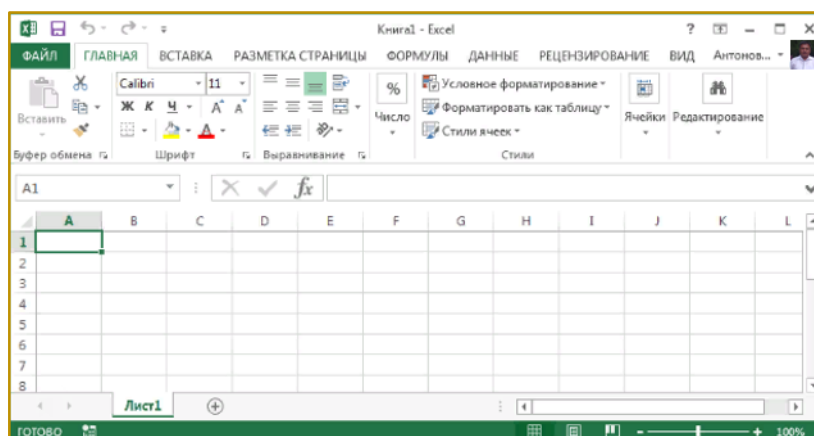
- | | |
|---------------------------|-----------------------|
| • <i>Microsoft Excel;</i> | • <i>Abacus;</i> |
| • <i>SuperCalc;</i> | • <i>Lotus 1-2-3;</i> |

- *OpenOffice.org Calc*;
- *Gnumeric*;
- *KSpread*;
- *Quattro Pro*.

Популярность табличных процессоров росла очень быстро. Появлялись новые программные продукты этого класса: *Multiplan*, *Quattro Pro*, *SuperCalc* и другие. Одним из самых популярных табличных процессоров сегодня является *Microsoft Excel*, входящий в состав пакета *Microsoft Office*.

Слово «*Excel*» в переводе с английского означает «*выдающийся, превосходящий другие*». И это, действительно, мощная программа *обработки табличных данных*. С ее помощью можно выполнять сложные расчеты - *экономические, инженерные* и так далее, создавать *математические модели*, строить *графики и диаграммы*, создавать *тестовые программы*.

Рабочее окно программы Microsoft Excel:



(рис.3.17.1). Рабочее окно программы Ms. Excel

Интерфейс программы

В строке заголовка отображается название программы (*Microsoft Excel*) и имя документа. *Документ Excel* называется Рабочей книгой. При сохранении файлу по умолчанию присваивается имя **Книга1**, расширение - **xlsx**.

Главное меню по умолчанию расположено в верхней части окна программы *под строкой заголовка*. С помощью программ главного меню можно получить доступ ко всем инструментам программы. Главное меню Excel содержит следующие пункты: *Файл, Главная, Вставка, Разметка страницы, Формулы, Данные, Рецензирование, Вид*.

Как и любое другое приложение, Excel имеет основную рабочую область называемой *пользовательский интерфейс*.

Пользовательский интерфейс - это сочетание экранов, меню и значков, которые используются для взаимодействия с приложением.

В программе Excel, *пользовательский интерфейс* в основном состоит из *ленты меню* и *рабочей тетради*.

Лента является наименованием, указанным в ряду *вкладок* и *кнопок*, которые находится в верхней части Excel. *Вкладки* и *кнопки* на ленте показывают основные команды в открытом доступе, сгруппированные в определенных категориях. Лента состоит из пяти основных компонентов: *панель быстрого доступа*, *вкладки*, *группы*, *командные кнопки* и *диалоговые окна*.

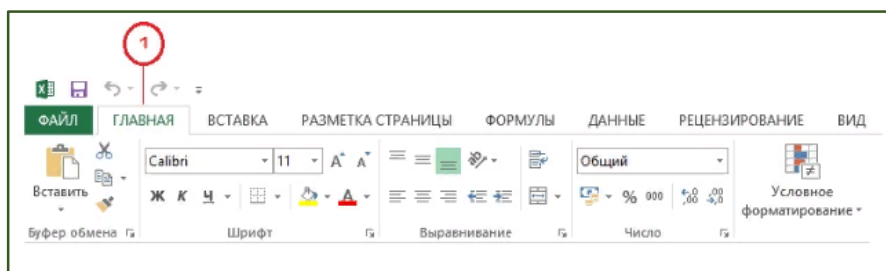
- **Панель быстрого доступа**, это настраиваемые панели инструментов, к которым можно добавить команды, которые используются наиболее часто.

- **Вкладки** содержат группы команд, связанные с основными задачами.

- **Группы** содержат наборы команд, которые подпадают под основные задачи этой вкладки. Каждая группа содержит кнопки, которые можно нажимать, чтобы активировать команду, необходимую к использованию.

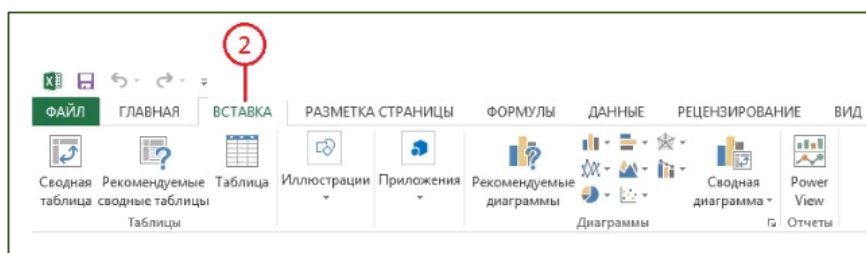
- **Диалоговые окна** активируются, нажав на маленькую стрелку, расположенную в правом нижнем углу некоторых групп. Нажатие на любую стрелку активирует диалоговое окно, содержащее все команды, доступные для данной группы.

Вкладка **«Главная»** содержит команды для общих действий, таких как *форматирование*, *копирование*, *вставка* и *удаление столбцов и строк*.



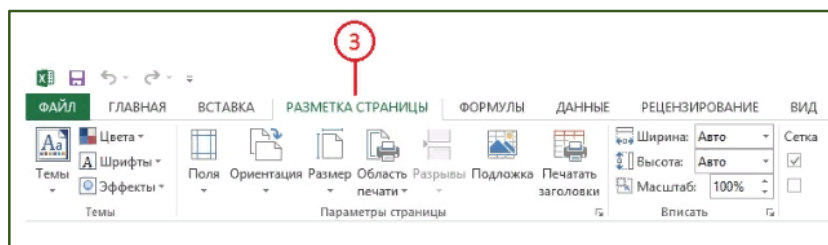
(рис.3.17.2). Вкладка «Главная»

Вкладка «**Вставка**» содержит команды, позволяющие *вставить* объекты, такие как *графики и формы* в таблицы.



(рис.3.17.3). Вкладка «Вставка»

Вкладка «**Разметка страницы**» содержит все команды, которые позволяют определить, как будет выглядеть ваша электронная таблица, как на экране, так и при печати. Эти команды управляют такими параметрами, как тематические цвета, полей страницы и области печати.



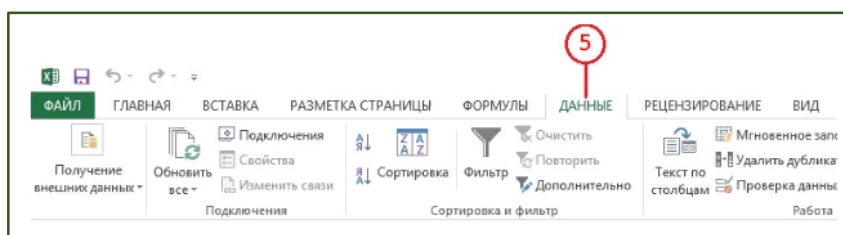
(рис.3.17.4). Вкладка «Разметка страницы»

Вкладка «**Формулы**». Эта вкладка содержит все команды, которые помогают определить, *контроль и аудит* формул Excel.



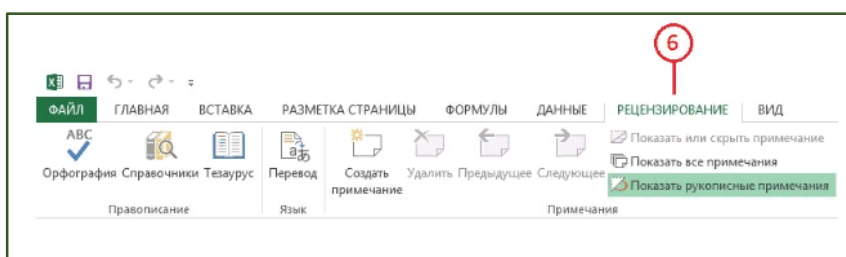
(рис.3.17.5). Вкладка «Формулы»

Вкладка «**Данные**». Эта вкладка имеет команды, позволяющие *подключаться к внешним данным*, а также *управлять данными* в электронной таблице.



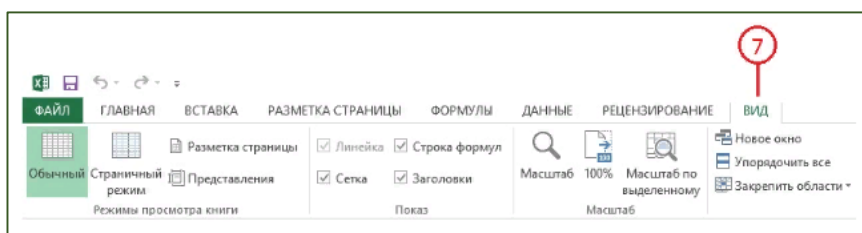
(рис.3.17.6). Вкладка «Данные»

Вкладка «**Рецензирование**». С такими командами, как *проверить орфографию*, *защитить лист*, *защитить рабочую книгу*, и *отслеживание изменений*, вкладка «Рецензирование» защищает *целостность данных* в электронной таблице.



(рис.3.17.7). Вкладка «Рецензирование»

Вкладка «**Вид**». Команды на этой вкладке, помогут контролировать то, как пользователь *визуально взаимодействует* с таблицей.



(рис.3.17.8). Вкладка «Вид»

Вкладка «**Файл**» выставляет представление, где находятся команды, которые помогают *открыть существующие книги Excel*, *создавать новые рабочие книги*, *сохранять книги*, и многое другое.

Кроме того, многие вкладки «*ленты*» являются *контекстно-зависимыми*, то есть появляются на экране только при выделении определённого объекта или установке на него курсора. Например, при

добавлении диаграммы в таблицу становятся доступными вкладки «Конструктор», «Макет» и «Формат», объединённые общим заголовком «Работа с диаграммами».

Любой лист представлен в виде сетки, состоящей из *строк (rows)* и *столбцов (columns)*. Таблица содержит 1048576 (2^{20}) строк и 16384 (2^{14}) столбцов. Каждому столбцу соответствует *буква (A, B, C... до XFD)*, а каждой *строке – цифра*. Пересечение строки со столбцом называется *ячейкой (cell)*, причем каждая из них имеет собственное имя, иногда называемое адресом ячейки. Например, на пересечении столбца *A* и строки *1* находится ячейка **A1**. Именно в активную ячейку происходит ввод *текста, чисел, формул* или другой информации.

В Excel существует также альтернативная адресация ячеек (*стиль ссылок*) – **R1C1**, при которой столбцы и строки листа имеют числовые имена. Адрес ячейки **A1** в таком стиле будет иметь вид *R1C1, B1 – R1C2, C4 – R4C3* и т.д. Переключение между двумя типами ссылок осуществляется в разделе «Формулы» диалогового окна «*Параметры*», находящегося во вкладке «Файл».

В нижней части книги находятся ярлыки, дающие доступ к различным листам. *Ms. Excel* позволяет присваивать листам имена, добавлять новые, а также удалять пустые и устаревшие листы. Каждая книга имеет полосы прокрутки, с помощью которых можно перемещаться между листами, а также по пространству активного листа.

Помимо «*мыши*» для перемещения по листу можно использовать ряд клавиатурных комбинаций (*табл. 3.17.1*). Если при использовании *ползунков прокрутки* изменяется видимая часть экрана, то перемещение с помощью клавиатурных комбинаций приводит к изменению положения активной ячейки.

Клавиатурные комбинации, предназначенные для перемещения по листу

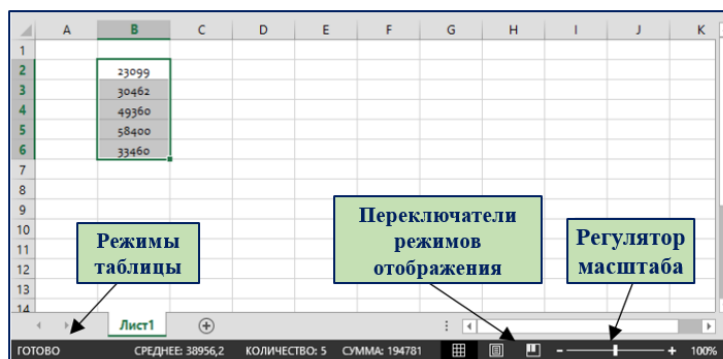
Комбинация клавиш	Значение
Стрелки вверх, вниз, вправо, влево.	На соседнюю ячейку в направлении стрелки
Ctrl+стрелка вверх, Ctrl+стрелка вниз, Ctrl+стрелка вправо, Ctrl+стрелка влево.	На соседнюю ячейку, содержащую данные в направлении стрелки
Home	В столбец А текущей строки
Page Up	На один экран выше
Page Down	На один экран ниже
Alt+Page Up	На один экран влево
Alt+Page Down	На один экран вправо
Ctrl+Home	В ячейку А1
Ctrl+End	В ячейку последнего столбца и последней строки, содержащую данные
Ctrl+пробел	Изменяет видимую часть листа, перемещая вас к активной ячейке или к выделенному набору ячеек, которые были скрыты из виду

(табл.3.17.1). Клавиатурные комбинации

В нижней части таблицы находится строка состояния. В её левом углу отображается режим, в котором сейчас находится таблица:

- ✓ «Ввод» – пользователь не закончил вводить информацию,
- ✓ «Готово» – ввод завершён.

Если в таблице выделен диапазон с числовыми данными, то в строке состояния отображается *среднее значение выделенных чисел*, их количество и сумма. В правой части находятся кнопки переключения режимов отображения таблицы, а также регулятор, позволяющий динамически изменять масштаб отображения.



(рис.3.17.9). Строка состояния

Содержимое ячеек

Microsoft Excel позволяет вводить информацию разного типа: *текстовые данные, числовые значения, а также формулы*. Текстовое значение может представлять собой любую комбинацию *прописных и строчных букв, чисел и символов*. Текст выравнивается по *левой границе* ячейки. Если соседние ячейки не содержат никакой информации, появляется возможность ввести длинный текст, тем самым вызвав перекрывание ячеек. Если же находящаяся справа ячейка содержит какое-то значение, показываемый текст обрезается. При этом полную версию текстового значения можно посмотреть в *строке формул*, выделив ячейку. Ячейки, содержащие текстовые данные, не могут использоваться в *вычислениях*.

Числовые значения могут быть целыми (5), десятичными дробями (555,5), Целыми дробями или числами в экспоненциальном представлении (4.09E+13). Если вы вводите слишком большое число, которое не помещается в ячейку, программа автоматически изменяет её ширину или переводит число в экспоненциальное представление. Ряд значков # в ячейке означает, что для просмотра числа нужно вручную увеличить её ширину. Действительный вид введённого числа можно всегда посмотреть в *строке формул*. По умолчанию числовые значения выровнены *по правому краю* ячейки. Ячейки, содержащие числовые данные, могут *использоваться в вычислениях*.

Формулы представляют собой выражения, по которым выполняются вычисления на странице. Формулы могут содержать *числа, математические операторы, ссылки на другие ячейки и встроенные уравнения*, называемые функциями.

Все формулы в *Ms. Excel* начинаются со знака равенства (=). При этом в ячейках таблицы можно видеть только результат вычислений формулы. Сама формула отображается в *строке формул*. При построении формул используются стандартные и специальные математические операторы.

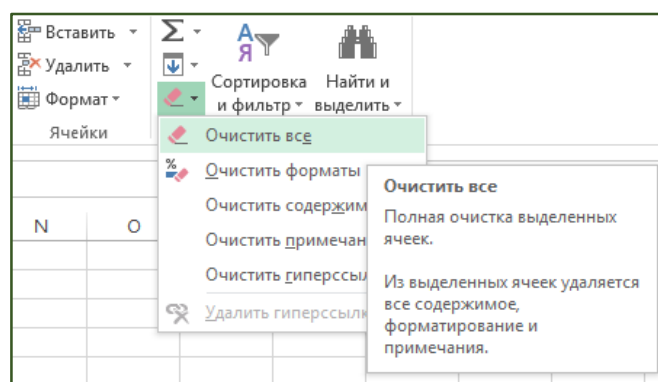
Оператор	Название	Пример	Результат
()	Скобки	(1+2)*3	9
^	Возведение в степень	2^2	4

*	Умножение	2*3	6
/	Деление	9/3	3
+	Сложение	9+3	12
-	Вычитание	9-3	6

(табл.3.17.2). Основные арифметические операторы

Редактирование данных.

Операции выделения, копирования, вставки аналогичны операциям в текстовом редакторе. Однако жёсткая табличная структура листа вносит как ограничения, так и дополнительные возможности.

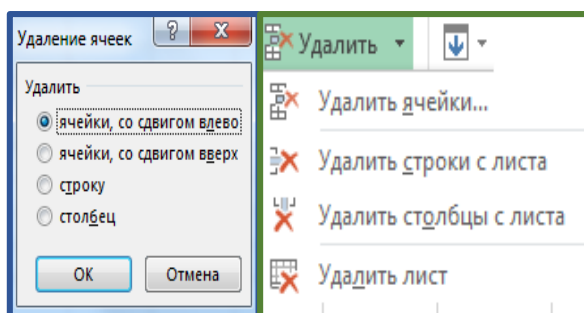


(рис.3.17.10). Команда «Очистить»

Выделение строк и столбцов осуществляется путём нажатия на их заголовки. Удерживая при этом кнопку мыши и перемещая её указатель, можно выделить несколько столбцов или строк. Однако если номера строк или буквы столбцов идут не по порядку, это значит, что лист содержит скрытые строки или столбцы. В этом случае операции со строками или столбцами выделенного диапазона затронут и эти скрытые данные. Для снятия выделения достаточно щёлкнуть мышью по любой ячейке рабочего листа.

Для удаления данных из ячеек следует (после выделения нужного диапазона) выбрать в контекстном меню команду «Очистить содержимое». При этом форматирование ячеек сохранится. Для просмотра всех возможных способов очистки ячеек выберите команду «Очистить» в меню «Редактирование» на главной вкладке.

Для удаления самих ячеек (а не данных в них) используется команда «Удалить» в меню «Ячейки» (либо контекстное меню на листе). При этом необходимо выбрать направление смещения ячеек, граничащих с удаляемой областью.

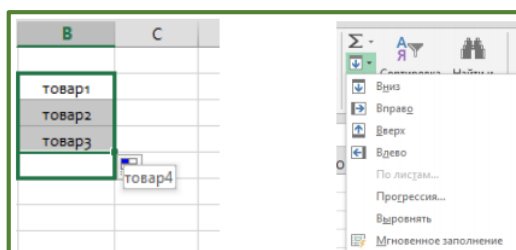


(рис.3.17.11). Удаление ячеек на рабочем листе

Microsoft Excel облегчает задачу ввода в ячейки повторяющихся или изменяющихся стандартным образом значений, называемых прогрессиями, что помогает сохранить время при вводе групп слов, чисел или дат. Например, можно скопировать одну и ту же цену продукта в несколько ячеек. Также *Ms. Excel* может самостоятельно продолжать прогрессию, основываясь на установленном образце.

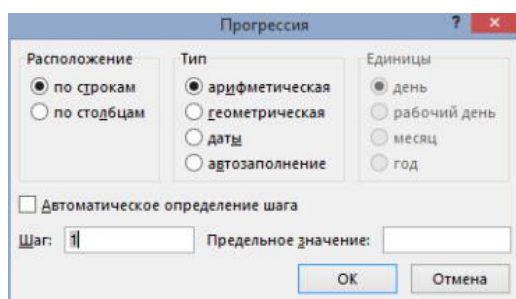
Для создания прогрессии можно использовать маркер автозаполнения – *маленький черный квадратик*, расположенный в правом нижнем углу выделенной ячейки. То, что указатель мыши примет форму жирного черного креста, означает включение функции автозаполнения. Для создания прогрессии текста, чисел или дат нужно выделить несколько ячеек, чтобы определить образец, и, щелкнув на маркере авто-заполнения, перетащить указатель мыши, выделяя ячейки, которые нужно заполнить данными.

Заполнение ячеек данными происходит в соответствии с рядом правил. При перетаскивании маркера автозаполнения вниз или вправо значения выделяемых ячеек увеличиваются в соответствии с образцом. А при перетаскивании маркера вверх или влево – значения уменьшаются. Для получения доступа к этим и сходным командам можно воспользоваться дополнительным меню, появляющимся при выборе команды «*Заполнить*» меню «*Редактирование*». Это меню пригодится, например, при копировании содержимого одной ячейки в несколько ячеек, расположенных рядом.



(рис.3.17.12). Маркер автозаполнения и команда «Заполнить»

При необходимости определения прогрессии чисел, увеличивающихся на некоторую величину, или указания максимального значения последовательности чисел, используйте команду «Прогрессия» (команда «Заполнить» меню «Редактирование»). Перед этим нужно ввести в ячейку исходное значение и выделить интервал заполнения.

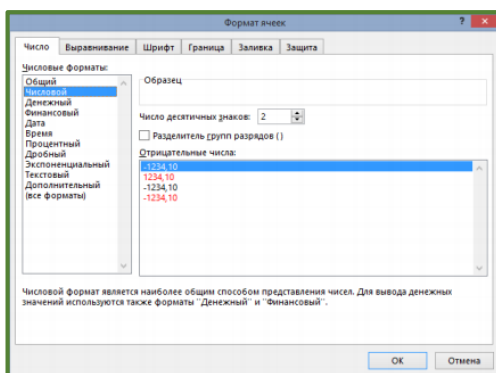


(рис.3.17.13). Диалоговое окно «Прогрессия»

Для заполнения блока ячеек одинаковыми числами необходимо выделить заданный диапазон ячеек, ввести с клавиатуры нужное число и нажать сочетание клавиш **Ctrl+Enter**.

Форматирование данных

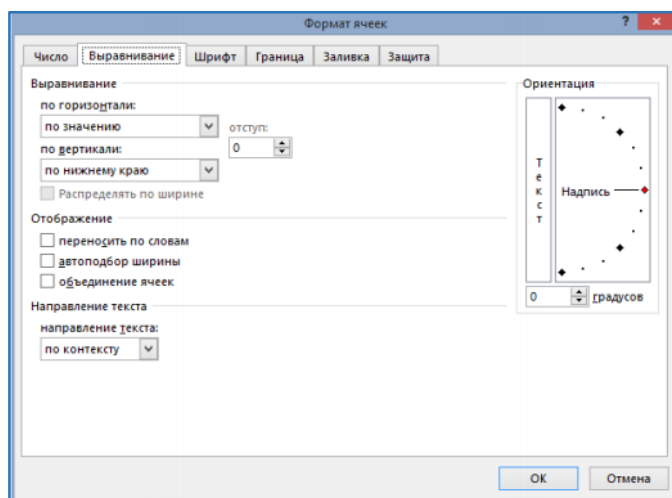
Любое число, вводимое в программу Ms Excel, отображается в определенном формате. Все возможные форматы чисел сгруппированы в меню «Формат ячеек» на вкладке «Число». Если при вводе числовых данных они отображаются «неправильным» образом, следует активировать данное диалоговое окно и выбрать формат числа, соответствующий поставленной задаче.



(рис.3.17.14). Диалоговое окно «Формат ячеек»

Операции форматирования внешнего вида текстовых данных (а также чисел) группированы на вкладках «Выравнивание», «Шрифт», «Граница» и «Заливка» того же диалогового окна «Формат ячеек». Ms. Excel предоставляет широкие возможности выравнивания данных в ячейке. По умолчанию выбран вариант выравнивания «По значению». Это значит, что выравнивание информации зависит от её типа: текст выравнивается по левому краю, а цифры - по правому. Выбор вариантов «По левому краю», «По правому краю» и «По центру» помещает данные указанным образом вне зависимости от того, к какому типу они принадлежат.

Выбор варианта «с заполнением» приводит к тому, что данные, введенные в ячейку, повторяются до тех пор, пока не заполнят всю ячейку целиком. При этом реально информация содержится только в первом сегменте ячейки. Данный вариант применим только при форматировании текста по горизонтали.



(рис.3.17.15). Вкладка «Выравнивание» диалогового окна «Формат ячеек»

Формат	Описание
Общий	Используемый по умолчанию формат чисел. Выравнивание по правому краю
Числовой	Более гибкий формат представления чисел, в котором можно изменять количество знаков, показываемых после запятой, разделять группы разрядов и изменять цвет для отрицательных значений, а также выделять их путем заключения в скобки
Денежный	Общий формат для представления денежных знаков, для которого возможно указать знак валюты, число знаков после запятой, а также выделить отрицательные значения цветом и скобками
Финансовый	Отличается от денежного формата выравниванием величин по разделителю целой и дробной части
Дата	Общий формат представления даты с использованием нескольких стандартных стилей
Время	Общий формат представления времени с использованием нескольких стандартных стилей
Процентный	Формат, в котором значение выделенной ячейки умножается на 100 и выводится на экран с символом процента Дробный Формат вывода чисел в виде дробей
Экспоненциальный	Запись чисел, содержащих слишком большое число знаков, в экспоненциальном представлении
Текстовый	Формат, в котором любые значения обрабатываются как строки, вне зависимости от их содержания. Выравниваются по левой границе ячейки и показываются в том виде, как были введены
Дополнительный	Формат, предназначенный для работы с базами данных и списками адресов. Выстраивает данные в порядке убывания или возрастания
Все форматы	Список стандартных, а также созданных пользователем числовых форматов

(табл.3.17.3) *Форматы чисел в Ms. Excel*

Формулы и функции.

Формулой называется уравнение, вычисляющее новое значение на основе уже существующих.

Примеры простейших формул:

$$=B2+C2+D2+E2;$$

$$=СУММ(B2:E2)*1,2.$$

В первой из них напрямую суммируются значения набора ячеек, а во второй для выполнения этой операции используется стандартная функция, а затем результат умножается на постоянное число. Эти формулы имеют несколько общих черт:

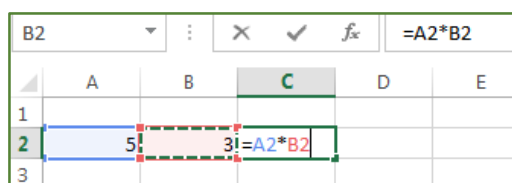
- каждая из них начинается со знака равенства (=). Этот знак указывает программе, что следующие за ним символы являются частью формулы, которую нужно вычислить, а результат этих вычислений должен быть показан в ячейке. Если пропустить этот знак, *Excel* воспримет формулу как текст и вычисление результата производиться не будет.

- каждая формула использует один или несколько арифметических операторов (в данном случае это знаки сложения, умножения, скобки). Однако эти операторы не являются необходимой принадлежностью любой формулы. Ее можно создать и с помощью одной или нескольких функций.

Следует отметить, что в математике (на листе бумаги) формулы «двумерные», а в *Ms. Excel* формулы нужно располагать в одной строке, т.е. они *одномерные* поэтому приходится вводить дополнительные скобки, которых нет в исходной формуле. Формула вида $\frac{1+5x}{y}$ будет выглядеть в *Ms. Excel* следующим образом: $= (1+5*x)/y$.

Таким образом, под формулой в *Ms. Excel* понимается набор чисел и ссылок на числовые ячейки, соединенных знаками математических операций.

Для того чтобы задать ссылку на ячейку (или диапазон данных), нужно указать в формуле её имя: вручную (используя символы латинского алфавита) либо щелчком (выделением) соответствующей ячейки (диапазона данных) в процессе ввода формулы.



(рис.3.17.16). Ввод формулы в Excel

После завершения ввода формулы в таблице отобразится *результат вычисления*. Собственно, формулу можно при этом увидеть либо в *строке формул*, либо в режиме редактирования *формулы (F2)*.

Копирование формул в соседние ячейки, осуществляется с помощью команды подменю, появляющегося при выборе в меню «*Редактирование*» команды «*Заполнить*» (*главная вкладка*). При этом происходит автоматическая подстройка формулы таким образом, что она начинает соответствовать строкам и столбцам, в которые происходит копирование. Например, если формула копируется в ячейки, расположенные ниже в столбце, Excel изменяет номера строк, включая в них «правильные» ссылки на ячейки. Однако в ряде случаев удобнее продублировать формулу, используя функцию *автозаполнения*.

Относительные и абсолютные ссылки на ячейки

При дублировании формул программа меняет *ссылки на ячейки* в соответствии с новым местоположением формулы по заданному шаблону. Чтобы изменить такой порядок действий, нужно указать, что ссылка на ячейку является *абсолютной*. Это делается путем введения знака *доллара (\$)* перед индикаторами строки и столбца. Адрес такой ссылки не будет изменяться при копировании её в другую ячейку.

Тип ссылок, содержащий абсолютные и относительные ссылки одновременно, называется *смешанным*. Например, ссылка вида ***\$C8*** состоит из *абсолютной ссылки на столбец* и *относительной ссылки на строку*.

При дублировании формулы, содержащей такую ссылку, она всегда будет относиться к столбцу ***C***, в то время как индикатор строки будет изменяться. Соответственно, в ссылке вида ***C\$8*** номер строки фиксирован, в то время как ссылка на столбец меняется. Для изменения типа ссылок необходимо после их ввода нажать клавишу ***F4***.

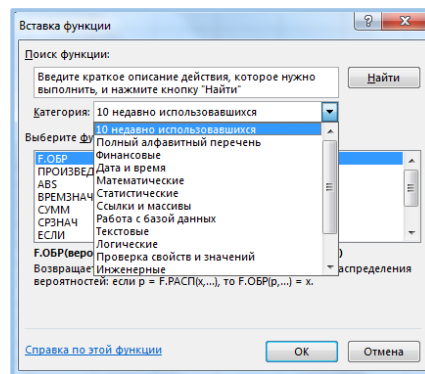
Для выполнения более сложных операций над числами и текстом используются встроенные функции, то есть заранее заданные уравнения, которые на основе одного или нескольких значений возвращают один результат. Приложение Excel содержит более двухсот функций, относящихся к нескольким категориям.

Категория функций	Описание
Финансовые функции	Используются для вычисления выплаты по ссудам, повышения и понижения ценности
Функции даты и времени	Вычисления, относящиеся к датам и времени
Математические функции	Получение результатов арифметических и тригонометрических функций
Статистические функции	Вычисление средних, дисперсии, среднеквадратичного отклонения и т.д.
Функции ссылок и массивы	Используются для вычислений на основе таблиц данных
Функции баз данных	Работа со списками и внешними базами данных
Текстовые функции	Сравнение, преобразование и форматирование текста в ячейках
Логические функции	Вычисления, результатом которых является одно из двух значений: <i>Истина</i> или <i>Ложь</i>
Информационные функции	Определяют, имеется ли в вычислениях ошибка

(табл.3.17.4). Категории функций MS Excel

Необходимо соблюдать *синтаксис* при вводе функций так, как только в этом случае программа сможет вернуть корректный результат. Аббревиатуры в скобках называются аргументами. Это значения, на основе которых функция возвращает итоговый результат.

Диалоговое окно вставки функций находится во вкладке «*Формулы*». Кроме того, кнопка вызова мастера функций находится слева от строки формул.



(рис.3.17.17). Диалоговое окно «Вставка функции»

Ошибки функций

Если в процессе ввода функции была допущена ошибка, в ячейке вместо результата вычислений появится код, называемый значением ошибки. Этот код начинается знаком решетки (#) и заканчивается восклицательным знаком. Исправить ошибку можно в строке формул. Также можно удалить формулу и ввести её заново или щелкнуть на появившейся слева кнопке с

восклицательным знаком, вызвать справку по этой ошибке, посмотреть этапы вычислений или запустить *панель аудита формул*. Часто встречающиеся значения ошибок перечислены в таблице.

Значение ошибки	Описание
#ДЕЛ/0! (#DIV/0!)	В формуле происходит деление на ноль
#Н/Д(#N/A)	Не был указан аргумент функции
#ИМЯ? (#NAME?)	Ms. Excel не может распознать имя, используемое в формуле
#ПУСТО! (#NULL!)	В формуле задано пересечение двух областей, которые в действительности не имеют общих ячеек
#ЧИСЛО! (#NUM!)	Данная ошибка возникает при неправильных числовых значениях в формуле или функции
#ССЫЛ! (#REF!)	В формуле имеется ссылка на несуществующие данные
#ЗНАЧ! (#VALUE!)	В качестве аргумента используется текстовое значение
#####	Результат вычислений слишком широк для того, чтобы поместиться в ячейке

(табл.3.17.5). Значения ошибок функций

Имена ячеек

Для упрощения работы с функциями можно присвоить имя ячейке или диапазону ячеек и затем использовать его в вычислениях. Имена должны начинаться с буквы и не могут содержать пробелы. Рекомендуется ограничить длину имен пятнадцатью знаками, чтобы имя могло уместиться в поле «Имя» и потом его можно было легко использовать в формулах.

Для присвоения имени диапазону ячеек с помощью меню, необходимо сначала выделить набор ячеек. Затем нужно выбрать команду «Присвоить имя» в контекстном меню. Аналогичную операцию можно выполнить, используя поле «Имя». После именованной одной ячейки (или диапазона) её имя появляется в текстовом поле «Имя» при любом её выделении. Если же имя было присвоено диапазону, для его появления в этом поле необходимо выделить все ячейки диапазона. Помимо этого, для работы с именами рекомендуется использовать раздел «*Определённые имена*» вкладки «*Формулы*».

При редактировании имени программа автоматически обновляет все содержащие его формулы. После удаления имени, включенного в формулу, в содержащей эту формулу ячейке появится значение ошибки #ИМЯ? Для того чтобы получить доступ к редактированию и удалению имен, используется

команда «Диспетчер имён», расположенная в меню «*Определённые имена*». В этом диалоговом окне, помимо редактирования имён, можно также изменять ссылки на ячейки диапазона, используемого в имени.

Автоматизация задач с помощью средства записи макросов - Excel.

Для автоматизации повторяющихся задач в **Microsoft Excel** можно *быстро записать макрос*. При записи макроса записываются все действия, описанные в **Visual Basic** для приложений (*VBA*) коде.

Макрос (от англ. *macros*) - программный объект, который во время вычисления заменяется на новый объект, создаваемый определением макроса на основе его аргументов, затем выражается обычным образом.

Эти действия могут включать ввод *текста* или *чисел*, *нажатие ячеек* или *команд* на ленте или в меню, *форматирование ячеек*, *строк* или *столбцов*, а также *импорт данных* из внешнего источника, например, **Microsoft Access**.

Приложение **Visual Basic (VBA)** - это подмножество мощного языка программирования Visual Basic, которое входит в большинство приложений Office.

Несмотря на то, что VBA обеспечивает возможность автоматизации процессов между приложениями Office, нам не нужно знать код **VBA** или программное программирование, если это нужно.

Важно знать, что *при записи макроса* регистрируются почти все наши действия. Поэтому если мы допустим ошибку, например, нажмем не ту кнопку, средство записи макросов регистрирует это действие.

В таком случае можно снова записать всю последовательность или изменить код VBA. Поэтому перед записью процесса следует хорошо проработать его. Чем точнее мы запишем последовательность, тем более эффективно будет работать макрос.

Макросы и средства **VBA** находятся на вкладке **Разработчик**, которая по умолчанию скрыта, поэтому сначала нужно включить ее.

Отображение вкладки «Разработчик»

Вкладка "разработчик" не отображается по умолчанию, но ее можно добавить на ленту.

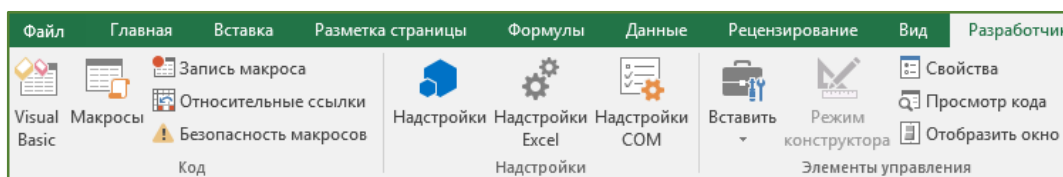
На вкладке файл выберите **Параметры-Настройка ленты**.

В разделе Настройка ленты в списке Основные вкладки установите флажок **Разработчик**.

После того как мы отобразили вкладку, вкладка разработчик останется видимой, если не снять флажок или переустановить приложение Microsoft Office.

Вкладка "разработчик" используется для того, чтобы перейти к следующему элементу или использовать следующее:

- написания *макросов*;
- запуска ранее записанных макросов;
- использования *XML-команд*;
- использования *элементов ActiveX*;
- создания приложений для *программ Microsoft Office*;
- использования элементов управления формы в *Microsoft Excel*;
- работы с *таблицей* свойств фигуры в *Microsoft Visio*;
- создания *фигур и наборов элементов* в *Microsoft Visio*.



(рис.3.17.18). Вкладка "разработчик"

Запись макроса

Перед *записью макросов* полезно знать следующее:

- Макрос, записанный для работы с диапазоном *Excel*, будет выполняться только для ячеек этого диапазона. Поэтому если мы добавим в диапазон новую строку, макрос не будет применяться к ней.

- Если нам нужно записать длинную последовательность задач, то вместо этого использовать несколько более мелких макросов.

- В макросе могут содержаться и задачи, не относящиеся к *Excel*. Процесс макроса *может охватывать прочие приложения Office* и другие программы, которые поддерживают *Visual Basic* для приложений (*VBA*). Например, мы можем записать макрос, который сначала обновляет таблицу в *Excel*, а затем открывает *Outlook* для ее отправки по электронной почте.

Внимание: *Макросы нельзя отменить.* Прежде чем впервые запускать записанный макрос, надо сохранить книгу или создать ее копию, чтобы предотвратить внесение нежелательных изменений. Если не устраивают результаты выполнения макроса, мы можем закрыть книгу, не сохраняя ее.



Контрольные вопросы и задания

1. Дайте определение *электронной таблицы*.
2. Перечислите *основные возможности* электронных таблиц.
3. Опишите *принцип работы* с электронными таблицами.
4. Перечислите *функции* в электронных таблицах.
5. Опишите правила *ввода и обработки данных*.
6. Возможности электронных таблиц
7. Типы данных в таблице *MS-EXCEL*
8. Что такое макрос. Как редактировать макрос?
9. Что такое VBA. Какие типы данных поддерживает VBA?
10. Что такое Макрорекодер?
11. Заполните таблицу «*Диагностика ошибок*».

Ошибка	Ошибка
#####	
#ЗНАЧ!	
#ДЕЛ/0!	
#ИМЯ?	
#Н/Д	
#ССЫЛКА!	
#ЧИСЛО!	

18-§. ОСНОВНЫЕ ПОНЯТИЯ МУЛЬТИМЕДИА. ОСНОВЫ РАБОТЫ С АУДИО И ВИДЕО ИНФОРМАЦИЕЙ

Ключевые слова: *мультимедиа, мультимедийность, текст, звук, видеоизображение, графическое изображение, анимация, аудио, видео, слайд, кино, браузер, autoplay, кодирование, декодирование, провайдер, значение, очередь, мобильное устройство, базы данных.*

Впервые термин «мультимедиа» появился в 1965 году и активно использовался вплоть до конца семидесятых годов для описания экстравагантных для того времени театрализованных шоу, использующих разные виды и формы представления информации: *слайды, кино, видео, аудио фрагменты, световые эффекты и живую музыку.*

Мультимедиа (*multimedia*) - это современная компьютерная информационная технология, позволяющая объединить в компьютерной системе *текст, звук, видеоизображение, графическое изображение и анимацию (мультипликацию).*

Однако всплеск интереса в конце 80-х годов к применению *мультимедиа-технологии* в гуманитарных областях (*в частности, в историко-культурной*) связан несомненно с именем выдающегося американского компьютерщика-бизнесмена **Билла Гейтса**, которому принадлежит идея создания и успешной реализации на практике мультимедийного продукта на основе служебной (!) музейной инвентарной базы данных с использованием в нем всех возможных *"сред"*: изображений, звука, анимации, гипертекстовой системы (*"National Art Gallery. London"*).

Именно этот продукт аккумулировал в себе три основных принципа мультимедиа:

1. Представление информации с помощью комбинации множества воспринимаемых человеком сред (*собственно термин происходит от англ. multi - много, и media - среда*);

2. Наличие нескольких сюжетных линий в содержании продукта (в том числе и выстраиваемых самим пользователем на основе "свободного поиска" в рамках предложенной в содержании продукта информации);

3. Художественный дизайн интерфейса и средств навигации.

Мультимедиа



(рис.3.18.1). Мультимедиа



(рис.3.18.2). Объект-контейнер

Несомненным достоинством и особенностью технологии являются следующие возможности мультимедиа, которые активно используются в представлении информации:

✚ возможность хранения большого объема самой разной информации на одном носителе (до 20 томов авторского текста, около 2000 и более высококачественных изображений, 30-45 минут видеозаписи, до 7 часов звука);

✚ возможность увеличения (детализации) на экране изображения или его наиболее интересных фрагментов, иногда в двадцатикратном увеличении (режим "луна") при сохранении качества изображения. Это особенно важно

для презентации произведений искусства и уникальных исторических документов;

- ✚ возможность сравнения изображения и обработки его разнообразными программными средствами с научно - исследовательскими или познавательными целями;

- ✚ возможность выделения в сопровождающем изображении текстовом или другом визуальном материале *"горячих слов (областей)"*, по которым осуществляется немедленное получение справочной или любой другой пояснительной (в том числе визуальной) информации (*технологии гипертекста и гипермедиа*);

- ✚ возможность осуществления непрерывного музыкального или любого другого аудио сопровождения, соответствующего статичному или динамичному визуальному ряду;

- ✚ возможность использования видефрагментов из фильмов, видеозаписей и т.д., функции *"стоп-кадра"*, покадрового *"пролистывания"* видеозаписи;

- ✚ возможность включения в содержание диска баз данных, методик обработки образов, анимации (к примеру, *сопровождение рассказа о композиции картины графической анимационной демонстрацией геометрических построений ее композиции*) и т.д.;

- ✚ возможность подключения к глобальной сети *Internet*;

- ✚ возможность работы с различными приложениями (*текстовыми, графическими и звуковыми редакторами, картографической информацией*);

- ✚ возможность создания собственных *"галерей"* (выборки) из представляемой в продукте информации (*режим "карман" или "мои пометки"*);

- ✚ возможность *"запоминания пройденного пути"* и создания *"закладок"* на заинтересовавшей экранной *"странице"*;

- ✚ возможность автоматического просмотра всего содержания продукта (*"слайд-шоу"*) или создания анимированного и озвученного

"путеводителя-гида" по продукту ("говорящей и показывающей инструкции пользователя"); включение в состав продукта игровых компонентов с информационными составляющими;

✚ возможность "свободной" навигации по информации и выхода в основное меню (*укрупненное содержание*), на полное оглавление или вовсе из программы в любой точке продукта.

Основные носители

В качестве носителей мультимедийных продуктов используются средства, способные хранить огромное количество самой разнообразной информации. Как правило, мультимедийные продукты ориентированы либо на компьютерные носители и средства воспроизведения (*CD-ROM*), либо на специальные телевизионные приставки (*CD-i*), либо на телекоммуникационные сети и их системы.

- **CD-ROM** (*CD - Read Only Memory*) - оптический диск, предназначенный для компьютерных систем. Среди его достоинств - многофункциональность, свойственная компьютеру, среди недостатков - отсутствие возможности пополнения информации - ее "*до записи*" на диск, не всегда удовлетворительное воспроизведение видео и аудио информации.

- **CD-i** (*CD - Interactive*) - специальный формат компакт-дисков, разработанный фирмой *Philips* для TV приставок. Среди его достоинств - высокое качество воспроизведения динамичной видеoinформации и звука. Среди недостатков - отсутствие многофункциональности, неудовлетворительное качество воспроизведения статичной визуальной информации, связанное с качеством *TV мониторов*.

- **Video-CD** (*TV формат компакт-дисков*) - замена видеокассет с гораздо более высоким качеством изображения. Среди недостатков - отсутствие многофункциональности и интерактивности (*на которые он при создании и не был рассчитан*).

- **DVD** - это семейство оптических дисков, одинакового размера с компакт-дисками (*CD*), но значительно большей емкости хранения, достигнутой за счет увеличения плотности записи.

Важнейшие преимущества DVD технологии

Сегодня DVD - это уже широко распространенная, проверенная временем и в тоже время динамично развивающаяся технология с огромным потенциалом.

- запись и воспроизведение высококачественного *видео и аудио* в реальном времени, эффективная работа с компьютерной мультимедийной информацией, а также обеспечение эффективного произвольного доступа к данным, хранимыми в виде множества мелких файлов;

- объем диска до *4,7 ГБ (около 2-х часов MPEG-2)* на сторону для записи в один слой и *8,5 ГБ* на сторону для двуслойной записи;

- возможность записи информации в *два слоя* на каждую из сторон;

- единая файловая система *UDF*;

- возможность записи и многократной *перезаписи DVD дисков*;

- обратная совместимость с существующими *CD-дисками* - геометрические размеры *DVD* и *CD дисков* идентичны, все DVD оборудование способно читать диски *CD-Audio* и *CD-ROM (спецификация MultyRead)*.

Форматы DVD только для чтения.

DVD-ROM (*Digital Versatile Disc Read Only Memory*). Диски формата *DVD-ROM* предназначены для использования в компьютерной технике. Информация заносится на диск единственный раз - при его производстве.

Современные *DVD-ROM* - устройства третьего поколения - используют режим контроля вращения (*CAV*) с максимальной скоростью чтения *4x-6x (5.5 — 8.3 Мб/с)* и более. Современные *DVD-ROM* приводы (*дисководы*) поддерживают чтение практически всех форматов, включая диски *CD*.

DVD-Video. Формат *DVD-Video* предназначен для хранения и воспроизведения видео. Как и *DVD-ROM*, эта спецификация определяет

возможность *только чтения информации* - воспроизведение записей с помощью видеоплееров (*видеорекодеров*).

DVD-RAM (*Digital Versatile Disc Random Access Memory*) перезаписываемый формат, разработанный компаниями *Panasonic, Hitachi, Toshiba*.

DVD+RW (*Digital Versatile Disc ReWritable*). Формат DVD+RW продвигается только его разработчиками - компаниями *Hewlett-Packard, Mitsubishi Chemical, Philips, Ricoh, Sony и Yamaha* (*не поддержан DVD-форумом*).

На дисках *DVD+RW* можно записать как потоковое видео или звук, так и компьютерные данные. Диски формата *DVD+RW* могут быть перезаписаны около 1000 раз.

Формат	Спецификация	Количество сторон	Количество слоев на сторону	Емкость, ГБ ³
DVD-Video и DVD-ROM	DVD-5	1	1	4.7, или более 2 часов видео
	DVD-9	1	2	8.5, или более 4 часов видео
	DVD-10	2	1	9.4, или более 4,5 часов видео
	DVD-14	2	1+2	13.2, или более 6,5 часов видео
	DVD-18	2	2	17.1, или более 8 часов видео
DVD-RAM (DVD-VR)	DVD-RAM 1.0	1	1	2.6
		2	1	5.2
	DVD-RAM 2.0	1	1	4.7
		2	1	9.4
DVD-R	DVD-R 1.0	1	1	3.9
	DVD-R 2.0	1	1	4.7
		2	1	9.4
DVD-RW	DVD-RW 2.0	1	1	4.7
		2	1	9.4

(табл.3.18.1). Емкости DVD дисков

³1ГБ=1073741824 байт

ТИПЫ ДАННЫХ МУЛЬТИМЕДИА-ИНФОРМАЦИИ И СРЕДСТВА ИХ ОБРАБОТКИ

Стандарт MPC (*точнее средства пакета программ Multimedia Windows - операционной среды для создания и воспроизведения мультимедиа-информации*) обеспечивают работу с различными типами данных мультимедиа.

Мультимедиа-информация содержит не только традиционные статистические элементы: *текст, графику*, но и *динамические видео -, аудио- и анимационные последовательности*.

Неподвижные изображения. Сюда входят *векторная графика и растровые картинки*; последние включают изображения, полученные путем оцифровки с помощью различных *плат захвата, грубберов, сканеров*, а также созданные на компьютере или закупленные в виде *готовых банков изображений*.

Для построения мультимедиа системы необходима дополнительная аппаратная поддержка: *аналого-цифровые и цифроаналоговые преобразователи* для перевода аналоговых аудио и видео сигналов в цифровой эквивалент и обратно, *видеопроцессоры* для преобразования обычных телевизионных сигналов к виду, воспроизводимому *электронно лучевой трубкой дисплея*, *декодеры* для взаимного преобразования телевизионных стандартов, специальные интегральные схемы для сжатия данных в файлы допустимых размеров и так далее. Все оборудование, отвечающее за звук, объединяются в так называемые *звуковые карты*, а за видео в *видео карты*. Далее рассматривается подробно и в отдельности об устройстве и характеристиках *звуковых карт, видео карт и CD-ROM приводах*.

Звуковые карты

Для звуковых карт IBM совместимых компьютеров прослеживаются следующие тенденции:

Во-первых, для воспроизведения звука вместо частотной модуляции (*FM*) теперь все больше используют табличный (*wavetable*) или *WT синтез*,

сигнал полученный таким образом, более похож на звук реальных инструментов, чем при *FM синтезе*. Используя соответствующие алгоритмы, даже только по одному тону музыкального инструмента можно воспроизводить все остальное, то есть восстановить его полное звучание. Выборки таких сигналов хранятся либо в постоянно запоминающем устройстве (*ROM*) устройства, либо программно загружаются в оперативную память (*RAM*) звуковой карты.

В более дешевых платах чаще реализован частотно модулированный синтез с использованием *синусоидальным колебаний*, что в результате приводит к не совсем точному звучанию инструментов, отражение звука и рева, характерных для последнего поколения игр. Расположенная на плате микросхема для волнового синтеза хранит записанные заранее оцифрованные образцы (*Samples*) звучания музыкальных инструментов и звуковых эффектов.

Пионером в реализации *WT* синтеза стала в 1984 году фирма *Ensoning*. Вскоре *WT синтезаторы* стали производить такие известные фирмы, как *Emu, Korg, Roland и Yamaha*.

Видеокарты

Имеется большое количество устройств, предназначенных для работ с видеосигналами на ПК. Условно можно разбить на несколько групп: устройства для ввода и захвата видеопоследовательностей (*Capture play*), *фреймграбберы (Framegrabber)*, *TV-тюнеры*, преобразователи сигналов *VGATV и MPEG-плейеры*.

TV тюнеры.

Эти устройства выполняются обычно в виде карт или бокса (*небольшой коробочки*). Они преобразуют аналоговый видеосигнал, поступающий по сети кабельного телевидения или от антенны, от видеомагнитофона или камкордера (*camcorder*). *TV-тюнеры* могут входить в состав других устройств таких как *MPEG-плейеры* или *фреймграбберы*. Некоторые из них имеют встроенные микросхемы для преобразования звука. Ряд тюнеров имеют возможность для вывода телетекста.

Преобразователи VGA-TV.

Данные устройства транслируют сигнал в цифровом образе VGA изображения в *аналоговый сигнал*, пригодный для ввода на телевизионный приемник. Производители обычно предлагают подобные устройства, выполненные либо как внутренние ISA карта, либо как внешний блок.

MPEG-плейеры.

Данные устройства позволяют воспроизводить последовательности видеоизображения (*фильмы*) записываемых на компакт-дисках качеством VNS. Скорость потока сжатой информации не превышает обычно *150 Кбайт/с*.

Основная сложность задачи, решаемой *MPEG кодером*, состоит в определении для каждого конкретного видеопотока оптимального соотношения между тремя видами изображения: (*I*) ntra, (*P*) redicted и (*B*) idirectional. Первым MPEG - плейерам была плата *Reel Magic* компании *Sigina Desing* в 1993 году.



(рис.3.18.3). Аппаратные средства МТ

Технологии мультимедиа

Мультимедиа-технология *позволяет одновременно использовать различные способы представления информации: числа, текст, графику, анимацию, видео и звук.*

Важной особенностью *мультимедиа-технологии* является ее интерактивность, т.е. то, что в диалоге с компьютером пользователю отводится активная роль. Графический интерфейс мультимедийных проектов обычно содержит различные управляющие элементы (*кнопки, текстовые окна и т.д.*).

В последнее время создано много мультимедийных программных продуктов:

- энциклопедии по *истории, искусству, географии, биологии* и др.;
- обучающие программы по *иностранным языкам, физике, химии* и т.д.

Мультимедийный компьютер, т.е. компьютер, который может работать с мультимедийными данными, должен иметь *звуковую плату* для воспроизведения и синтеза звука с подключенными *акустическими колонками (наушниками)* и *микрофоном* и *дисковод CD-ROM*, позволяющий хранить большие по объему мультимедийные данные.

Одним из типов мультимедийных приложений являются *компьютерные презентации*.

Компьютерная презентация *представляет собой последовательность слайдов, содержащих мультимедийные объекты: числа, текст, графику, анимацию, видео и звук.*

Публикации во Всемирной паутине реализуются в форме *мультимедийных Веб-сайтов*, которые кроме текста могут включать в себя *иллюстрации, анимацию, звуковую и видеoinформацию*.

Виды мультимедиа технологий:

- *Презентация*
- *Интерактивная доска*
- *Система интерактивного опроса*
- *Различные образовательные программы*

- *Мультимедийный экран*
- *Сетевые образовательные программы*
- *Имитационные технологии*
- *Диагностические комплексы.*

Сферы применения мультимедиа:

- Информационная и рекламная деятельности
- Шоу–бизнес
- Создание персональных фоно и видеотек
- Компьютерные тренажеры
- Компьютерные игры
- Обучающие программы
- Энциклопедии.

Программные средства мультимедийной технологии



(рис.3.18.4). Программные средства МТ

Аудиоинформация

Любой мультимедийный - ПК имеет в своем составе плату-аудио адаптер. Аудио адаптер дал компьютеру не только стереофоническое звучание, но и возможность записи на внешние носители звуковых сигналов. Как уже было сказано ранее, дисковые накопители ПК совсем не подходят для записи обычных (аналоговых) звуковых сигналов, так как рассчитаны для записи только цифровых сигналов, которые практически не искажаются при их передаче по линиям связи.

Работа с аудиоинформацией.

Звук - это колебание частиц воздуха, непрерывный сигнал с меняющейся амплитудой.

При кодировании звука этот сигнал надо представить в виде последовательности *нулей и единиц*. На качество воспроизведения закодированного звука в основном влияют два параметра: *частота дискретизации* - количество измерений амплитуды за секунду в герцах и *глубина кодирования звука (битрейт⁴)* - размер в битах, отводимый под запись значения амплитуды.

Чем больше значения обоих показателей, тем точнее представление звука. Но при этом увеличивается размер звукового файла. Поэтому в зависимости от характера звука, требований, предъявляемых к его качеству и объему занимаемой памяти, выбирают некоторые компромиссные значения. Для компьютерной обработки такой аналоговый сигнал нужно преобразовать в последовательность *двоичных чисел*.

Форматы аудиофайлов.

В процессе развития *индустрии звукозаписи* для решения различных задач были разработаны различные физические методы хранения аудиоданных, например, *виниловые пластинки, магнитная лента, компакт-диски, теперь - DVD*, и т.д., точно таким же образом появилось множество различных компьютерных методов хранения аудиоданных (*форматов аудиофайлов*). На сегодняшний день их известно около четырехсот.

Полезными в работе могут быть редакторы аудиофайлов, которые могут *укоротить длинный звуковой файл, нарезать звуковые фрагменты, произвести сглаживание звучания*, добавить какие-либо звуковые эффекты к *оригинальному треку*, и многое другое. Среди таких программ можно отметить:

⁴ *Битрейт (Bit Rate, поток данных)*

- ✚ **Volume2** - удобная программа по управлению уровнем громкости;
- ✚ **Soundboard 4 Voice Chat** - программа может помочь при озвучивании каких-либо видеофрагментов, позволяет создавать свои и загружать аудио пакеты;
- ✚ **Mixxx** - программа для микширования различных типов звуковых файлов, поддерживает работу с плейлистами, имитацию аналоговых регуляторов и проигрывания виниловых пластинок, есть возможность синхронизации скорости треков, изменения темпа музыки и регулировки частотных характеристик и т.д.



(рис.3.18.5). Работа с аудиоинформацией

Видеоинформация

Видео (*от лат. Video - смотрю, вижу*) - множество технологий записи, обработки, передачи, хранения и воспроизведения визуального или аудиовизуального материала, а также распространённое название для собственно видеоматериала, телесигнала или кинофильма, в том числе записанного на физическом носителе (*видеокассете, видеодиске и т. п.*).

Работа с видеоинформацией.

Видеоинформация непосредственно - это изображение, зафиксированное на *магнитной ленте, кинолентке, фотоснимке* или на *оптическом диске*, с которых оно может быть воспроизведено.



Основные параметры видеосигнала:

- количество (*частота*) кадров в секунду (*число неподвижных изображений, сменяющих друг друга при показе 1 секунды видеоматериала и создающих эффект движения объектов на экране*);
- чересстрочная развёртка;
- разрешение;
- соотношение сторон экрана;
- количество цветов и цветовое разрешение;
- битрейт или ширина видеопотока (*для цифрового видео*).

Программы для обработки видеоматериалов

Для редактирования на компьютере видео потребуется специальное программное обеспечение. Для простейшей работы можно использовать встроенную в Windows программу *Windows Movie Maker*. Для более сложного видеомонтажа можно воспользоваться программой *Ulead Media Studio Pro*.

Видеомонтаж подразумевает получение видеоизображения с одного или нескольких источников, выполнение над видео различных действий и сохранение полученного в результате монтажа видео. Известны два вида монтажа – *монтаж сборкой и монтаж вставкой*. Монтаж сборкой используется для создания, отредактированного видео путем перезаписи из нескольких других записей или источников видеосигнала. Новая сцена добавляется к концу предыдущей. Монтаж вставкой используется для замены одной сцены на другую.

Различают также *линейный и нелинейный* видеомонтаж. Особенность *линейного видеомонтажа* состоит в том, что все операции происходят в реальном времени. Чтобы добиться высокой скорости работы, эффекты и операции осуществляют с помощью специальной аппаратуры. В этом случае роль компьютера сводится к координации работы устройств линейного монтажа и автоматизации рутинной ручной работы.

При использовании *нелинейного видеомонтажа* все фрагменты исходного видео должны быть введены в компьютер, а затем с помощью специальной программы над этим фрагментом выполняются различные операции. При этом в зависимости от используемой программы можно выполнить практически любые преобразования над исходными фрагментами видео. В результате полученное видео можно сохранить на *диске компьютера, записать на видеомаягнитофон или цифровую камеру.*



Контрольные вопросы и задания

1. Основные понятия мультимедиа.
2. Типы данных мультимедиа-информации и средства их обработки
3. Звуковые карты
4. Лазерные диски, CD-ROM
5. Видеокарты
6. TV тюнеры.
7. Фрейм грабберы.
8. Преобразователи VGA-TV.
9. MPEG-плейеры.
10. Технологии мультимедиа
11. Работа с аудио и видео информацией

19-§. "КОМПЬЮТЕРНАЯ ГРАФИКА. ВИДЫ КОМПЬЮТЕРНОЙ ГРАФИКИ"

Ключевые слова: *вектор, растр, фрактал, графика, изображение, 3D-моделирование, компьютерная графика, графический редактор, фрагмент*

Компьютерная графика - область деятельности, в которой компьютеры наряду со специальным программным обеспечением используются в качестве инструмента как для создания (синтеза) и редактирования изображений, так и для оцифровки визуальной информации, полученной из реального мира, с целью дальнейшей её обработки и хранения.

Конечным продуктом компьютерной графики является изображение (*графическая информация*). Изображение можно разделить на:

1. *Рисунок* – графическая форма изображения, в основе которой лежит линия.
2. *Чертеж* – это контурное изображение проекции некоторых реально существующих или воображаемых объектов.
3. *Картина* – тоновое черно-белое или цветное изображение.

Разрешение изображения – свойство самого изображения. Оно измеряется в точках на дюйм (*dpi*) и задается при создании изображения в графическом редакторе или с помощью сканера. Значение разрешения изображения хранится в файле изображения и неразрывно связано с другим свойством изображения – его физическим размером.

Физический размер изображения. Может измеряться как в пикселях, так и в единицах длины (*миллиметрах, сантиметрах, дюймах*). Он задается при создании изображения и хранится вместе с файлом.

Виды компьютерной графики

Различают *три вида* компьютерной графики. Это *растровая графика, векторная графика* и *фрактальная графика*. Они отличаются принципами формирования изображения при отображении на экране монитора или при печати на бумаге.

Растровая графика

Растровое изображение - изображение, представляющее собой сетку пикселей - цветных точек (обычно прямоугольных) на мониторе, бумаге и других отображающих устройствах.

Важными характеристиками изображения являются:

- Размер изображения в пикселях – может выражаться в виде количества пикселей по ширине и по высоте (*800x600px, 1024x768px, 1600x1200px* и т.д.) или же в виде общего количества пикселей (*так,*

изображение размером 1600×1200 px состоит из 1920000 точек, то есть примерно из двух мегапикселей);

- Количество используемых цветов или глубина цвета (эти характеристики имеют следующую зависимость: $N=2^I$, где N – количество цветов, а I – глубина цвета);
- цветовое пространство (цветовая модель) *RGB*, *CMYK*, *XYZ*, *YCbCr* и др.
- разрешение – справочная величина, говорящая о рекомендуемом размере пикселя изображения.
- Разрешение изображения – величина, определяющая количество точек (элементов растрового изображения) на единицу площади (или единицу длины). Не путать с размером сетки изображения.

Растровую графику редактируют с помощью *растровых графических редакторов*. Создаётся растровая графика *фотоаппаратами, сканерами, непосредственно в растровом редакторе*, а также путём экспорта из *векторного редактора* или в виде снимков экрана.

Форматы

Растровые изображения обычно *хранятся в сжатом виде*. В зависимости от типа сжатия может быть возможно или невозможно восстановить изображение в точности таким, каким оно было до сжатия (*сжатие без потерь или сжатие с потерями соответственно*). Так же в графическом файле может храниться дополнительная информация: об авторе файла, фотокамере и её настройках, количестве точек на дюйм при печати и др.

- **BMP** или *Windows Bitmap* - обычно используется без сжатия, хотя возможно использование алгоритма **RLE**.
- **GIF** (*Graphics Interchange Format*) - устаревающий формат, поддерживающий не более *256 цветов* одновременно. Всё ещё популярен из-за поддержки анимации, которая отсутствует в чистом *PNG*, хотя *ПО* начинает поддерживать *APNG*.

- **PCX** устаревший формат, позволявший хорошо сжимать простые рисованные изображения (*при сжатии группы подряд идущих пикселей одинакового цвета заменяются на запись о количестве таких пикселей и их цвете*).

- **PNG** (*Portable Network Graphics*).

- **JPEG** очень широко используемый формат изображений. Сжатие основано на усреднении цвета соседних пикселей (*информация о яркости при этом не усредняется*) и отбрасывании высокочастотных составляющих в пространственном спектре фрагмента изображения.

- **TIFF** поддерживает большой диапазон изменения глубины цвета, разные цветовые пространства, разные настройки сжатия (*как с потерями, так и без*) и др.

- **RAW** хранит информацию, непосредственно получаемую с матрицы цифрового фотоаппарата или аналогичного устройства без применения к ней каких-либо преобразований, а также хранит настройки фотокамеры. Позволяет избежать потери информации при применении к изображению различных преобразований (*потеря информации происходит в результате округления и выхода цвета пикселя за пределы допустимых значений*). Используется при съёмке в сложных условиях (*недостаточная освещённость, невозможность выставить баланс белого и т.п.*) для последующей обработки на компьютере (*обычно в ручном режиме*). Практически все полупрофессиональные и профессиональные цифровые фотоаппараты позволяют сохранять *RAW* изображения. Формат файла зависит от модели фотоаппарата, единого стандарта не существует.

К программным средствам обработки растровой графики относятся растровые графические редакторы: *GIMP, Paint.NET, Tux Paint, Adobe Photoshop, Adobe Fireworks, Corel Photo-Paint, Corel Paint Shop Pro, Corel Painter, Microsoft Paint.*

Достоинства растровой графики:

1. Растровая графика позволяет создать (*воспроизвести*) практически любой рисунок, вне зависимости от сложности, в отличие, например, от векторной, где невозможно точно передать эффект перехода от одного цвета к другому без потерь в размере файла.

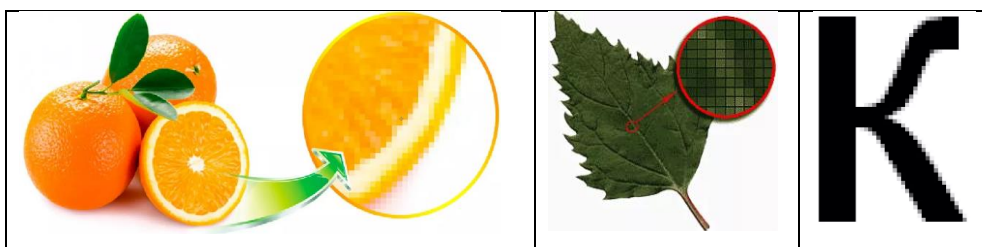
2. Распространённость - растровая графика используется сейчас практически везде: *от маленьких значков до плакатов*.

3. Высокая скорость обработки сложных изображений, если не нужно *масштабирование*.

4. Растровое представление изображения естественно для большинства устройств *ввода-вывода* графической информации, таких как мониторы (*за исключением векторных*), матричные и струйные принтеры, цифровые фотоаппараты, сканеры.

Недостатки растровой графики:

1. *Большой размер файлов* с простыми изображениями.
2. Невозможность идеального *масштабирования*.
3. Невозможность *вывода на печать* на плоттер.



(рис.3.19.1). Растровые изображения

Векторная графика

Векторная графика - способ представления объектов и изображений (*формат описания*) в компьютерной графике, основанный на математическом описании элементарных геометрических объектов, обычно называемых примитивами, таких как: точки, линии, сплайны, кривые Безье, круги и окружности, многоугольники.

К программным средствам создания и обработки векторной графики относятся следующие графические редакторы: *CorelDraw, Adobe Illustrator,*

Adobe Fireworks, а также *векторизаторы (трассировщики)* - специализированные пакеты преобразования *растровых изображений в векторные*.

Как в растровой, так и в векторной графике необходим способ кодирования цвета.

Форматы файлов векторной графики

- *SVG;*
- *CDR;*
- *CGM;*
- *DXF;*
- *GXL;*
- *WMF;*
- *EPS;*
- *AI и другие.*

Достоинства векторной графики

1. полная свобода трансформации (*изменение масштаба без потери качества и практически без увеличения размеров исходного файла*);
2. огромная точность;
3. небольшой размер файла по сравнению с растровым изображением;
4. прекрасное качество печати;
5. отсутствие проблем с экспортом векторного изображения в растровое;
6. объектно-ориентированный характер векторной графики (*возможность редактирования каждого элемента изображения в отдельности*).

Недостатки векторной графики

1. практически невозможно экспортировать из растрового формата в векторный (*можно, конечно, трассировать изображение, хотя получить хорошую векторную картинку нелегко*);
2. невозможно применение обширной библиотеки эффектов, используемых при работе с растровыми изображениями.

Сравнение растровой и векторной графики

Критерий сравнения	Растровая графика	Векторная графика
---------------------------	--------------------------	--------------------------

Способ представления изображения	Растровое изображение строится из множества пикселей	Векторное изображение описывается в виде последовательности команд
Представление объектов реального мира	Растровые рисунки эффективно используются для представления реальных образов	Векторная графика не позволяет получать изображения фотографического качества
Качество редактирования изображения	При масштабировании и вращении растровых картинок возникают искажения	Векторные изображения могут быть легко преобразованы без потери качества
Особенности печати изображения	Растровые рисунки могут быть легко распечатаны на принтерах	Векторные рисунки иногда не печатаются или выглядят на бумаге не так, как хотелось бы
		
	<i>.jpeg, .gif</i>	<i>.svg, .cdr</i>

(табл.3.19.1). Сравнение растровой и векторной графики

Фрактальная графика

Фрактальная графика, как и векторная, основана на математических вычислениях. Однако её базовым элементом является сама математическая формула, то есть никаких объектов в памяти компьютера не хранится и изображение строится исключительно по уравнениям либо системам уравнений. Таким способом строят как простейшие регулярные структуры, так и сложные иллюстрации, имитирующие природные ландшафты и трехмерные объекты.

Фрактал - это объект, отдельные элементарные части которого повторяют (наследуют) свойства своих «родительских» структур.

Фрактальная графика является на сегодняшний день одним из самых быстро развивающихся перспективных видов компьютерной графики.

Математической основой фрактальной графики является *фрактальная геометрия*. Здесь в основу метода построения изображений положен принцип

наследования от, так называемых, «родителей» геометрических свойств объектов-наследников.

Понятия *фрактал, фрактальная геометрия и фрактальная графика*, появившиеся в конце 70-х, сегодня прочно вошли в обиход математиков и компьютерных художников. Слово *фрактал* образовано от латинского *fractus* и в переводе означает *«состоящий из фрагментов»*. Оно было предложено математиком **Бенуа Мандель-Бротом** в 1975 году для обозначения нерегулярных, но само подобных структур, которыми он занимался.

Фрактальная компьютерная графика позволяет создавать абстрактные композиции, где можно реализовать такие композиционные приёмы как, *горизонтали и вертикали, диагональные направления, симметрию и асимметрию.*

С точки зрения *машинной графики* фрактальная геометрия незаменима при генерации искусственных *облаков, гор, поверхности моря*. Фактически благодаря фрактальной графике найден способ эффективной реализации *сложных неевклидовых объектов*, образы которых весьма похожи на природные. Геометрические фракталы на экране компьютера - это узоры, построенные самим компьютером по заданной программе.

Таким образом, с фракталами тесно связаны еще два понятия - *итерация и рекурсия.*

Рекурсия - процесс повторения элементов самоподобным образом.

Итерация - упрощенно говоря - повторное применение какой-либо математической операции.

Следует отметить, что слово *«фрактал»* не является математическим термином и не имеет общепринятого строгого математического определения. Оно может употребляться, когда рассматриваемая фигура обладает какими-либо из следующих свойств:

- имеет нетривиальную структуру во всех масштабах - этим фрактал отличается от регулярных фигур (*таких как окружность, эллипс, график*

гладкой функции): если мы рассмотрим небольшой фрагмент регулярной фигуры в очень крупном масштабе, то он будет похож на фрагмент прямой. Для фрактала увеличение масштаба не ведет к упрощению структуры, поэтому на всех шкалах мы увидим одинаково сложную картину;

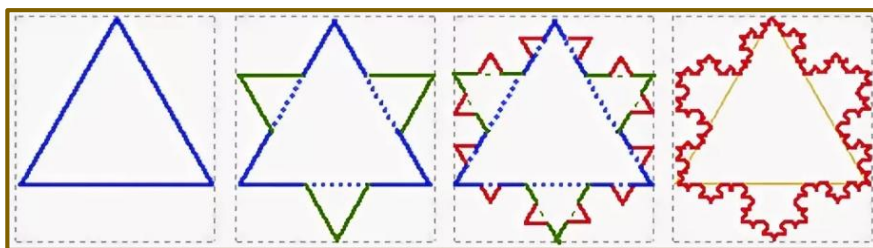
- является самоподобной или приближенно самоподобной;
- имеет дробную метрическую размерность или метрическую размерность, превосходящую топологическую.

Кроме того, для построения фрактала необходимо учитывать начальное состояние и описывающую его формулу - так называемое исходное множество, которое пропускается через некий механизм, вызывающий его отображение и добавляющий отображенное множество к исходному. Этот процесс и называется *итерацией*.



(рис.3.19.2). Чудеса фрактальной графики

Простейшим фрактальным объектом является фрактальный треугольник. Алгоритм его построения следующий. Построить равносторонний треугольник со стороной, a разделить каждую его сторону на три равных отрезка. На среднем отрезке каждой стороны построить внешний равносторонний треугольник со стороной, равной $1/3$ стороны исходного треугольника. На внешних отрезках полученных треугольников, построить равносторонние треугольники, со стороной, равной $1/9a$. Повторять операцию n раз. Фрактальный треугольник можно достраивать до бесконечности. Треугольники последующих поколений наследуют свойства родительских структур.



(рис.3.19.3). Фрактальный треугольник

Существует множество программ, служащих для генерации фрактальных изображений:

Apophysis 3D- используется *fractal flame* - алгоритм. В ней можно создавать как 2Д - изображения, так и псевдо 3Д - графику. На данный момент не поддерживается разработчиками.

Apophysis 7X- Логическое продолжение *Apophysis 3D*, но с поддержкой от разработчиков. Работать в программе нужно, манипулируя треугольниками. Есть генератор мутаций - случайное редактирование треугольников. Серьёзные возможности по трансформациям и практически всем параметрам фрактала.

Chaotica- Инструмент на основе *fractal flame* - алгоритма, но ориентированный более на рендеринг изображения.

Ultra Fractal- Мощный инструмент для фрактальных художников. Поддерживает слои, альфа-каналы, градиенты, создание собственных формул и многое другое.

Форматы фрактальной графики

Форматы фрактальной графики определяют форму и способ хранения файловых данных. Некоторые из них включают в себя большой объем информации. Поэтому их необходимо сжимать. Причем делать это не посредством архивирования, а непосредственно в файле.

Если правильно его выбрать, то сжатие будет происходить автоматически. Есть несколько алгоритмов этой процедуры. Если перед пользователем аппликация, большая часть которой выдержана в одном цвете, то разумно использовать форматы *BMP* и *PCX*. Здесь заменяется последовательность повторяющихся величин.

Диаграмму, которая очень редко, но все-таки используется во фрактальной графике, логично поместить в *TIFF* или *GIF*.

Часть форматов является универсальной. То есть, их можно просмотреть в большинстве редакторов. Но если пользователю важна качественная обработка изображений, тогда нужно применять оригинальную программу.

Форматы фракталы *не поддерживаются браузерами*. Именно поэтому осуществляется их преобразование, если есть необходимость загрузить на тот или иной сайт.

Сферы применения

Применение фрактальной графики можно назвать фактически повсеместным. Более того, эта область постоянно расширяется. На данный момент можно отметить следующие области:

- компьютерная графика;
- применяется в создании компьютерных игр.

Анализ фондовых рынков. Фракталы здесь используются для того, чтобы отметить повторения, которые впоследствии сыграют трейдерам на руку.
Естественные науки. В физике с помощью фрактальной графики моделируются *нелинейные процессы*. В *биологии* она описывает строение кровеносной системы.

На данный момент практикуется применение фракталов в производстве различного оборудования. Например, уже запущен конвейер по созданию антенн, отлично принимающих сигналы.

Достоинства фрактальной графики

Свое распространение совсем недавно заполучила фрактальная графика. Достоинства и недостатки ее слишком размыты, поскольку отсутствует нормальная теоретическая база. Терминология и принципы ее использования до конца не изучены, несмотря на то, что они *действенные и рабочие*.

Достоинства фрактальной графики заключаются в нескольких факторах:

- ✓ Небольшой размер при масштабном рисунке.

✓ Нет конца масштабированию, сложность картинки можно увеличивать бесконечно.

✓ Нет другого такого же инструмента, который позволит создавать сложные фигуры.

✓ Реалистичность.

✓ Простота в создании работ.

Недостатки фрактальной графики

Недостатки фрактальной графики тоже присутствуют.

Во-первых, без компьютера здесь не обойтись. Причем, чем длиннее количество повторений, тем больше *загружается процессор*. Соответственно, только качественное компьютерное оборудование способно справиться с построением сложных изображений.

Во-вторых, присутствуют ограничения в исходных математических фигурах. Некоторые изображения создать посредством фракталов не удастся.

Сходства и различия между фракталом и вектором

Векторная и фрактальная графика очень различаются между собой:

- По кодированию изображений. Вектор задействует контуры разных геометрических фигур, *фрактал – математическую формулу*, в основе которой лежит треугольник.

- По применению. Вектор используют везде, где нужно получить четкий контур. Фрактальная графика более специализирована, она нашла свое применение в *математике и искусстве*.

- По аналогам. Векторными аналогами являются слайды или функции на графиках. У фракталов это – *снежинки или кристаллы*.

Несмотря на многообразие отличительных черт, эти два вида графики объединяет качество изображения. Оно остается неизменным, независимо от уровня масштабирования.

Трехмерная, векторная, растровая, фрактальная графика схожи в одном – все они широко используются *в решении различных компьютерных задач*.

Чтобы получить действительно качественное изображение, нужно задействовать каждую из них.



Контрольные вопросы и задания

1. Компьютерная графика
2. Виды компьютерной графики
3. Растровая графика
4. Достоинства и недостатки растровой графики:
5. Векторная графика
6. Достоинства и недостатки векторной графики
7. Фрактальная графика
8. Достоинства и недостатки фрактальной графики
9. Классы программ для работы с растровой графикой
10. Средства создания изображений:
11. Средства каталогизации изображений:
12. Средства создания и обработки векторных изображений
13. Средства создания фрактальных изображений
14. Наиболее распространенные форматы графических файлов

20-§. ГРАФИЧЕСКИЙ РЕДАКТОР COREL DRAW И ЕГО ВОЗМОЖНОСТИ

Ключевые слова: *объект, контур, заливка, градиент, пипетка, цветовая модель, маска, свойства, сегмент, маркеры, узловые точки, направляющие линии, слой, группа, разгруппировка.*

Работа с компьютерной графикой - одно из самых популярных направлений использования персонального компьютера.

CorelDRAW - графический редактор векторной графики, разработанный канадской корпорацией Corel.

Популярность **CorelDraw** объясняется большим набором средств *создания и редактирования* графических образов, удобным интерфейсом и высоким качеством получаемых изображений. С его помощью можно создавать как простые *контурные рисунки*, так и *эффективные иллюстрации* с поразительным воображением, переливом красок и ошеломляющими эффектами.

Несомненным преимуществом *CorelDraw* является *низкая цена и легкость в использовании*. Формат файлов *CorelDraw* компактный. По умолчанию в него упаковывается как растровая графика, так и векторная графика, градиентные заливки, регулярные текстуры. Файлы *CorelDraw* могут быть экспортированы в *формат PDF*.

Одной из сильных сторон *CorelDraw* является возможность создавать *многостраничные документы* и *создание плакатов большого размера*.

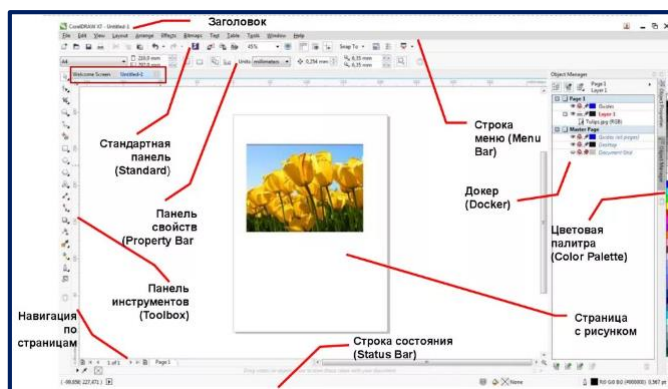
CorelDraw позволяет импортировать файлы следующих форматов:

- Adobe Illustrator (*AI*)
- Растровое изображение Windows (*BMP*)
- Corel DRAW (*CDR*)
- Corel PHOTO-PAINT (*CPT*)
- Ресурс курсоров (*CUR*)
- Microsoft Publisher (*PUB*)
- HTML.

CorelDRAW - это целый программный комплекс, который включает в себя:

- собственно, *CorelDRAW* - редактор векторной графики;
- *CorelPhotoPaint* - редактор растровой графики;
- *CorelCapture* - программу для захвата изображения с экрана компьютера;
- *CorelTrace* - программу для перевода растровой картинке в векторное изображение.

Программа *CorelDRAW* имеет стандартный оконный *интерфейс*



(рис.3.20.1). Интерфейс CorelDraw X7

Элемент	Описание
Набор инструментов	Плавающая панель с инструментами для создания, заливки и изменения объектов на рисунке
Строка заголовка	Область, в которой отображается название рисунка, открытого в данный момент
Строка меню	Область, в которой содержатся раскрывающиеся меню с наборами параметров
Панель инструментов	Перемещаемая панель, содержащая клавиши быстрого вызова меню и других команд
Окно рисования	Область за пределами страницы рисования, ограниченная полосами прокрутки и элементами управления приложением
Панель свойств	Перемещаемая панель с командами, относящимися к активному инструменту или объекту. Например, когда активен инструмент ввода текста, на панели свойств отображаются команды для создания и редактирования текста
Окно настройки	Окно, в котором содержится набор доступных команд и параметров, относящихся к определенному инструменту или задаче
Линейки	Горизонтальные и вертикальные границы, которые используются для определения размера и положения объекта на рисунке
Навигатор документов	Область в левой нижней части окна приложения, в которой содержатся элементы управления для перехода между страницами и добавления страниц
Страница рисования	Прямоугольная область в окне рисования. Это часть рабочей области, доступная для печати
Строка состояния	Область в нижней части окна приложения, в которой содержатся данные о свойствах объекта, например, тип, размер, цвет, заливка и разрешение. В строке состояния показано также текущее положение курсора мыши
Навигатор	Кнопка в правом нижнем углу, при нажатии которой открывается маленькое окно, с помощью которого можно перемещаться по рисунку
Цветовая палитра	Закрепляемая панель, содержащая поля образцов цвета

(табл.3.20.1). Инструменты рабочего пространства

Инструменты рабочего пространства

Команды приложения доступны через *строку меню, панель инструментов, набор инструментов, панель свойств и окна настройки*. *Панель свойств и окна настройки* обеспечивают доступ к командам, относящимся к активному инструменту или текущему заданию. *Панель свойств, окна настройки и набор инструментов* можно открывать, закрывать и перемещать по экрану в любое время.

Многие инструменты рабочего пространства можно настраивать так, чтобы они соответствовали вашим потребностям.

Стандартная панель инструментов

Стандартная панель инструментов, отображаемая по умолчанию, содержит *кнопки и элементы управления быстрого вызова* многих команд меню. Для получения сведений о настройке положения, содержимого и внешнего вида панелей инструментов.

Кнопка	Цель	Кнопка	Цель
	Создание нового рисунка		Открытие рисунка
	Сохранение рисунка		Печать рисунка
	Вырез выделенного объекта в буфер обмена		Копирование выделенных объектов в буфер обмена
	Вставка содержимого буфера обмена в рисунок		Отмена действия
	Импорт рисунка		Восстановление отмененного действия
	Экспорт рисунка		Запуск приложений Corel
	Открытие экрана приветствия		Открытие диалогового окна «Параметры»
	Установка уровня масштабирования		Включение или отключение автоматического выравнивания для сетки, направляющих, объектов и динамических направляющих

(табл.3.20.2). Стандартная панель инструментов

Дополнительные сведения о панелях

Кроме стандартной панели инструментов, в программе *CorelDRAW* имеются панели инструментов для *определенных задач*. Например, на панели инструментов Текст содержатся команды, относящиеся к использованию

инструмента *Текст*. При частом использовании панели инструментов можно указать, чтобы она отображалась в рабочем пространстве *постоянно*.

В следующей таблице описываются панели инструментов, отличные от стандартной панели инструментов.



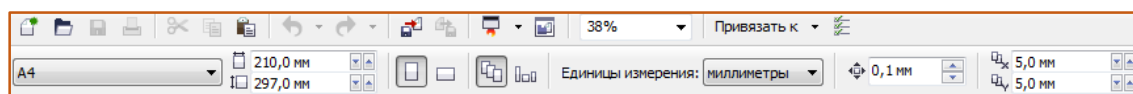
(рис.3.20.2). Панель инструментов для определенных задач

Если в рабочем пространстве по умолчанию щелкнуть раскрывающую стрелку инструмента **Форма**, то откроется раскрывающаяся кнопка *Изменение формы*.

Панель свойств

На панели свойств отображаются часто используемые функции, относящиеся к *активному инструменту* или *выполняемой задаче*. Хотя панель свойств и выглядит как панель инструментов, ее содержимое изменяется в зависимости *от инструмента или задачи*.

Например, при выборе в наборе инструментов инструмента *Текст* на панели свойств отображаются команды, относящиеся к работе с текстом. В примере ниже на панели свойств отображаются инструменты форматирования, выравнивания и редактирования текста.



(рис.3.20.3). Панель свойств

Установка приложений CorelDRAW Graphics Suite

Мастер установки позволяет легко установить приложения и компоненты *CorelDRAW Graphics Suite*. Мастера установки можно использовать для быстрой установки приложений с использованием *параметров по умолчанию* или *специальной установки*, в процессе которой выбираются различные значения параметров.

Мастер установки используется также для выполнения следующих операций:

- *добавление* и *удаление* компонентов в процессе текущей установки;
- исправление текущей установки *путем переустановки* всех функций приложения;

- удаление программы *Coreldraw graphics suite*.

Исправление установки помогает, если возникают проблемы с использованием приложения или имеются признаки повреждения установки. Прежде чем исправить установку, попробуйте выполнить сброс параметров текущего рабочего пространства *до значений по умолчанию*, удерживая нажатой клавишу **F8** во время запуска программы.

Установка приложений, входящих в состав CorelDRAW Graphics Suite.

1. Закройте все приложения, включая все *антивирусные программы*
2. Вставьте диск в дисковод для *компакт-дисков*
3. Если мастер установки не запускается автоматически, нажмите кнопку **Пуск** на панели задач *Windows* и выберите пункт **Выполнить**. Введите **X:\Setup\Setup**, где **X** - это буква, соответствующая дисководу компакт-дисков.
4. Прочтите лицензионное соглашение, а затем установите флажок **я принимаю условия лицензионного соглашения**
5. Нажмите кнопку **Далее**
6. Введите свое имя в поле **Имя пользователя**
7. Введите серийный номер своего продукта в поле **Серийный номер**
8. При вводе серийного номера ни регистр, ни дефисы не учитываются
9. Нажмите кнопку **Далее**.
10. Для установки программного обеспечения следуйте инструкциям мастера установки.

Для запуска мастера установки можно также нажать кнопку **Пуск** на панели задач *Windows®*, выбрать пункт **выполнить** и ввести **D:\autorun**.

Различия цветовых моделей RGB, CMYK, HSB

Цветовая модель - это способ описания цвета с помощью количественных характеристик.

Под цветовой моделью обычно подразумевают термин, который обозначает абстрактную модель описания представления цветов в виде трех- или четырехзначных чисел, называемых цветовыми компонентами (иногда - цветовыми координатами).

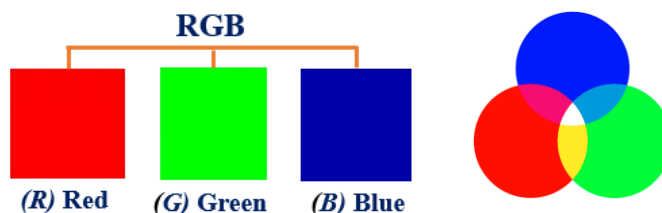
Цветовая модель используется для описания, излучаемого и отраженного цветов. Вместе с методом интерпретации этих данных множество цветов цветовой модели и определяет цветовое пространство.

Цветовая модель RGB

RGB- цветовая модель, названная так по трём заглавным буквам названий цветов, лежащих в ее основе: **Red, Green, Blue**, или *красный, зелёный, синий*. Эти же цвета образуют и все промежуточные. Научное название- аддитивная модель (*от англ. слова add-* «добавлять»). Служит для вывода изображения на экраны мониторов и другие электронные устройства. Обладает большим цветовым охватом.

Цветовая модель RGB предполагает, что вся палитра складывается из светящихся точек. Это значит, что на бумаге невозможно изобразить цвет в цветовой модели RGB, так как бумага поглощает цвет, а не светится. Исходный цвет можно получить, если прибавить к несветящейся- или изначально чёрной- поверхности проценты от каждого из ключевых цветов.

RGB-цвет получается в результате смешения красного, синего и зелёного в разных пропорциях: каждый оттенок можно описать тремя числами, обозначающими яркость трёх основных цветов.

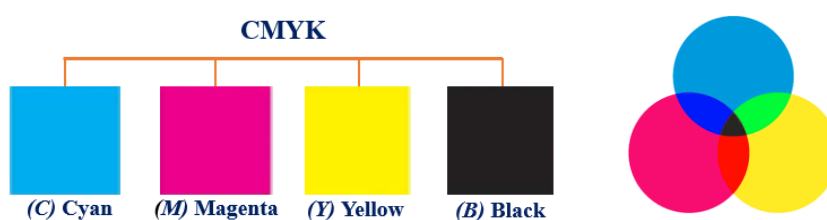


(рис.3.20.4). Цветовая модель RGB

Цветовая модель CMY(K)

Эти три цвета лежат в основе цветовой модели **СМУК**- субтрактивная модель (*от* англ. слова *subtract*- «вычитать»), которая основана на вычитании из белого первичных цветов: голубой цвет вычитает из белого цвета красный, желтый- синий, а пурпурный- зелёный. Модель СМУ(К) используется в полиграфии для стандартной триодной печати и в сравнении с *RGB*-моделью обладает меньшим цветовым охватом. Бумага и другие печатные материалы- это поверхности, которые отражают свет.

Если вычесть из белого три первичных цвета- *RGB*, получаются три дополнительных цвета *СМУ*.



(рис.3.20.5). Цветовая модель СМУК

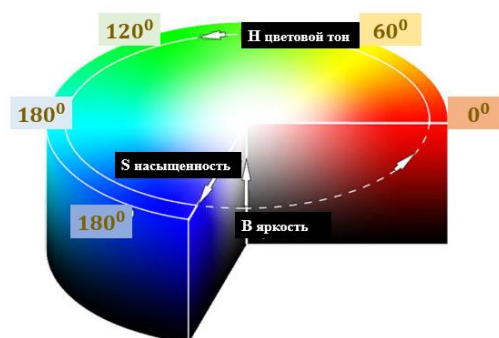
В модель СМУК добавлен дополнительный черный цвет, и на это есть веские причины. В теории при смешении трёх основных цветов должен получиться чёрный цвет. В реальности же в красках есть примеси, и вместо чистого черного получается неопределенный грязно-коричневый. Тем более при печати смешение сразу трёх цветов ради получения черного очень сильно увлажняет бумагу, возрастает риск ее переувлажнения при не всегда идеальных внешних условиях и в силу особенностей самих красителей. Именно поэтому в модель введён чёрный цвет для получения тёмных оттенков и непосредственно самого чёрного. Буква **К** в названии модели СМУК взята у слова **Black**, и она обозначает ключевой цвет- **Key Color**.

Цветовая модель HSB

Перед тем, как подвести итог, подчеркнём: модели *RGB* и *СМУК* не так хорошо соответствуют понятию собственно цвета, как цветовая модель **HSB**. Это аббревиатура с английских слов: **Hue, Saturation, Brightness**- *тон, насыщенность, яркость*. HSB основана на модели RGB, но у неё другая система координат: *каждый цвет в этой модели получается путем*

добавления к основному спектру черной или белой краски. При этом тон- это собственно цвет и есть, насыщенность- процент добавленной к цвету белой краски, а яркость- процент добавленной чёрной краски.

Описание цветов в этой модели не соответствует цветам, воспринимаемых человеческим глазом. Эта модель используется в графических редакторах при настройке палитры цвета. Художники используют её для *тщательного подбора оттенков*.



(рис.3.20.6). Цветовая модель HSB

В чем отличие RGB от CMYK?

Итак, *RGB*- цветовая модель, по которой строятся цвета на экране. Основана на сложении цветов. *CMYK*- цветовая модель, по которой формируется изображение для печати. Основана на вычитании цветов.

Разница между *CMYK* и *RGB* заключается в том, что *RGB-цвет* по сути лишь излучаемый цвет (*или свет*), а *CMYK-цвет*- цвет отражаемый (*краска*). Первый образуется за счёт *интенсивности свечения*, а второй получается, как *результат наложения красок в полиграфии*. Соответственно, любые изображения в электронном виде- рисунки на мониторе компьютера, фотографии на экране телефона- основываются на *RGB-модели*. Модель *CMYK* применяется для полноцветной печати. А чтобы цвета не потерялись, изображение перед печатью выводят из аддитивной модели в *субтрактивную*⁵. Говоря на языке дизайнеров и специалистов подготовки

⁵ *Субтрактивный- это метод синтеза, основанный на вычитании элементов друг из друга.*

макетов, модель СМҮК- рабочий инструмент офсетной типографии, который выводит цвета на бумагу.



Контрольные вопросы и задания

1. В чем заключаются преимущества и недостатки векторной графики?
2. Окно приложения
3. Инструменты рабочего пространства
4. Стандартная панель инструментов
5. Дополнительные сведения о панелях
6. Инструменты
7. Панель свойств
8. Новые возможности *CorelDRAW*
9. Установка приложений *CorelDRAW Graphics Suite*

21-§. СОЗДАНИЕ И УПРАВЛЕНИЕ БАЗОЙ ДАННЫХ (БД).

Ключевые слова: база данных, СУБД, поле, запись, форма, запрос, отчет, ключ, первичный ключ, внешний ключ, фильтр, сортировка и группировка, реляционная база данных.

СУБД- комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными (вставлять, обновлять, удалять и выбирать). Система обеспечивает безопасность, надёжность хранения и целостность данных, а также предоставляет средства для администрирования БД.

Базы данных - служат для хранения и поиска большого объёма информации.

База данных – это структурная информационная модель.

Основные функции СУБД.

- управление данными во внешней памяти (на дисках);

- управление данными в оперативной памяти с использованием *дискового кэша*;
- журнализация изменений, резервное копирование и восстановление базы данных *после сбоев*;
- поддержка языков БД (*язык определения данных, язык манипулирования данными*).

Состав СУБД

Обычно современная *СУБД* содержит следующие компоненты:

- **ядро**, которое отвечает за управление данными во внешней и оперативной памяти и журнализацию;
- **процессор языка базы данных**, обеспечивающий оптимизацию запросов на извлечение и изменение данных и создание, как правило, машинно-независимого исполняемого внутреннего кода;
- **подсистему поддержки времени исполнения**, которая интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с *СУБД*;
- **сервисные программы (внешние утилиты)**, обеспечивающие ряд дополнительных возможностей по обслуживанию информационной системы.

Классификации СУБД - по модели данных.

- *Иерархические*
- *Сетевые*
- *Реляционные*
- *Объектно-ориентированные*
- *Объектно-реляционные*

Популярные системы управления базами данных

Название	Разработчик	Лицензия	Написана на
Oracle	Oracle Corporation	Проприетарная	Assembly, C, C++
MySQL	Oracle Corporation	GPL v2 или проприетарная	C, C++
Microsoft SQL Server	Microsoft Corporation	Проприетарная	C, C++
PostgreSQL	PostgreSQL Global	Development Group	Лицензия PostgreSQL (бесплатное ПО с открытым исходным кодом, либеральная лицензия)

MongoDB	MongoDB Inc.	Различные варианты лицензирования	C++, C, JavaScript
DB2	IBM	Проприетарная EULA	Assembly, C, C++
Microsoft Access	Microsoft Corporation	Пробное ПО	
Redis	Salvatore Sanfilippo	Лицензия BSD	ANSI C

(табл.3.21.1). Системы управления базами данных

Примеры баз данных:

- *Записная книжка*
- *Словари*
- *Справочники*
- *Энциклопедии*

Основные операции работы с базы данных

- *создание БД;*
- *редактирование БД;*
- *просмотр БД;*
- *поиск информации в БД.*

Реляционная база данных

Реляционная база данных представляет собой множество взаимосвязанных *двумерных таблиц* - реляционных таблиц, называемых также отношениями, в каждой из которых содержатся сведения об одной сущности автоматизируемой предметной области. Логическую структуру реляционной базы данных образует совокупность реляционных таблиц, между которыми установлены логические связи.

В таблицах базы должны сохраняться все данные, необходимые для решения *задач предметной области*. Причем каждый элемент данных должен храниться в базе *только в одном экземпляре*. Для создания таблиц, соответствующих реляционной модели данных, используется процесс, называемый *нормализацией данных*.

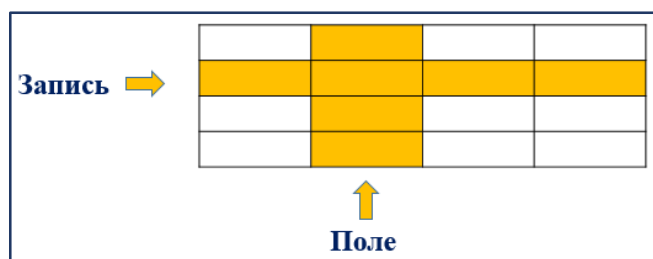
Нормализация - это удаление из таблиц повторяющихся данных путем их переноса в новые таблицы, записи которых не содержат повторяющихся значений.

Структура реляционной таблицы определяется составом полей. Каждое поле отражает определенную характеристику сущности. Для поля указывается

тип и размер элементарного данного, размещаемого в нем, и ряд других свойств. Содержимое поля отображается в *столбце таблицы*. Столбец таблицы содержит данные одного типа.

Содержание таблицы заключено в ее *строках*, однотипных по структуре. Каждая строка таблицы содержит данные о конкретном экземпляре сущности и *называется записью*.

Для однозначного определения (*идентификации*) каждой записи таблица должна иметь *уникальный (первичный) ключ*. По значению ключа таблицы отыскивается единственная запись в таблице. Ключ может состоять из одного или нескольких *полей таблицы*. Значение уникального ключа не может повторяться в нескольких *записях*.



(рис.3.21.1). Структура таблицы реляционной БД.

Логические связи между таблицами дают возможность объединять данные из разных таблиц. Связь каждой пары таблиц задается одинаковыми полями в них - *ключом связи*. Таким образом обеспечивается рациональное хранение недублированных данных и их объединение в соответствии с требованиями решаемых задач.






СУБД Microsoft Access является *системой управления реляционной базой данных*, включающей все необходимые инструментальные средства для создания локальной и общей базы данных в локальной сети (*интернет*), а также для создания приложения пользователя, работающего с этими базами данных.

База данных Access, создаваемая на локальном компьютере, хранит в файле не только все таблицы с данными, но и объекты приложения - *формы, отчеты*, а также *программный код*.

СУБД Access включает разнообразные и многочисленные относительно *автономные инструментальные средства*, ориентированные на создание объектов базы данных и приложений пользователя. Разнообразные мастера в режиме ведения диалога с пользователем позволяют создавать объекты и выполнять разнообразные функции по реорганизации и преобразованию баз данных.

Средства программирования СУБД включают язык структурированных запросов SQL, язык макрокоманд и язык объектно-ориентированного программирования для приложений *Microsoft Visual Basic for Applications (VBA)*. VBA является частью семейства *Microsoft Visual Basic*, которое входит в состав *Visual Studio*.

Средства графического конструирования позволяют создавать объекты базы данных и объекты приложения с помощью многочисленных графических элементов, не прибегая к программированию. Среди многочисленных средств графического конструирования и диалоговых средств Access следует выделить средства для создания:

-  таблиц и схем баз данных, отображающих их связи;
-  запросов выборки, отбирающих и объединяющих данные нескольких таблиц в *виртуальную таблицу*, которая может использоваться во многих задачах приложения;
-  запросов на изменение данных базы;
-  экранных форм, предназначенных для ввода, просмотра и обработки данных в диалоговом режиме;
-  отчетов, предназначенных для просмотра и вывода на печать данных из базы и результатов их обработки в удобном для пользователя виде.

Требования к системе для Access 2013

Access 2013 входит в состав Office 2013 (*в наборы приложений выпусков Профессиональный, Профессиональный Плюс*). Кроме того, можно оформить подписку на *"облачные" версии Office 365* - Access входит в состав следующих

выпусков Office 365: для дома *расширенный*, профессиональный плюс, *корпоративный*.

Для работы *Access 2013* рекомендуется использовать компьютер с процессором архитектуры *x86* или *x64* с набором инструкций **SSE2** и частотой не ниже **1 ГГц**.

Памяти (*RAM*) необходимо не менее *1 Гбайт* (для *32-разрядных систем*) или *2 Гбайт* (для *64-разрядных*). Компьютер должен работать под управлением одной из операционных систем: *Windows 7*, *Windows 8* или *Windows Server 2012*.

При стандартной установке *Access 2013* требуется примерно *3 Гбайт* *свободного дискового пространства*. Часть места на диске будет освобождена после установки при удалении с него исходного *загрузочного пакета*. В процессе установки на диске необходимо иметь дополнительное пространство примерно такого же объема.

Рекомендуется монитор с разрешением не менее *1024×768 пикселей* или с более высоким и с поддержкой *256 цветов*.

Объекты Access

База данных *Access* включает следующие сохраняемые в одном *ACCDB⁶-файле* объекты:

- ✓ *таблицы, запросы, схемы данных*, непосредственно имеющие отношение к базе данных;
- ✓ *формы, отчеты, макросы и модули*, называемые объектами приложения.

Формы и **отчеты** предназначены для типовых процессов обработки данных: *просмотра, обновления, поиска по заданным критериям, получения отчетов*. Эти объекты приложений конструируются из графических элементов, называемых *элементами управления*. Основные элементы

⁶ *ACCDB* – расширение файла

управления служат для отображения полей таблиц, являющихся источниками данных объекта.

Объекты представлены в области навигации окна базы данных Access. Все операции по работе с объектами собственно базы данных и приложений начинаются в этом окне.

❖ **Таблицы** (*Tables*) создаются пользователем для хранения данных об одной сущности - одном информационном объекте модели данных предметной области. Таблица состоит из полей (столбцов) и записей (строк). Каждое поле содержит одну характеристику информационного объекта предметной области. В записи собраны сведения об одном экземпляре информационного объекта. База данных Access может включать до 32768 объектов (*в том числе формы, отчеты и т.д.*). Одновременно может открываться до 2048 таблиц.

❖ **Запросы** (*Queries*). Запросы на выборку служат для выборки нужных данных из *одной или нескольких связанных таблиц*. Результатом выполнения запроса является виртуальная таблица. В запросе можно указать, какие поля исходных таблиц следует включить в запись таблицы запроса и как отобразить нужные записи. Таблица запроса может быть использована наряду с другими таблицами базы при обработке данных. Запрос может формироваться с помощью *конструктора запросов* или *инструкции языка SQL*.

❖ **Схема данных** (*Relationships*) определяет, с помощью каких полей таблицы связываются между собой, как будет выполняться объединение данных этих таблиц, нужно ли проверять связную целостность при *добавлении и удалении записей*, изменении ключей таблиц.

❖ **Формы** (*Forms*) являются основным средством создания диалогового интерфейса приложения пользователя. Форма может создаваться для работы с электронными документами, сохраняемыми в таблицах базы данных. Включаемые в форму процедуры обработки событий позволяют управлять процессом обработки данных в приложении. Такие процедуры

хранятся в модуле формы. В формы могут вставляться *рисунки, диаграммы, звуковые фрагменты, видео*.

❖ **Отчеты** (*Reports*) предназначены для формирования на основе данных базы выходных документов любых форматов, содержащих результаты решения задач пользователя, и вывода их *на печать*. Как и *формы*, отчеты могут включать процедуры *обработки событий*.

❖ **Макросы** (*Macros*) являются программами, состоящими из *последовательностей макрокоманд*, которые выполняются по вызову или при наступлении некоторого события в объекте приложения или его *элементе управления*. Макросы данных выполняются при наступлении некоторого события в исходных таблицах. Макросы позволяют автоматизировать некоторые действия в приложении пользователя. Создание макросов осуществляется в *диалоговом режиме* путем выбора нужных макрокоманд и задания параметров, используемых ими при выполнении.

❖ **Модули** (*Modules*) содержат процедуры на языке *Visual Basic for Applications*. Могут создаваться процедуры-подпрограммы, процедуры-функции, которые разрабатываются пользователем для реализации *нестандартных функций* в приложении пользователя, и процедуры для *обработки событий*. Использование процедур позволяет создать законченное приложение, имеющее собственный графический интерфейс пользователя, позволяющий запросить выполнение всех функций приложения, обработать все ошибки и нестандартные ситуации.

Основные типы данных.

Типы данных	Описание
Краткий текст	Одна строка текста до 255 символов
Длинный текст	Текст, состоящий из нескольких строк, который можно посмотреть при помощи полос прокрутки (<i>до 65536 символов</i>)
Числовой	Число любого типа (<i>можно использовать в вычислениях</i>)
Денежный	Поле, содержащее дату или время
Дата и время	Поле, содержащее дату или время
Логический	Содержит одно из значений <i>True (истина)</i> или <i>False (ложно)</i> и применяется в логических операциях

Поле объекта OLE	Содержит рисунки, звуковые файлы, таблицы Excel, документ Word и т.д.
-----------------------------	---

(табл.3.21.2). Основные типы данных.

Шаблоны баз данных.

После запуска Access отображается *стартовое окно Access 2013*, где можно управлять файлами. В момент открытия представлен список последних использованных баз данных, команда **открыть другие Файлы** (*Open Other Files*) баз данных, в области создания баз данных предоставлена возможность создать новую базу: **Пустая база данных рабочего стола** (*Blank Desktop Database*) и Пользовательское веб-приложение (*Custom Web App*).



(рис.3.21.2). Стартовое окно Access 2013

Шаблоны типовых баз данных включают все *необходимые таблицы, формы, запросы и отчеты* для предметных областей различных сфер деловой и личной жизни.

Средства программирования.

Наряду с диалоговыми средствами создания *объектов базы данных и объектов приложения*, которые позволяют решить многие задачи пользователя, Access располагает мощными *средствами программирования*. Эти средства могут использоваться как для *доработки приложений*, созданных диалоговыми средствами, так и для реализации сложных задач и создания приложений с необходимым пользователю интерфейсом в целом. Без использования программного кода практически невозможно получить законченное автоматизированное приложение пользователя.

Одним из средств программирования в Access является *язык макрокоманд*. Программы, созданные на этом языке, называемые *макросами*, позволяют легко связывать отдельные действия, реализуемые с помощью *форм, запросов, отчетов*.

Наряду с языком макрокоманд Access включает развитую интегрированную среду *объектно-ориентированного программирования Visual Basic for Applications (VBA)*, позволяющую реализовать любые программные решения. Программы на VBA реализуются процедурами, которые объединяются в объектах, называемых *модулями*.



Контрольные вопросы и задания

1. Понятие системы управления базы данных
2. Функции СУБД
3. Основные операции работы с базой данных.
4. Функции система управления базы данных (СУБД):
5. Основные *типы данных*
6. Какие объекты составляют базу данных Access?
7. Чем определяется структура реляционной таблицы?
8. Для чего предназначен первичный ключ реляционной таблицы?
9. В файле с каким расширением сохраняется база данных Access 2013?

22-§. ПРОГРАММЫ ПЕРЕВОДЧИКИ

Ключевые слова: *компьютерные словари, программы переводчики, программы распознавания документов, перевод, машинный перевод, автоматизированный перевод, онлайн переводчики, электронный словарь, система, коллекция, неточность, эвристический анализ.*

Среди многочисленных сложных проблем, которые изучает *современное языкознание*, важное место занимает изучение лингвистических аспектов межъязыковой речевой деятельности, которую называют «*переводом*» или «*переводческой деятельностью*».

С самого начала перевод выполнял важнейшую социальную функцию, делая возможным *межъязыковое общение людей*. Распространение письменных переводов открыло людям широкий доступ к культурным достижениям других народов, сделало возможным взаимодействие и взаимообогащение *литератур и культур*.

Перевод – это сложное многогранное явление, отдельные аспекты которого могут быть предметом исследования разных наук.

В рамках переводоведения изучаются *психологические, литературоведческие, этнографические* и другие стороны переводческой деятельности, а также история переводческой деятельности в той или иной стране или странах.

В настоящее время главная функция перевода *информативная* или *коммуникативная*, поэтому перевод-это средство обеспечить возможность общения (*коммуникации*) между людьми, говорящими на разных языках. Поэтому для теории перевода в настоящее время особое значение имеют данные коммуникативной лингвистики об особенностях процесса речевой коммуникации, специфике прямых и косвенных речевых актов, о соотношении выраженного и подразумеваемого смысла в высказывании и тексте, влиянии контекста и ситуации общения на понимание текста, других факторах, определяющих коммуникативное поведение человека.

Перевод в своем развитии прошел несколько этапов был перевод и *дословный*, был и период когда авторы при переводе оригинала получали новое произведение, в настоящее же время индивидуально - авторский стиль не так существенен. Известно, что полное тождество между оригиналом и переводом не возможно. Языковое своеобразие любого текста, ориентированность его содержания на определенный языковой коллектив, обладающий лишь ему присущими *«фоновыми»* знаниями и культурно-историческими особенностями, не может быть с абсолютной полнотой *«воссоздано»* на другом языке. Отсутствие тождественности отнюдь не

мешает переводу выполнять те же коммуникативные функции, для выполнения которых был создан текст оригинала.

В настоящее время при переводе авторы часто используют реалии (*обозначаются бытовые и специфические слова и обороты, не имеющие эквивалента в быту, а, следовательно, и на языках других стран*).

Программы-переводчики, средства проверки *орфографии* и *распознавания текста* включают:

- программы-переводчики, предназначенные для создания подстрочника исходного текста на указанном языке;
- словари орфографии, используемые при проверке текстов;
- словари синонимов, используемые для стилевой правки текстов;
- программы для распознавания считанной сканерами информации и преобразования в текстовое представление.

Машинный перевод

Машинный перевод – *процесс перевода текстов (письменных, а в идеале и устных) с одного естественного языка на другой с помощью специальной компьютерной программы*

Принято считать, что мировая история машинного перевода началась с развитием компьютеров, однако идея создания механизма, способного осуществлять перевод текста с одного языка на другой, появилась еще в *первой половине XVII века*. Появление компьютеров позволило начать *воплощение этих проектов* в жизнь.

Современные технологии машинного перевода далеко ушли от первых попыток *"научить"* компьютер переводить текст. Сегодня системы машинного перевода по праву принадлежат к классу систем *искусственного интеллекта*, поскольку выполняют отдельные функции интеллекта человека: они конструируют текст на выходном языке на основе входного текста,

пользуясь сводом определенных правил, заданных в виде структур данных и алгоритмов.

Классификация систем перевода

1. Системы машинного перевода (МП)

Системы машинного перевода - программы, осуществляющие полностью *автоматизированный перевод*. Главным критерием программы является *качество перевода*. Кроме этого, для пользователя важными моментами является удобство интерфейса, лёгкость интеграции программы с другими средствами обработки документов, выбор тематики, утилита пополнения словаря. С появлением Internet основные поставщики систем машинного перевода включили в свои продукты *Web-интерфейсы*, обеспечив при этом их интеграцию с остальным *программным обеспечением и электронной почтой*, что позволило применять механизмы МП для перевода *Web-страниц, электронной корреспонденции и онлайн-разговорных сеансов*.

2. Системы перевода с функцией *Translation Memory (TM)*

Предоставляют средства для так называемого *Machine Assisted Human Translation (МАНТ)* - перевода, выполняемого человеком с помощью машины. Первоначальное предназначение систем TM - облегчение работы переводчиков при локализации программных продуктов и создании терминологических баз данных, в дальнейшем системы стали развиваться как вспомогательный инструмент переводчика. Особое внимание уделяется поддержке форматов всевозможных процессоров.

3. Контролируемый язык и машинный перевод на основе базы знаний

В системах на основе контролируемого языка реализован переход от свободного входного языка к контролируемому входному языку. Контроль входного языка предусматривает определённые ограничения *лексики, грамматики, семантики*. Контролируемый входной язык используется для упрощения выражений исходного текста, чтобы повысить качество перевода.

Современные программы – переводчики

1. Онлайн переводчики

Службы онлайн перевода выполняют перевод непосредственно в окне Web-браузера, не требуя установки *программы-переводчика* на компьютер пользователя.

Как правило, существует ограничение на объем вводимого текста. Также существует различие в количестве языков, на которые переводчик способен переводить и с которых.

Онлайн перевод с *Английского - English:*

- ✚ Компания **PROMT** Текст 500/2000 знаков, web.
- ✚ **Babel Fish Translation** Текст 150 слов, web.
- ✚ **Free2Professional** Текст 750 слов.
- ✚ **Google Переводчик** Текст ? слов, web.
- ✚ **Im Translator** Текст 1000 знаков.
- ✚ **Pragma он-лайн перевод** Текст 1 Кб.
- ✚ **ProLingOffice** Текст 750 знаков.
- ✚ **ProLinGOnline** Текст 500 знаков.

2. Онлайн словари.

Существует возможность поиска, есть строка ввода и меню для выбора входного и выходного языка, среди словарей - *SDL-Trados MultiTerm, Langenscheidt's New College Dictionary online, ABBYY Lingvo* и т.д.

3. Услуги по переводу в Интернете

Практически каждая *компания-производитель систем машинного перевода* предлагает услуги по переводу. Чаще всего пользователю представляется возможность выбора: *получить только машинный перевод*, либо машинный перевод с последующим редактированием профессиональным переводчиком, либо перевод, выполненный профессиональным переводчиком. Такого рода сервис можно встретить по адресу *<http://www.plustranslation.com>* (сервис компании *Transparent Language*).

Подобные услуги есть у *Systran, L&H, Langscheidt, Language Engineering Corporation*.

Перечень онлайн-переводчиков и словарей

✚ Ace Translator 8.3.2.503

Ace Translator - интернет переводчик который позволяет быстро и качественно переводить как любой сложности *тексты*, так и *Веб-страницы*, *электронные письма*, *чаты* и т.д. Программа имеет многоязычный интерфейс и поддерживает перевод около *58 языков мира*.

✚ **Pragma 6.0.101.11 Business + Словари 6.0.100.10** - это многоязычная программа для *перевода текстовых документов* с одного языка на другой. Программа поддерживает семь языков: *английский, русский, украинский, немецкий, латышский, польский и французский*. Pragma выполняет перевод непосредственно в окне активного приложения или в отдельном окне быстрого перевода.

✚ QDictionary v1.6 Full Portable (134 словаря) (2010/Rus)

Быстрый *электронный словарь - переводчик*. Имеет возможность переводить фразы а также выражения простым наведением на них курсора. Функционирует с *Internet Explorer, Outlook Express* на основе Explorer. В остальных приложениях (*Firefox, Opera а также др.*) есть возможность переводить текст двойным щелчком мыши или другими "*горячими*" клавишами.

✚ Lingvo 12 12.0.0.356 (RU)

Lingvo 12 (*многоязычная версия*). Двенадцатая по счету версия самого популярного продукта компании *ABBYY-Lingvo*, в очередной раз доказывает, что больше не существует трудностей в переводе с иностранных языков. Предоставляя множество словарей и дополнительные функции, помогающие освоить иностранный язык, *Lingvo 12* станет нашим лучшим помощником в переводе.

Lingvo OnLine! позволяет переводить слова с английского на русский и обратно с помощью одного из двух сервисов – *lingvo.yandex.ru* или *lingvo.ru*.

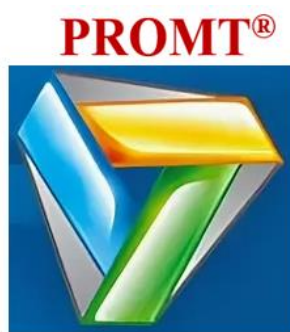
PROMT Freelance 9.0.410

PROMT Freelance 9.0 – Инструмент для автоматизации процесса *перевода текстов*, предназначенный для переводчиков, работающих вне офиса. Лингвистический редактор *"Переводчик PROMT"* – основа *PROMT Freelance 9.0*, позволяющая выполнить перевод любой сложности. Благодаря обновлению алгоритмов и существенному расширению словарей (*в них добавлено более 300000 новых слов*) перевод высокого качества можно получить уже при работе с базовым комплексом настроек системы.

Prompt NET Expert 8.5 Server + Expert 8.5 Giant + 4U 8.5 Giant + Снеж. словари.

PROMT Expert 8.5 – мощная система для профессионального перевода документов. Интегрируется с профессиональной системой *Translation Memory SDL TRADOS 2007*, а также встраивается в *MS Office 2003-2007*, популярные web-браузеры *Internet Explorer* и *FireFox*, электронную почту, *PDF-документы* и *ICQ*.

Компания PROMT является одной из старейших российских **IT компаний**, с 1991 года успешно развивая *технологии машинного перевода*.



(рис.3.22.1). Переводчик PROMT

GoldenDict 0.9.0 Rus Portable

GoldenDict – словарь для перевода слов с одного языка на другой. Поддерживает форматы *StarDict*, *Babylon*, *Lingvo .dsl*, *dictd*, имеет систему морфологии, всплывающее окошко и многое другое. *GoldenDict* – Удобный в использовании электронный многоязычный словарь с функцией интеграции с онлайн-сервисами.

TranslateIt! 8.0 build 7 ML RUS

TranslateIt! – контекстный *англо-русский, русско-английский, немецко-русский и испанско-русский* словарь. Ненавязчивый помощник, с помощью которого можно читать новости и почту, просматривать сайты на иностранном языке, работать с нерусифицированными справками и программами.

Dictor 3.05

Dictor является неофициальным клиентом *Google-переводчика*. По сути онлайн переводчик Dictor то же самое, что и переводчик Google с одним важным отличием. Чтобы перевести какое-либо слово в онлайн переводчике dictor достаточно его выделить и нажать **Ctrl + Alt**, чтобы получить перевод во всплывающем окне. Программа бесплатна, полезна и удобна – не требует никаких дополнительных словарей. Также переводчик Dictor знает практически все языки, а точнее 42, на котором общается весь мир.

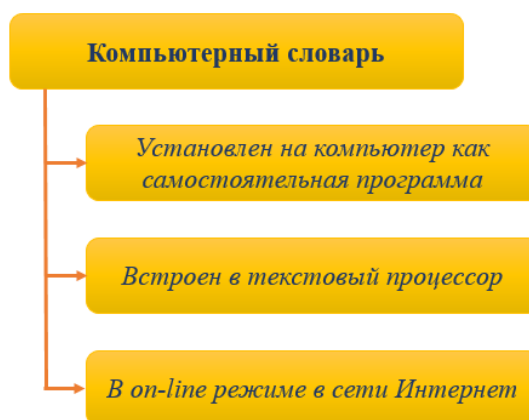
Client for Google Translate PRO v4.4.360

Client for Google Translate – очень удобная и практичная программа, которая моментально переводит выделенный в окне браузера текст с более 50 языков. Client for Google Translate работает также в приложениях Windows таких как Outlook, MS Word и пр. После установки программы появится иконка *Client for Google Translate* в системе и каждый раз, когда есть необходимость перевести некоторый текст, все что нужно сделать, это выбрать (*выделить*) текст и все будет переведено мгновенно!

Коллекция дополнительных словарей для **ABBYY Lingvo 12**

ABBYY Lingvo – без преувеличения лучшая программа в линейке компьютерных словарей на сегодняшний день. Коллекция дополнительных словарей для ABBYY Lingvo 12 поможет пользователю более полно использовать все возможности этой замечательной программы. В архиве 43 *дополнительных словаря*.

Компьютерные словари *выполняют перевод отдельных слов и словосочетаний*.



(рис.3.22.2). Виды компьютерных словарей

✚ **Babylon Pro 8.0.5 (r2) Rus**

Babylon - словарь, позволяющий переводить слова с одного языка на другой. Словарь Babylon имеет более *75 языков* для перевода. Работать со словарем очень просто – после нажатия кнопкой мыши на интересующее слово появляется окно с его переводом. *Babylon* прост в использовании, доступ к *1300 словарям и энциклопедиям*, проверка правильности написания, пересчет валют, времени и физических величин посредством одного щелчка, широкий спектр тем: *наука, техника, спорт, медицина, военное дело, аппликатура аккордов* и т.д, вывод переведенного текста в разных вариантах перевода подходящего к предложению.

✚ **Socrat Personal-4.1.1 portable**

Сократ персональный - Неплохой и компактный переводчик. Это система автоматического перевода *с английского языка на русский и наоборот*, предназначенная для работы со сравнительно небольшими текстами общей тематики. Перевод почтовых сообщений. Различные варианты ввода текста. Низкие системные требования. Перевод стандартных помощников других программ.

При переводе произведений главная задача *программ - переводчиков* - использовать все знание теоретических основ перевода для передачи коммуникативной функции оригинала, так как знание теоретических основ перевода и экстралингвистических реалий необходимое условие адекватности перевода.

Недостатком программ - переводчиков является неточность перевода слов, имеющих несколько значений. Для более адекватного перевода в перспективе можно предложить более глубокий эвристический анализ грамматического построения предложения, с улучшением качества перевода различных частей речи и их грамматических характеристик, а так же исключить конфликт словарей при переводе специализированных текстов.

Электронные переводчики справляются с переводом слов во множественном и единственном числе, но имеется определенная трудность в переводе падежей и постановки глаголов в нужное число. Это объясняется различной интерпретацией падежей в *русском и английском языках*: в русском - через окончание, в английском - через предлоги.

Компьютерный перевод лишен человечности, поэтому никогда не сможет занять *главенствующее место* при переводе как художественных, так и технических текстов. При компьютерном переводе существует нехватка образности и полноты высказывания.



Контрольные вопросы и задания

1. Системы машинного перевода
2. Назначение *программ-переводчиков*.
3. Дайте определение программы *PROMT*.
4. В каких форматах допустимо сохранять переведенный текст *программой PROMT*?
5. Назначение программы *PROMT*.
6. Как можно поменять настройки *программы PROMT*?
7. Как перевести содержимое готового файла?
8. Как прослушать звучание оригинала и перевода?
9. Как получить синхронный перевод текста?
10. *Web Translator*. Назначение программы, *функциональные возможности, недостатки*.

11. TranslatIt. Назначение программы, *функциональные возможности, недостатки.*

12. Lingvo OnLine. Назначение программы, *функциональные возможности, недостатки.*

13. Назначение словарей-переводчиков.

14. Назначение программы *ABBYY Lingvo.*





ПРАКТИЧЕСКИЕ РАБОТЫ

23-§. АВТОМАТИЗАЦИЯ РЕДАКТИРОВАНИЯ И ФОРМАТИРОВАНИЯ ТЕКСТА

Краткая теоретическая информация:







Редактирование - это изменение содержания документа.

К операциям редактирования относятся следующие действия:

-  набор текста;
-  исправление опечаток;
-  копирование, перестановка, удаление частей текста;
-  вставка рисунков, таблиц и других информационных объектов.

Форматирование - это изменение внешнего вида документа и его отдельных частей с целью повышения их удобочитаемости.

К операциям форматирования относятся различные способы выделения текста, а именно:

-  изменение свойств символов;
-  изменение свойств абзацев;
-  оформление заголовков и подзаголовков;
-  преобразование текста в список;
-  преобразование текста в табличный вид;
-  вставка колонтитулов, номеров страниц и пр.

Форматирование текста

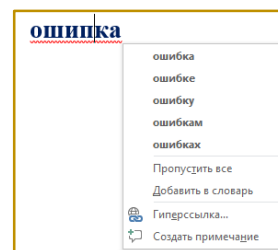
Параметры страницы	Команда Разметка страницы - Параметры страницы	
Изменение шрифта	Кнопка на панели инструментов «Шрифт»	
Изменение размера шрифта	Кнопка на панели инструментов «Шрифт»	
Изменение начертания шрифта	Кнопки на панели инструментов «Шрифт»: полужирный , <i>курсив</i> , <u>подчеркнутый</u>	
Выравнивание текста	Кнопки на панели инструментов «Шрифт»: по левому краю, по центру, по правому краю.	
Изменение цвета текста	Кнопки на панели инструментов «Шрифт»:	
Отступ	<ol style="list-style-type: none"> 1. Команда Главная - Абзац (указать количество см отступа) 2. Использовать клавишу Tab клавиатуры 3. Линейка форматирования <p>Отступ первой строки</p> <p>Отступ всех строк кроме первой строки</p> <p>Отступ всего выделенного фрагмента</p>	

(табл.3.23.1). Форматирование текста

Проверка орфографии (F7).

Одним из важных качеств текста является отсутствие грамматических ошибок. По умолчанию *Microsoft Word* проверяет орфографию и грамматику автоматически при вводе текста, выделяя возможные орфографические ошибки *красной волнистой линией*, а возможные грамматические ошибки - *зеленой волнистой линией*. Система проверки орфографии по умолчанию включена всегда. Кроме того, для проверки наличия синонимов можно использовать *тезаурус*.

Если слово подчеркнуто красной линией, значит оно отсутствует в *орфографическом словаре системы* проверки правописания. Щелкните на нем правой кнопкой и выберите в контекстном меню *правильный вариант*, предлагаемый для замены.



✚ Для проверки орфографии необходимо установить курсор в то место, откуда необходимо начать проверку, и воспользоваться командой *Правописание* на вкладке *Рецензирование*.

✚ Система автоматически начнет проверку. При обнаружении ошибки откроется диалоговое окно, в котором будут предложены варианты исправления ошибки. Для того чтобы исправить ошибку, необходимо нажать кнопку *Изменить*.

Автозамена и Автотекст.

Для автоматизации ввода и исправления текста в среде *Microsoft Word* существуют инструменты *Автозамена* и *Автотекст*. Бывает, что при вводе текста с клавиатуры вместо нужной клавиши нажимается соседняя или две буквы, нажимаются в обратном порядке. Инструмент *Автозамена* имеет встроенный словарь наиболее типичных опечаток и ошибочных написаний.

При обнаружении таких опечаток слово автоматически заменяется на правильное. Словарь *автозамены* можно пополнять.

Практически у каждого пользователя есть свои особенности набора и «индивидуальные» опечатки и ошибки. Если в процессе набора вы ввели слово с опечаткой, то можно не только исправить его, но и включить в словарь *автозамен*. Для этого в контекстном меню следует выбрать команду *Автозамена*.

Инструменты *Автотекст* и *Автозамена* можно использовать для быстрого ввода стандартных фраз по нескольким первым буквам.

Функция Автотекст - это очень удобный способ экономии времени при работе в программе Microsoft Word.

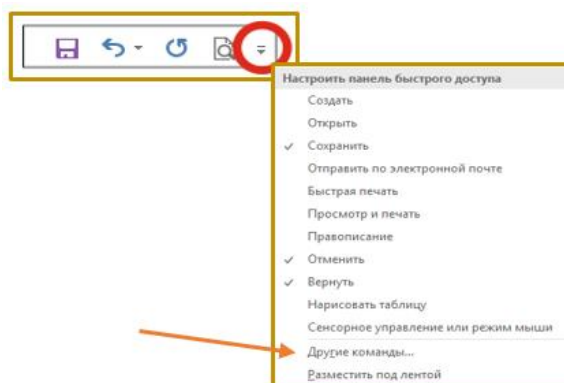
Чтобы добавить автотекст в *Ms. Word 2013*, необходимо перейти на закладку «Вставка», далее «Экспресс блоки», «Автотекст» или нужно добавить его на панель быстрого доступа. Затем легко можно создавать и вставлять свои собственные элементы *автотекста*.

Одним нажатием кнопки можно вставить часто используемые фразы или имена в документы. Например, можно использовать автотекст, чтобы вставить *приветствие, адрес* или *подпись* в письмах.

Задание: Добавьте иконку Автотекста на «панель быстрого доступа».

Порядок выполнения работы:

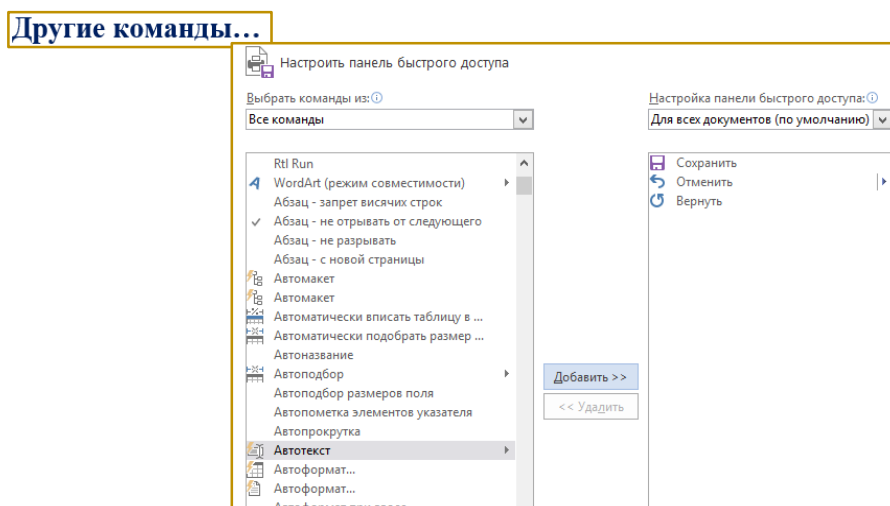
1. Нажмите на кнопку *Настройка панели быстрого доступа*. Откроется выпадающее меню, в котором можно проставить галочки на тех командах, которые необходимы для быстрого запуска.



(рис.3.23.1). *Настройка панели быстрого доступа*

2. Необходимо обратить внимание на два последних пункта. Пункт «*Другие команды*» открывает окно *Параметры Word*, в котором можно детально настроить *Панель быстрого запуска*.

3. Выберите иконку необходимой команды и нажмите кнопку **Добавить**. Иконка переместится в правое поле. Кнопочками с маленькими стрелками можно переместить иконки вверх или низ.



(рис.3.23.2). *Настройка панели быстрого доступа*

4. Этим изменяется очередность расположения *Команд (иконки)* на «*Панели быстрого доступа*». Иконки можно импортировать из *других программ* или *Интернета*.

5. Под левым полем можно установить галочку на *разместить панель быстрого доступа* под лентой. Эта функция хороша, когда на панели будет много иконок. Нажать кнопку **ОК** при выходе из настроек.

6. Так же можно добавить команды непосредственно из ленты. Для этого необходимо щелкнуть на любой команде правой кнопкой мыши и выбрать пункт *добавить* на панель быстрого доступа в контекстном меню.

Результат:



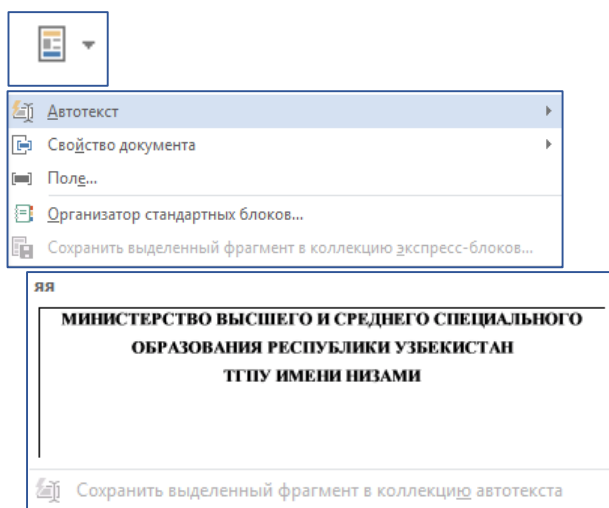
Инструмент *Автотекст* содержит список фраз длиной до 32 символов, которые среда автоматически предлагает вставить, когда набраны первые несколько букв. Эти фразы можно выбирать из списка элементов *автотекста*. Кроме того, в этом списке содержатся элементы для вставки служебной информации, которая, как правило, вставляется в колонтитул, например, *имя автора, дата создания, дата печати, имя файла*.

Чтобы добавить автотекст в *Ms. Word 2013* переходим на закладку «*Вставка*», далее «*Экспресс блоки*», «*Автотекст*».

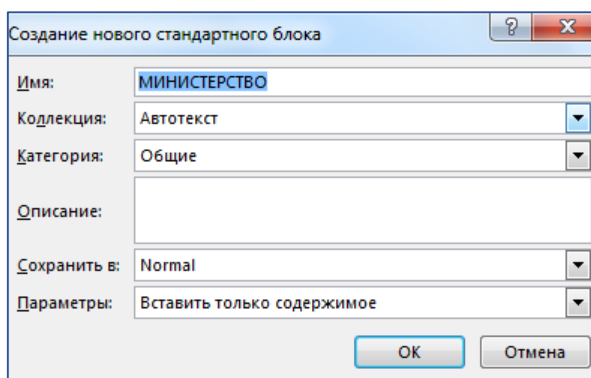
1. Ввести название организации в *Ms. Word*.

Например,

**МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО СПЕЦИАЛЬНОГО
ОБРАЗОВАНИЯ РЕСПУБЛИКИ УЗБЕКИСТАН
ТГПУ ИМЕНИ НИЗАМИ**



2. Выделить текст
3. Нажать «Вставка /Экспресс блоки /Автотекст/ Сохранить выделенный фрагмент в коллекцию автотекста».
4. Нажать **ОК**.



5. И теперь, когда будет необходимо печатать название организации, всегда можно будет увидеть подсказку и сразу вставить предложение без ошибок.

Редактирование и удаление записей автотекста.

Редактирование. Для редактирования свойств автотекста, таких как *имя, категория, описание*, нажмите правой кнопкой мыши по записи автотекста выбрать «Изменить свойства...». Чтобы изменить или переделать сам автотекст, его нужно вставить, изменить, а затем сохранить его, так же как новую запись, но с таким же именем. И программа перезапишет старый автотекст на новый.

Удаление. Чтобы *удалить запись автотекста*, нужно щелкнуть по нему правой кнопкой мыши и выберите «*Упорядочить и удалить...*». Откроется окно «*Организатор стандартных блоков*», в нем выделить нужную запись автотекста и нажать кнопку «**Удалить**». Обязательно необходимо подтвердить удаление, затем нажать «**Заккрыть**», чтобы закончить.

Задание:

А. На вкладке **Разметка страницы** в группе **Параметры страницы** выберите команду **Поля - Настраиваемые поля** в диалоговом окне этой команды задайте верхнее и нижнее поля равным 2 см, левое – равным 3,5 см, правое – 1 см.

В. Напечатать текст. Проверить правописание этого фрагмента средствами *MS. Word*. Если есть ошибки, исправить их в процессе ввода текста с помощью контекстного меню.



Контрольные вопросы и задания

1. Основное назначение *текстового редактора*?
2. Что такое *редактирование* текстового документа?
3. Что такое *форматирование* текстового документа?
4. Расскажите последовательность действий для создания «*автотекста*».
5. Опишите, как обеспечить, чтобы в документе во всех местах после *закрывающейся скобки* стоял пробел.
6. Перечислите наиболее характерные *ошибки* при вводе текста. Как наиболее эффективно их исправлять?
7. В документе в каждом абзаце вместо отступа первой строки введено по пять символов пробела. Опишите, *как удалить эти символы*.

24-§. ОСНОВЫ РАБОТЫ В MICROSOFT EXCEL

Цель работы: *Формирование практических навыков работы с электронными таблицами MS. Excel.*

Краткая теоретическая информация:

Microsoft Excel относится к классу программ, называемых электронными таблицами. Электронные таблицы ориентированы прежде всего на решение экономических и инженерных задач и позволяют систематизировать и обрабатывать данные из любой сферы деятельности.

Термины и понятия, используемые в Excel.

Книга– это файл, используемый для обработки и хранения данных, состоящий из отдельных листов.

Лист– разделен на строки и столбцы и служит основой для выполнения вычислений. Рабочий лист состоит из 16384 столбцов и 1048576 строк.

Строка– горизонтальный набор ячеек, заголовки столбцов– A, B, C, ..., XFD.

Столбец– вертикальный набор ячеек, заголовки строк – 1, 2, 3, ..., 1048576.


Ячейка– минимальный объект табличного процессора.

Адрес ячейки буквенно-цифровой код, который определяется названием (номером) столбца и номером строки (A1, F123, AC72).

Адрес диапазона (блока) ячеек- определяется адресом верхней левой и нижней правой ячейки, разделенных двоеточием (:), например, B2:C7.

Текст– последовательность букв, иногда цифр или некоторых специальных символов.

Числа могут включать цифры и различные символы: знак процента, знак мантиссы, круглые скобки, денежные обозначения, разделители и др., например, 5, -3, 14.

Формула– это конструкция, начинающаяся со знака , состоящая из математических операторов, значений, ссылок на ячейки и имен функций, при этом результатом выполнения формулы является некоторое новое значение.

Диаграмма– графический способ представления числовых значений. Программа Excel поддерживает работу с разными диаграммами: *гистограммы, круговые, пузырьковые, графики* и др.

Главный элемент пользовательского интерфейса *Microsoft Excel 2013* представляет собой *ленту*, которая идет вдоль верхней части окна каждого приложения.



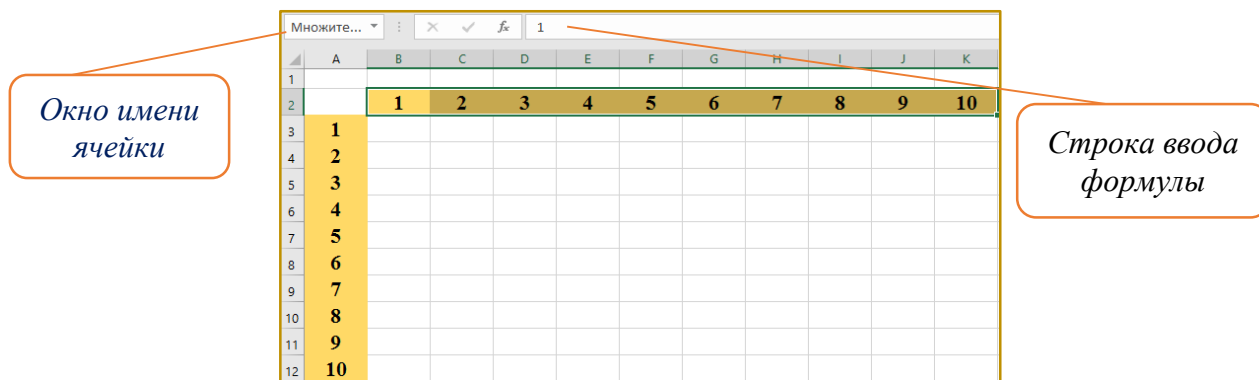
(рис.3.24.1). Лента

Практическое задание 1.

Создайте таблицу умножения до 10, где результат умножения должен находиться на пересечении строки и столбца.

Выполнение работы:

1. Запустите программу **Excel** (*Пуск Все программы Ms. Excel*).
2. Заполните ячейки **B2:K2** и **A3:A12** цифрами от 1 до 10, как показано на рисунке ниже.



(рис.3.24.1). Диапазон ячеек

3. Присвойте диапазону ячеек B2:K2 имя *Множитель1*. Для этого выделите диапазон ячеек B2:K2, в окне имени ячейки введите *Множитель1* (*без пробелов*) и нажмите клавишу **Enter**.

4. Присвойте диапазону ячеек A3:A12 имя *Множитель2*.

5. Вся таблица умножения работает на одной формуле. Для занесения формулы выделите диапазон ячеек B3:K12, в строку формул введите = *Множитель1*Множитель2* и нажмите клавиши **Ctrl+Shift+Enter**.

6. На экране появится искомая таблица умножения. Формула для каждой ячейки примет вид $\{=Множитель1*Множитель2\}$.

7. Выполните заливку таблицы умножения различными цветами.

8. Выделите всю область таблицы и установите все границы кнопкой панели инструментов.

9. Дважды щелкните на ярлычке *текущего рабочего листа* и дайте этому рабочему листу имя *Таблица умножения*.

10. Результаты выполнения задания покажите преподавателю.

Результат:

	A	B	C	D	E	F	G	H	I	J	K
1											
2		1	2	3	4	5	6	7	8	9	10
3	1	1	2	3	4	5	6	7	8	9	10
4	2	2	4	6	8	10	12	14	16	18	20
5	3	3	6	9	12	15	18	21	24	27	30
6	4	4	8	12	16	20	24	28	32	36	40
7	5	5	10	15	20	25	30	35	40	45	50
8	6	6	12	18	24	30	36	42	48	54	60
9	7	7	14	21	28	35	42	49	56	63	70
10	8	8	16	24	32	40	48	56	64	72	80
11	9	9	18	27	36	45	54	63	72	81	90
12	10	10	20	30	40	50	60	70	80	90	100

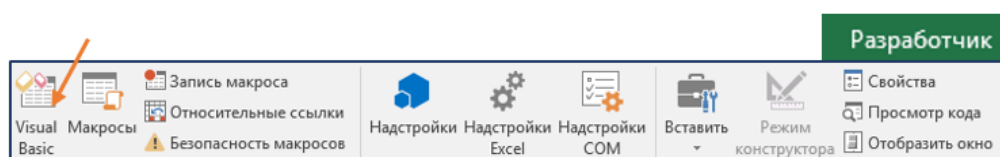
(рис.3.24.2). Результат работы

Практическое задание 2.

Сумма ячеек по цвету.

В *Microsoft Excel*, при создании таблиц, очень удобно и наглядно помечать ячейки цветом. Но что делать, если понадобится суммировать значения ячеек *одного цвета*? По умолчанию, данной функции в Excel до сих пор нет. Решить эту проблему можно написав макрос в **Visual Basic**, которая позволит нам суммировать ячейки с определенным цветом.

Для этого в Excel версии 2013 необходимо перейти на вкладку **Разработчик (Developer)** и нажать **Visual Basic**.



(рис.3.24.3). Вкладка Разработчик

Если такой вкладки нет, то нужно включить ее следующим образом: **Файл-Параметры-Настройка ленты** и поставить галочку напротив слова **Разработчик**.

В открывшемся окне редактора нажать **Insert/Module** (**Вставка/Модуль**) и скопировать туда следующий текст:

```
1 Public Function SumByColor(DataRange As Range, ColorSample As Range) As Double
2   Dim Sum As Double
3   Application.Volatile True
4
5   For Each cell In DataRange
6     If cell.Interior.Color = ColorSample.Interior.Color Then
7       Sum = Sum + cell.Value
8     End If
9   Next cell
10  SumByColor = Sum
11 End Function
```

Сохранить изменения. Далее вернуться в Excel, на вкладке **Формулы** выбрать пункт **«Вставить функцию»**. Необходимо выбрать категорию **Определенные пользователем** – тогда отобразится нужная функция – **SumByColor**. Вставить ее на лист.

	A	B	C	D	E	F
1	5					
2	15			95		
3	25					
4	35					
5	45					
6	55					

(рис.3.24.4). Функция – *SumByColor*

У этой функции 2 аргументы:

1. **DataRange**– диапазон раскрашенных ячеек с числами;
2. **ColorSample**– ячейка, цвет которой принимается как образец для суммирования.



(рис.3.24.4). Функция – SumByColor

Цвет шрифта

Легко изменить функцию, чтобы она учитывала не цвет *заливки фона*, а цвет *шрифта ячейки*. Для этого в строке **6** просто заменить свойство **Interior** на **Font** в обеих частях выражения.

Количество вместо суммы

Если нужно подсчитывать не сумму покрашенных определенным цветом ячеек, а всего лишь *их количество*, то функция будет еще проще. Нужно заменить в ней **7**-ю строку на:

$$Sum = Sum + 1$$

Нюансы пересчета

К сожалению, *изменение цвета заливки* или *цвета шрифта ячейки*, Excel не считает изменением ее содержимого, поэтому не запускает пересчет формул. То есть, при перекрашивании исходных ячеек с числами в другие цвета итоговая сумма по данной функции пересчитываться не будет. Полностью решить эту проблему невозможно, но можно ее существенно облегчить.

Для этого в третьей строке указанной функции используется команда **Application.Volatile True**. Она заставляет Excel пересчитывать результаты данной функции при изменении любой ячейки на листе (*или по нажатию F9*). Следует помнить о том, что данная функция перебирает все (*и пустые тоже*) ячейки в диапазоне.

Практическое задание 3.

Работа с макросами в Microsoft Excel.

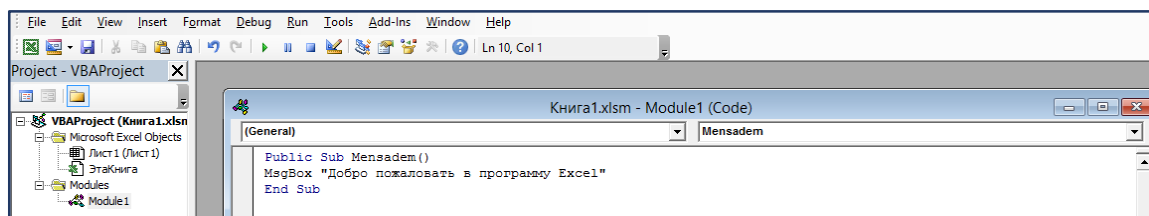
1. Показать сообщение «Добро пожаловать в программу Excel».

Для этого нужно вставить команду MsgBox в свой макрос. MsgBox "текст сообщения".

```
Public Sub Mensadem()
```

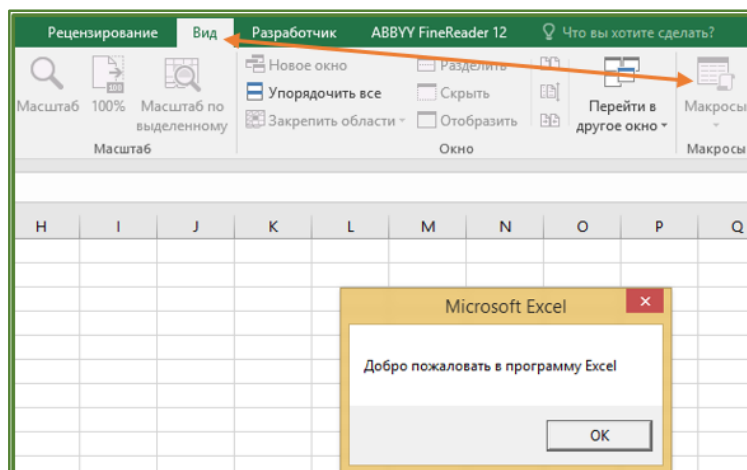
```
MsgBox "Добро пожаловать в программу Excel"
```

```
End Sub
```



(рис.3.24.5). Команда MsgBox

Результат:



(рис.3.24.6). Результат макроса

Эта строка кода может быть вставлена в любой макрос, что очень полезно и интересно.



Контрольные вопросы и задания

1. Сколько столбцов содержится в таблице? Укажите адрес последнего столбца.
2. Сколько строк содержится в таблице? Укажите адрес последней строки.
3. Введите в ячейку N35 свое имя, выровняйте его в ячейке по центру и примените начертание полужирное. Опишите порядок ваших действий.

4. Введите в ячейку C5 текущий год. Опишите порядок ваших действий.
5. Что такое *Visual Basic for Application*.
6. Порядок записи *простых макросов*.
7. Выполнение *макросов*.
8. Редактирование *макросов*.
9. Напишите отчет о проделанной работе, оформите в тетради и сдайте преподавателю на проверку.

25-§. СОЗДАНИЕ ПРЕЗЕНТАЦИЙ В MICROSOFT POWERPOINT.

Цель работы: *Формирование практических навыков работы по созданию и оформлению мультимедийной презентации в программе PowerPoint.*

Краткая теоретическая информация:

Microsoft PowerPoint (от англ. *PowerPoint* - *убедительный доклад*) – одна из прикладных программ Microsoft office, предназначенная для создания и демонстрации презентаций.

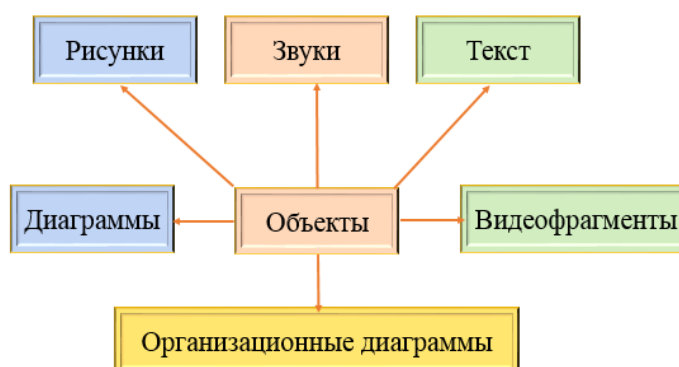
Презентация - это краткое и наглядное представление информации, помогающее докладчику более детально раскрыть суть его работы. Сейчас её используют во многих сферах нашей жизни.

Презентация состоит из множества *слайдов (страниц)*, которые хранятся в одном файле. Презентации можно представлять в электронном виде, распечатывать в виде раздаточного материала (*копии всех слайдов*) или распространять через *Интернет*.

Слайд – логически автономная структура, содержащая различные *объекты*, которые представляются на общем экране монитора.

Каждый слайд презентации обладает свойствами, которые влияют на его отображение во время демонстрации: *размер слайда, разметка слайда*

(расположение заголовков, текста и объектов на слайде), шаблон оформления (дизайн слайда), эффект перехода от слайда к слайду. На каждом слайде можно разместить разные объекты (рис.3.25.1.)

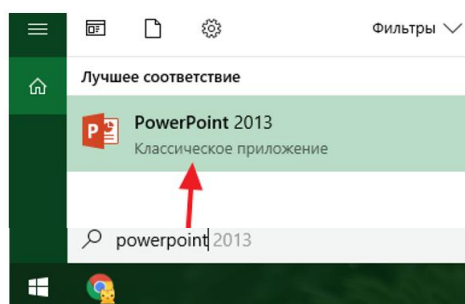


(рис.3.25.1). Размещение объектов на слайде

АЛГОРИТМ СОЗДАНИЯ ПРЕЗЕНТАЦИИ

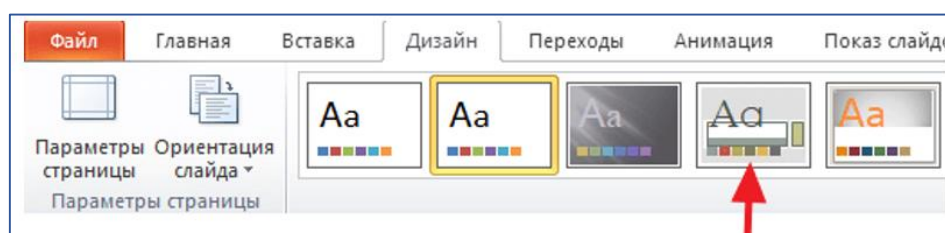
Шаг 1. Запуск PowerPoint.

Для того чтобы начать создание презентации, достаточно просто запустить программу *PowerPoint*. Это можно сделать с помощью ярлыка на рабочем столе.



Шаг 2. Выбор дизайна будущей презентации.

После того, как *PowerPoint* запущен можно сразу приступить к созданию презентации. Лучше всего начать с дизайна, для этого перейдите на вкладку «Дизайн». На этой вкладке вы увидите большой список готовых дизайнов презентации. Выберите один из доступных вариантов.



Справа от списка готовых дизайнов, есть кнопки «Цвета», «Шрифты», «Эффекты» и «Стили фона». С помощью этих кнопок можно подогнать выбранный дизайн под ваши требования. Вы можете изменить цвет и шрифт текста, цвет фона презентации, а также добавить дополнительные эффекты.



Если готовых дизайнов вам недостаточно, то вы можете поискать в интернете другие шаблоны для презентаций PowerPoint.

Шаг 3. Создание слайдов.

После выбора дизайна, можно начинать создавать слайды будущей презентации. Делается это при помощи кнопки «Создать слайд», которая находится на вкладке «Главная». Нажмите на стрелку вниз, под кнопкой «Создать слайд», так чтобы открылось меню с доступными слайдами.

В открывшемся меню **вы увидите** слайды нескольких типов. Это «Титульный слайд», слайд «Заголовок и объект», слайд «Заголовок раздела», слайд «Два объекта» и т.д. Выберите подходящий вам тип слайда и кликните по нему мышкой. Для примера создайте слайд «Заголовок и объект». В результате появится новый слайд с заголовком в верхней части слайда и пустым полем внизу.

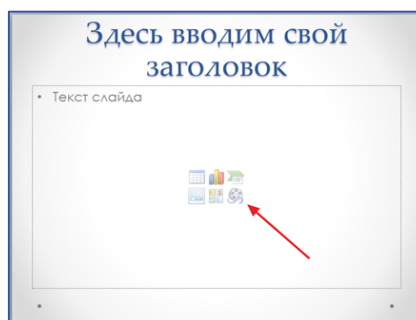
Шаг 4. Заполнение созданных слайдов.

После создания слайда его можно заполнить информацией. Для начала можно изменить заголовок слайда, для этого кликните два раза мышкой по надписи: «Заголовок слайда» и введите другой текст.

После того, как заголовок введен, можно заполнять пустое поле под заголовком. Если под заголовком должен быть текст, то просто кликните по пустому полю и введите нужный текст.

Если же под заголовком должна быть какая-то другая информация, например, видео или картинка, то для этого нужно воспользоваться кнопками,

которые находится посередине этого поля. Здесь доступно шесть кнопок для вставки *таблиц, диаграмм, рисунков SmartArt, фотографий, картинок* из библиотеки PowerPoint и видео.



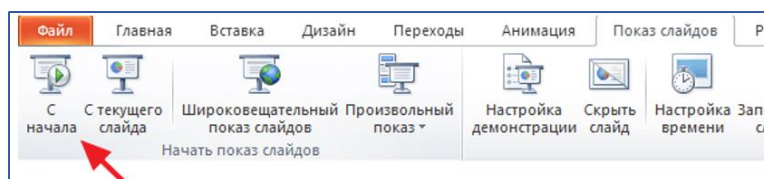
При создании презентаций в *PowerPoint* чаще всего вставляют *фотографии*, поэтому рассмотрим именно этот вариант. Для того чтобы вставить фотографию из файла, нужно нажать на кнопку с изображением фотографии. После этого появится окно для выбора фотографии. Выберите нужную фотографию и нажмите на кнопку «*Вставить*».

После этого выбранная фотография появится под заголовком слайда.

Таким образом вы создали презентацию *PowerPoint* из одного слайда. Для того чтобы добавить еще один слайд, вернитесь на вкладку «*Главная*», снова нажмите на кнопку «*Создать слайд*» и добавьте в презентацию еще один слайд. После этого можно будет заполнить информацией еще один слайд. Повторяйте эту процедуру, пока презентация не будет готова.

Шаг 5. Предварительный просмотр презентации.

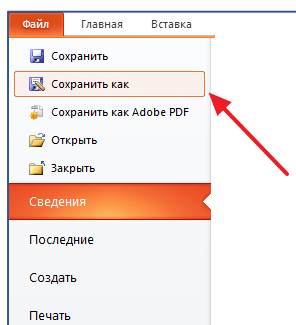
Для того чтобы оценить созданную презентацию, перейдите на вкладку «*Показ слайдов*» и нажмите там на кнопку «*С начала*». После этого презентация откроется на весь экран, и вы сможете увидеть, как это будет выглядеть в законченном виде.



Также можно нажать на кнопку «С текущего слайда». В этом случае воспроизведение презентации начнется не с начала, а с того кадра, на котором вы остановились при работе с презентацией.

Шаг 6. Сохранение презентации.

После того как презентация создана, ее нужно сохранить. Для этого нужно открыть меню «Файл» и выбрать вариант «Сохранить как».



В результате откроется окно для сохранения файла. В этом окне нужно выбрать папку и нажать на кнопку «Сохранить». Таким образом вы получите файл презентации PowerPoint в формате **PPTX**, который потом можно будет открыть и продолжить создание презентации.

Если ваша презентация уже готова к показу, то ее нужно дополнительно сохранить в другом формате. Для этого нужно точно также использовать меню «Файл - Сохранить как», но перед сохранением изменить формат на «Демонстрация PowerPoint».

После такого сохранения вы получите файл в формате **PPSX**. Преимуществом данного формата является то, что презентация начинает воспроизводиться сразу после его открытия, при этом интерфейс программы PowerPoint даже не появляется. Но, файлы в формате **PPSX** нельзя редактировать, поэтому презентацию обязательно нужно сохранять и в формате **PPTX** и в **PPSX**.

Практическое задание 2. Создание 3D эффект картинок в программе PowerPoint.

Порядок выполнения работы:

1. Открыть программу PowerPoint

2. Создать пустой слайд.
3. Выбрать картинку из имеющихся в компьютере.
4. Вставить картинку на слайд (*основной объект на картинке должен иметь контраст с фоном*)



5. Скопировать картинку и вставьте рядом.
6. Выбрать первую картинку и в меню **Формат** сделать рамку (*стили*).



7. В меню **Формат** выберите *Обрезка по фигуре Трапеция*, немного скосите угол.

8. Снова нажмите на *Обрезка* и уменьшите площадь рисунка. И снова нажмите на кнопку *Обрезать*.



9. Теперь работаем со второй картинкой. В меню **Формат** выберите *удалить фон*. Используя функции *удалить* или *добавить выделение* выделите фон, оставляя только объект. Нажмите *сохранить изменения*.

10. Совмещайте картинки. Выделите обе картинки с помощью клавиши **Ctrl**. В меню **Формат** выберите команду *Сгруппировать*.

11. Получился полноценная 3D-картинка, с которой можно работать.



12. *Файл - Сохранить как* выберите *Тип файла*.

13. Покажите результаты работы преподавателю.



Контрольные вопросы и задания

1. Для чего предназначена программа *PowerPoint*?
2. Что такое *презентация*? Что такое *слайд*?
3. Какие режимы просмотра встроены в систему *PowerPoint*?
4. Как запустить программу *PowerPoint*?
5. Как определить назначение элементов меню и кнопок панелей инструментов?
6. Как завершить работу с *PowerPoint*?
7. Создание 3D эффект картинок в программе Power Point со своими фотографиями.

26-§. ОСНОВНЫЕ ВОЗМОЖНОСТИ КОМПЬЮТЕРНОЙ ГРАФИКИ

Цель работы: Выяснить, как *компьютерная графика* связана с нашей жизнью; в каких сферах деятельности человеческого общества она применяется и как она влияет на самого человека.

Краткая теоретическая информация:

Компьютерная графика - область информатики, изучающая методы и свойства обработки изображений с помощью программно-аппаратных средств.

Под видами *компьютерной графики* подразумевается способ хранения изображения на плоскости монитора. Компьютерная графика в настоящее время уже вполне сформировалась как наука. Существует *аппаратное и программное обеспечение* для получения разнообразных *изображений* - от простых чертежей до реалистичных образов естественных объектов. Компьютерная графика используется почти во всех сферах нашей жизни.

Различают три вида компьютерной графики: *растровая, векторная, трёхмерная*. Они отличаются принципами формирования изображения при отображении на экране монитора или при печати на бумаге.

Растровую графику применяют при разработке электронных (*мультимедийных*) и полиграфических изданий. Иллюстрации, выполненные средствами растровой графики, создают с помощью компьютерных программ.

Программные средства для работы с *векторной графикой*, наоборот, предназначены для *создания иллюстраций* и в меньшей степени для их обработки. Такие средства широко используют в *рекламных агентствах, дизайнерских бюро, редакциях и издательствах*.

Программные средства для работы с *трёхмерной графикой* предназначены для автоматической генерации изображений путем математических расчетов. Создание фрактальной художественной композиции, состоит не в рисовании или оформлении, а в *программировании*. Фрактальную графику чаще используют в *развлекательных программах*.

Цветовые модели

Цветовая модель RGB. Любой цвет считается состоящим из трех основных компонентов: *красного (Red), зеленого (Green) и синего (Blue)*. Эти цвета называются основными. Считается также, что при наложении одного компонента на другой яркость суммарного цвета увеличивается. Совмещение

трех компонентов дает нейтральный *цвет (серый)*, который при большой яркости стремится к белому цвету. Метод получения нового оттенка суммированием яркостей составляющих компонентов называют аддитивным методом. Нетрудно догадаться, что чем меньше яркость, тем темнее оттенок. Поэтому в аддитивной модели центральная точка, имеющая нулевые значения компонентов $(0,0,0)$, имеет черный цвет (*отсутствие свечения экрана монитора*). Модель RGB является аддитивной, а ее компоненты: красный $(255,0,0)$, зеленый $(0,255,0)$ и синий $(0,0,255)$ - называют основными цветами.

Цветовая модель CMYK. Эту модель используют для подготовки не экранных, а печатных изображений. Они отличаются тем, что их видят не в проходящем, а в отраженном свете. Чем больше краски положено на бумагу, тем больше света она поглощает и меньше отражает. Совмещение трех основных красок поглощает почти весь падающий свет, и со стороны изображение выглядит почти черным. В отличие от *модели RGB* увеличение количества краски приводит не к увеличению визуальной яркости, а наоборот, к ее уменьшению.

Поэтому для подготовки печатных изображений используется не *аддитивная (суммирующая)* модель, а *субтрактивная (вычитающая)* модель.

Цветовая модель HSB. Некоторые графические редакторы позволяют работать с цветовой моделью HSB. Если модель RGB наиболее удобна для компьютера, а модель CMYK - для типографий, то модель HSB наиболее удобна для человека. Она проста и интуитивно понятна. В модели HSB тоже три компонента: оттенок цвета (*Hue*), насыщенность цвета (*Saturation*) и яркость цвета (*Brightness*). Регулируя эти три компонента, можно получить столь же много произвольных цветов, как и при работе с другими моделями. Оттенок цвета указывает номер цвета в спектральной палитре. Насыщенность цвета характеризует его интенсивность - *чем она выше, тем "чище" цвет*. Яркость цвета зависит от добавления чёрного цвета к данному - чем её больше, тем яркость цвета меньше.

Графические форматы

Формат графического файла - это способ представления графических данных на внешнем носителе. Выбор того или другого формата для сохранения изображения зависит от целей и задач работы с изображением. Если нужна фотографическая точность воссоздания цветов, то преимущество отдадут одному из растровых форматов. Логотипы, схемы, элементы оформления целесообразно хранить в векторных форматах. Формат файла влияет на объем памяти, который занимает этот файл. Графические редакторы позволяют пользователю самостоятельно избирать формат сохранения изображения. Если вы собираетесь работать с графическим изображением только в одном редакторе, целесообразно выбрать тот формат, какой редактор предлагает по умолчанию. Если же данные будут обрабатываться другими программами, стоит использовать один из универсальных форматов. Существуют универсальные форматы графических файлов, которые одновременно поддерживают и векторные, и растровые изображения.

Формат PDF (англ. Portable Document Format - портативный формат документа). В этом формате могут быть сохранены изображения и векторного, и растрового формата, текст с большим количеством шрифтов, гипертекстовые ссылки и даже настройки печатающего устройства. Размеры файлов достаточно малы.

EPS (*Encapsulated PostScript*). EPS позволяет хранить информацию, как о растровой, так и о векторной графике. EPS для передачи векторов и растра в издательские системы, создается почти всеми программами, работающими с графикой. EPS поддерживает все необходимые для печати цветовые модели. EPS имеет много разновидностей, что зависит от программы-создателя. Формат EPS поддерживается программами для разных операционных систем.

BMP (*Windows Device Independent Bitmap*). Применяется для хранения растровых изображений, предназначенных для использования в *Windows*. Графические файлы в этом формате имеют большой информационный объем, т.к. в них на хранение информации о цвете каждого пикселя отводится 24 бита.

GIF (*CompuServe Graphics Interchange Format*). В рисунках, сохраненных в формате GIF, можно использовать только 256 разных цветов. GIF использует LZW-компрессию, что позволяет неплохо сжимать файлы. Это особенно важно для графики, используемой во *Всемирной паутине*.

JPEG (Joint Photographic Experts Group). Строго говоря JPEG-ом называется не формат, а алгоритм сжатия, основанный не на поиске одинаковых элементов, а на разнице между пикселями. Чем выше уровень компрессии, тем больше данных отбрасывается, тем ниже качество.

TIFF (*Target Image File Format*). Аппаратно независимый формат TIFF, один из самых распространенных и надежных на сегодняшний день, его поддерживают практически все программы так или иначе связанные с графикой. Ему доступен весь диапазон цветовых моделей от монохромной до RGB, CMYK В формате TIFF есть возможность сохранения с применением нескольких видов сжатия: *JPEG, ZIP, LZW*.

CDR - формат популярного векторного редактора *CorelDraw*. Свою популярность и распространение пакет получил благодаря кажущейся простоте использования и интерактивным спецэффектам (*линзам, прозраčnostям, нестандартным градиентам и т.д.*). **CCX** - формат векторной графики от компании Corel. Кроме *CorelDraw* ничем не поддерживается. Для полиграфии и Интернета непригоден.

Итак, рассмотрев наиболее распространенные графические форматы, использующиеся для создания изображений, фотографий и т.д. мы систематизировали знания, по данному материалу.

Области применения компьютерной графики

Научная графика. Это направление появилось первым. Первые компьютеры использовались лишь для решения научных и производственных задач. Чтобы лучше понять полученные результаты, производили их графическую обработку, строили графики, диаграммы, чертежи рассчитанных конструкций. Современная научная компьютерная графика дает возможность

проводить вычислительные эксперименты с наглядным представлением их результатов.

Деловая графика - область компьютерной графики, предназначенная для наглядного представления различных показателей работы учреждений. Плановые показатели, отчетная документация, статистические сводки - вот объекты, для которых с помощью деловой графики создаются иллюстративные материалы. Программные средства деловой графики включаются в состав электронных таблиц.

Конструкторская графика используется в работе инженеров-конструкторов, архитекторов, изобретателей новой техники. Этот вид компьютерной графики является обязательным элементом САПР (*систем автоматизации проектирования*). Средствами конструкторской графики можно получать как плоские изображения (*проекции, сечения*), так и пространственные трехмерные изображения.

Иллюстративная графика - это произвольное рисование и черчение на экране компьютера. Программные средства, позволяющие человеку использовать компьютер для произвольного рисования, черчения подобно тому, как он это делает на бумаге с помощью карандашей, кисточек, красок, циркулей, линеек и других инструментов, относятся к иллюстративной графике.

Художественная и рекламная графика - ставшая популярной во многом благодаря телевидению. С помощью компьютера создаются рекламные ролики, мультфильмы, компьютерные игры. Отличительной особенностью этих графических пакетов является возможность создания реалистических изображений и "*движущихся картинок*".

Компьютерная анимация. В недавнем прошлом художники мультипликаторы создавали свои фильмы вручную. Чтобы передать движение, им приходилось делать тысячи рисунков, отличающихся друг от друга небольшими изменениями. Затем эти рисунки переснимались на киноленту. Система компьютерной анимации берет значительную часть

рутинной работы на себя. Например, художник может создать на экране рисунки лишь начального и конечного состояния движущегося объекта, а все промежуточные состояния рассчитает и изобразит компьютер. Полученные рисунки, выводимые последовательно на экран с определенной частотой, создают иллюзию движения. **Мультимедиа** - это объединение высококачественного изображения на экране компьютера со звуковым сопровождением.



Контрольные вопросы и задания

1. Приведите примеры сфер применения технологий *компьютерной графики*.
2. Дайте понятие «*графики*», «*компьютерной графики*» и назовите типы *компьютерных изображений*.
3. Охарактеризуйте *растровые* и *векторные* графические изображения. Что это такое?
4. Как получаются растровые изображения, из чего они состоят? Приведите пример.
5. От чего зависит качество растрового изображения? Что такое пиксель?
6. Назовите достоинства и *недостатки* растрового изображения.
7. Назовите примеры растровых графических редакторов и форматы графических файлов. Для чего они применяются?
8. Как получаются векторные изображения, из чего они состоят? Приведите пример.
9. Назовите *достоинства* и *недостатки* векторной графики.
10. Назовите примеры векторных графических редакторов и форматы графических файлов. Для чего они применяются?

27-§. ПОРЯДОК СЛЕДОВАНИЯ, ОБЪЕДИНЕНИЕ И ГРУППИРОВКА ОБЪЕКТОВ В ПРОГРАММЕ COREL DRAW.

Цель работы: Приобретение уверенных навыков работы с программой *Corel Draw*, создание в ней собственных разработок.

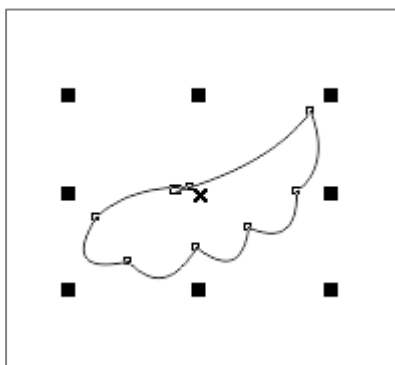
Краткая теоретическая информация:

CorelDRAW - графический редактор векторной графики, разработанный канадской корпорацией *Corel*.

В пакет **CorelDRAW** также входит редактор растровой графики *Corel Photo Paint* и другие программы - например, для захвата изображений с экрана - *Corel Capture*. Программа векторизации растровой графики *Corel Trace* до 12 версии входила в пакет как самостоятельная программа.

Создание открытки «Зимний пейзаж»

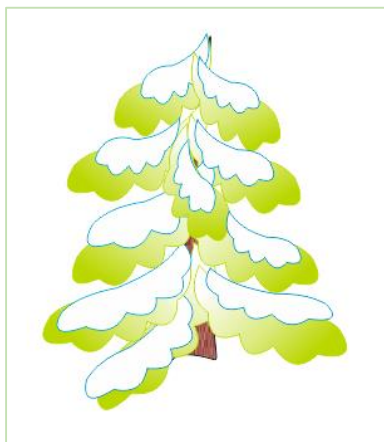
1. Запустите программу «*CorelDraw X7*», выберите альбомную ориентацию листа.
2. Инструментом «кривая через 3 точки» нарисуйте вот такую еловую ветку.



3. Закрасьте градиентной заливкой. Нарисуйте ствол, залейте узорной заливкой, скопируйте ветки, измените их размер и расположение и, таким образом, нарисуйте всю елочку.



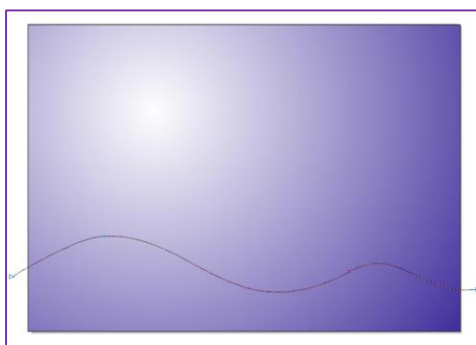
4. Скопируйте ветку, перекрасьте в белый снег с голубыми контурами, поверните, отразите зеркально, дублируйте и получите.



5. Сгруппируйте елку, состоящую из отдельных фигур, в единое целое.

6. Изобразите пейзаж. На весь лист нарисуйте прямоугольник и инструментом «фонтанная заливка» залейте прямоугольник радиально двумя оттенками синего цвета.

7. Нарисуйте инструментом «свободная форма» волнистую линию



8. Инструментом «интеллектуальная заливка» закрасьте градиентом область ниже линии.

9. Таким же образом нарисуйте весь пейзаж и получите примерно следующее.



10. Поместите елку на передний план страницы.

11. Нарисуйте месяц инструментом «свободная форма» и добавьте тень.

12. Откройте эффекты – художественное оформление и добавьте несколько звездочек.

13. В итоге вы должны получить следующий пейзаж.



14. Результаты *выполнения задания* покажите преподавателю.



Контрольные вопросы и задания

1. В чем заключаются преимущества и недостатки векторной графики, по сравнению с пиксельной графикой?

2. Какие варианты действий приводят к выделению нескольких объектов?

3. Как добавить узел на кривую Безье инструментом *Shape (Форма)*?

4. Какие существуют типы узлов на кривой *Безье*?

5. Каким образом можно создать линию переменной толщины?

6. В окне *Fountain Fill* (*Градиентная заливка*) есть список *Type* (*Тип*).
Какие типы градиентной заливки там присутствуют?

7. Какие инструменты интерактивных эффектов есть в программе?

8. Как сохранить файл **CDR** в требуемой версии?

28-§. СОЗДАНИЕ И УПРАВЛЕНИЕ БАЗОЙ ДАННЫХ.

Цель работы: Формирование умений и навыков создания и управления базой данных в СУБД Access

Краткая теоретическая информация:

Информационная система (ИС) - это система, построенная на базе компьютерной техники, предназначенная для *хранения, поиска, обработки и передачи* значительных объемов информации, имеющая определенную практическую сферу применения.

База данных - это информационная система, которая хранится в электронном виде.

База данных (БД) - *организованная совокупность данных, предназначенная для длительного хранения во внешней памяти ПК, постоянного обновления и использования.*

База данных служат для хранения и поиска большого объема информации. Примеры баз данных: *записная книжка, словари, справочники, энциклопедии* и т.д.

Классификация баз данных:

1. По характеру хранимой информации:

- *Фактографические* – содержат краткие сведения об описываемых объектах, представленных в строго определенном формате (*картотеки, БД книжного фонда библиотеки, БД кадрового состава учреждения*),

- *Документальные* – содержат документы (*информацию*) самого разного типа: *текстового, графического, звукового, мультимедийного* (*справочники, словари, и др.*)

2. По способу хранения данных:

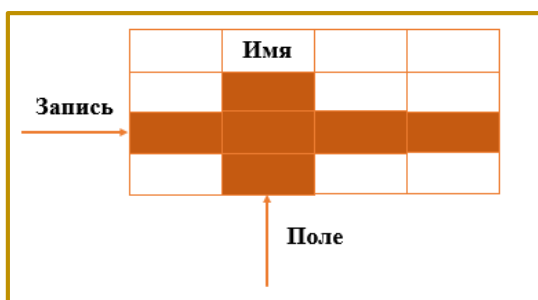
- *Централизованные* (*хранятся на одном компьютере*),
- *Распределенные* (*используются в локальных и глобальных компьютерных сетях*).

3. По структуре организации данных:

- *Реляционные* (*табличные*),
- *Нереляционные*.

Термин «*реляционный*» (*от лат. relatio – отношение*) указывает на то, что такая модель хранения данных построена на взаимоотношении составляющих её частей. *Реляционная база данных*, по сути, представляет собой двумерную *таблицу*.

Каждая строка такой таблицы называется *записью*. Столбцы таблицы называются *полями*: каждое поле характеризуется своим именем и типом данных. Поле БД – это *столбец таблицы*, содержащий значения определенного свойства.



(рис3.28.1). Структура таблицы

Система управления базами данных (СУБД) – это система, обеспечивающая *поиск, хранение, корректировку данных*, формирование ответов на запросы. Система обеспечивает сохранность данных, их конфиденциальность, перемещение и связь с другими программными средствами.

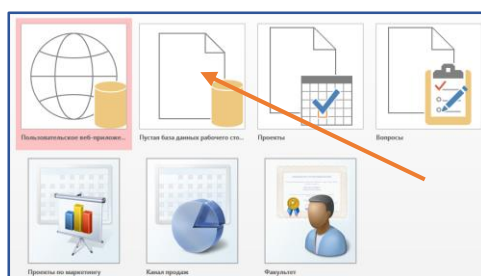
Основные действия, которые пользователь может выполнять с помощью СУБД:

- ✚ создание структуры БД;
- ✚ заполнение БД информацией;
- ✚ изменение (*редактирование*) структуры и содержания БД;
- ✚ поиск информации в БД;
- ✚ сортировка данных;
- ✚ защита БД;
- ✚ проверка целостности БД.

Задание: Создайте базу данных «Библиотека». База данных должна содержать 4 взаимосвязанных таблицы: **Книги, Авторы, Издательства, Книги-Авторы** (для связи таблиц **Книги** и **Авторы**).

Ход работы

1. Откройте *MS Access*, используя путь: *Пуск\Все Программы\Microsoft office\ MS Access*.
2. Выберите *Пустая база данных* рабочего стола.



(рис3.28.2). Пустая база данных

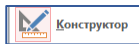
3. Укажите имя файла "**БД_Библиотека_Фамилия студента** (например, **БД_Библиотека_Иванов И.И.**)". Если имя файла указано без расширения, расширение будет добавлено автоматически. Щелкните по кнопке *Создать*.

4. *Приложение Access* создаст базу данных с пустой таблицей с именем «**Таблица1**» и откроет эту таблицу в режиме таблицы. Курсор находится в первой пустой ячейке столбца «**Добавить поле**».

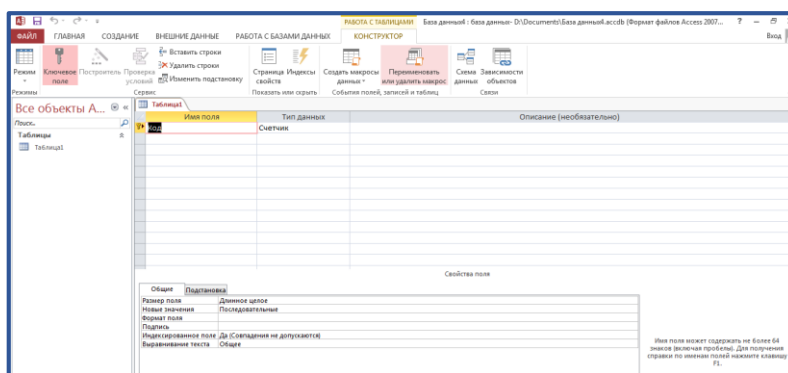
Создание таблиц в режиме Конструктора

В режиме конструктора сначала создается структура новой таблицы. Затем можно переключиться в режим таблицы для ввода данных или ввести данные, используя другой метод, например, вставку или импорт.

5. На вкладке Создание в группе Таблицы щелкните Конструктор таблиц.



6. В режиме конструктора необходимо задать поля таблицы, определить их типы данных и другие свойства. Для каждого поля в таблице введите имя в столбце *Имя поля*, а затем в списке *Тип данных* выберите тип данных.



(рис.3.28.3). Интерфейс программы Access

Справочная информация

Таблица может содержать следующие типы полей:

Свойство **Тип данных** (*Data Type*) определяет тип данных, сохраняемых в поле таблицы. В каждое поле допускается ввод данных только одного типа.

Тип данных	Хранить
Текстовый	Алфавитно-цифровые знаки Используется для текста или текста и чисел, не применяемых в расчетах.
Поле МЕМО	Алфавитно-цифровые знаки (<i>более 255 знаков</i>) или форматированный текст.
Числовой	Числовые значения (<i>целые или дробные</i>). Используется для хранения числовых данных, используемых в вычислениях, за исключением денежных значений.
Дата/время	Даты и время. Используется для хранения значений даты и времени.
Денежный	Денежные значения. Используется для хранения денежных значений (<i>валюты</i>).

Счетчик	Уникальное числовое значение, которое автоматически вводит Office Access при добавлении записи. Используется для создания уникальных значений, например, для первичного ключа. В поле с типом данных «Счетчик» числа могут последовательно увеличиваться на указанное приращение или выбираться случайно.
Логический	Логические значения. Используется для полей, которые могут содержать одно из двух значений, например «Да» и «Нет» или <i>True</i> и <i>False</i> .
Поле объекта OLE	OLE-объекты или другие двоичные данные. Используется для хранения OLE-объектов других приложений Microsoft Windows.
Вложение	Рисунки, изображения, двоичные файлы, файлы <i>Microsoft Office</i> . Стандартный тип данных для сохранения цифровых изображений и любого типа двоичных файлов.
Гиперссылка	Гиперссылки. Используется для хранения гиперссылок вызова веб-страниц одним щелчком с помощью URL-адреса или файлов с помощью формата универсального имени UNC. Кроме того, можно использовать ссылку на объекты Access, хранящиеся в базе данных.
Мастер подстановок	Фактически типом данных не является, а вызывает мастер подстановок. Используется для запуска мастера подстановок, с помощью которого можно создать поле, позволяющее выбрать значение из другой таблицы, запроса или списка значений, используя поле со списком.

(табл.3.28.1). Таблица типы данных

7. При желании можно ввести описание для каждого поля в столбце Описание. Это описание будет отображаться в строке состояния, когда в режиме таблицы курсор будет находиться в данном поле.

8. Задайте поля таблицы Книги в соответствии со столбцами **Поле** и **Тип данных** (табл.3.28.1). Тип данных указывается выбором элемента из раскрывающегося списка.

Поле	Тип данных	Размер поля	Маска ввода
Код книги	Число	Целое	
Название книги	Краткий текст	30	
Количество страниц	Краткий текст		###"стр."

Количество экземпляров	Число		
Стоимость книги	Денежный		
Год издания	Дата и время		
Код издательства	Число	Целое	
Предметная область	Краткий текст	20	
Вид печатной продукции	Краткий текст	10	

(табл.3.28.2). Раскрывающий список

9. Сделайте поле Код книги ключевым - вкладка **Конструктор**/ группа **Сервис**/команда **Ключевое поле**.

10. Щелкая курсором на полях, задайте для них свойство **Размер поля** (*выбирая из списка или вводя значение с клавиатуры*) в соответствии с таблицей 1.

11. Для полей Вид печатной продукции, Предметная область определите *список фиксированных значений*.

Вид печатной продукции: газета, журнал, книга, брошюра.

Предметная область - информатика, математика, физика.

Для этого в окне свойств полей выберете вкладку **Подстановка**, затем **Тип элемента управления** установить **Поле со списком** (*или Список*), далее **Тип источника строк** - **Список значений**, и в свойстве **Источник строк** ввести список элементов, разделяемых точкой с запятой (;).

Например, для поля **Предметная область** - информатика; математика; физика.

12. Задайте для поля Количество страниц *Маску ввода* в соответствии с табл.1.

Справочная информация

Свойство полей «Маска ввода».

Свойство *Маска ввода* (*InputMask*) задает маску ввода, облегчающую ввод данных в элемент управления- поле. Например, удобно создать

следующую маску ввода для поля <Телефон>, позволяющую вводить только цифры и автоматически добавляющую промежуточные символы:

(____) ____-____ Значение данного свойства определяется автоматически при использовании мастера по созданию масок ввода.

При создании маски ввода пользователь имеет возможность указать, что часть данных следует вводить обязательно (*например, региональный код для телефонных номеров*), а другие данные являются необязательными (*например, добавочный номер телефона*). Эти символы определяют тип данных, например, номер символа, который необходимо ввести для каждого символа маски ввода.

Символы, которые следует вводить в маску ввода определяются следующими специальными символами.

Символ	Описание
0	Цифра (<i>обязательный символ; знаки (+) и (-) не разрешены</i>).
9	Цифра или пробел (<i>необязательный символ; знаки (+) и (-) не разрешены</i>).
#	Цифра или пробел (<i>необязательный символ; незаполненные позиции выводятся как пробелы в режиме редактирования, но удаляются при сохранении данных; знаки (+) и (-) не разрешены</i>).
L	Буква (<i>обязательный символ</i>).
?	Буква (<i>необязательный символ</i>).
A	Буква или цифра (<i>обязательный символ</i>).
a	Буква или цифра (<i>необязательный символ</i>).
&	Любой символ или пробел (<i>обязательный символ</i>).
C	Любой символ или пробел (<i>необязательный символ</i>).
.,:;- /	Десятичный разделитель, разделители групп разрядов, времени или даты.
<	Преобразует все символы к нижнему регистру.
>	Преобразует все символы к верхнему регистру.
!	Указывает, что маска ввода заполняется справа налево; этот символ следует использовать, если в левой части маски находятся позиции, заполнять которые не обязательно. Маски ввода обычно заполняются слева направо. Символ восклицательного знака можно помещать в произвольную позицию в маске ввода.
\	Указывает, что следующий символ следует воспринимать как постоянный (<i>a не специальный</i>) символ (<i>например, \A представляет символ A</i>).

(табл.3.28.3). Таблица специальных символов

13. Для поля Код книги задайте свойство *Подпись - Регистрационный номер* и поле *Обязательное поле* – Да.

Справочная информация

Свойство "Подпись".

Свойство *Подпись (Caption)* определяет текст, который выводится в подписях объектов в различных режимах. Подпись поля указывает текст, который выводится в подписях, присоединенных к элементам управления, создаваемых путем переноса с помощью мыши из списка полей. Этот текст также выводится для таблицы или запроса в заголовке столбца в режиме таблицы.

Свойство "Обязательное поле".

Свойство *Обязательное поле (Required)* указывает, требует ли поле обязательного ввода значения. Если это свойство имеет значение "Да", то при вводе новой записи необходимо ввести значение в это поле или в любой присоединенный к нему элемент управления. *Пустые (Null)* значения в этом поле не допускаются. Например, можно потребовать, чтобы в элементе управления *"Фамилия"* в каждой записи обязательно выводилась какая-либо фамилия. Чтобы позволить ввод в поле пустых значений, недостаточно указать для свойства *Обязательное поле* значение *"Нет"*.

14. Когда все необходимые поля будут добавлены, сохраните таблицу.

15. Перейдите в режим заполнения таблицы, щелкнув по кнопке «*Вид*» инструментальной панели.

16. Заполните таблицу *10-ю записями* о книгах из несколько предметных областей.

17. Аналогично таблице Книги создайте и сохраните еще 3 таблицы: Авторы, Издательства и Книги-Авторы, в соответствии с таблицей 2. В столбце Поле названия полей, которые нужно сделать ключевыми - выделены жирным шрифтом.

Таблица	Поле	Тип данных	Размер поля
Авторы	Код автора	Число	Целое
	Фамилия	Краткий текст	30

	Имя	Краткий текст	20
	Отчество	Краткий текст	15
	Страна	Краткий текст	20
Книги-Авторы	Код автора	Число	Целое
	Код книги	Число	Целое
Издательства	Код издательства	Число	Целое
	Название	Краткий текст	40
	Город	Краткий текст	
	Примечание	Краткий текст	

(табл.3.28.4). - Структура таблиц: Авторы, Издательства и Книги-Авторы

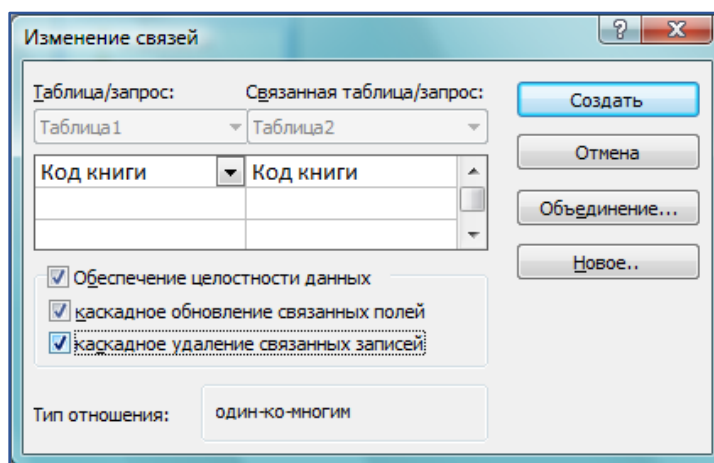
18. Задайте схему данных таблиц: вкладка Работа с базами данных/группа «Показать» или скрыть/команда Схемы данных.

19. В появившемся диалоге Добавление таблицы, добавьте все таблицы (*Авторы, Издательства, Книги, Книги-Авторы*), последовательно указывая на названия таблиц и щелкая кнопку Добавить. Затем нажмите кнопку Закреть.

20. Таблицы в схеме данных можно перемещать и менять размер. Сделайте все поля таблиц видимыми и расположите таблицы последовательно.

21. Свяжите первичный ключ таблицы *Книги (Код книги)* с внешним ключом таблицы *Книги-Авторы (Код книги)* связью «один – ко - многим». Для этого позиционируемся курсором на поле первичного ключа, щелкаем левой кнопкой мыши, не отпуская, перетаскиваем появившийся значок на поле внешнего ключа.

22. В появившемся диалоге Изменение связей установите обеспечение целостности данных, каскадное обновление и удаление данных. Нажмите кнопку Создать.



(рис.3.28.4). Окно изменение связей

23. Аналогичным образом установите связи между таблицами Авторы (*Код автора*) и Книги-Авторы (*Код автора*), Книги (*Код издательства*) и Издательства (*Код издательства*).

24. Сохраните и закройте окно Схема данных.

25. Заполните таблицы Авторы, Издательства и Книги-Авторы записями: для каждой таблицы - *10 записей*. При этом значения внешних ключей не должны отличаться от значений первичных.



Контрольные вопросы и задания

1. С чего начинается создание БД?
2. Что такое СУБД?
3. Какая СУБД установлена на компьютерах в вашем вузе?
4. Перечислите основные объекты СУБД.
5. Какие функции они выполняют?

29-§. СОЗДАНИЕ ФОРМЫ БАЗЫ ДАННЫХ В СРЕДЕ MS. ACCESS

Цель работы: Освоение приемов создания формы и заполнения таблицы в режиме формы при работе с БД в среде MS Access.

Краткая теоретическая информация:

Форма - это объект, с помощью которого пользователи могут добавлять, редактировать и отображать данные, хранящиеся в базе данных Access.

Если база данных **Access** используется несколькими пользователями, хорошо продуманные **формы** - залог точности данных и эффективности работы с ними.

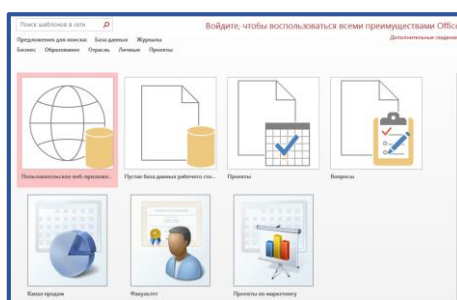
Создать форму на компьютере в базе данных *Access* можно несколькими способами:

1. Создание формы из существующей таблицы или запроса в Access
2. Создание пустой формы в Access
3. Создание разделенной формы в Access
4. Создание формы, отображающей несколько записей, в Access
5. Создание формы, содержащей подчиненную форму, в Access
6. Создание формы навигации в Access

Задание: Создание пустой базы данных с помощью шаблонов таблиц.

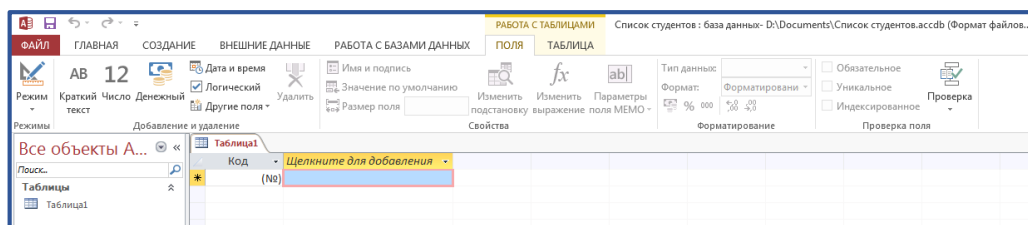
Порядок работы

1. Запустите программу СУБД *Microsoft Access*. Для этого выполните: *Пуск - Все программы - Microsoft office - Microsoft office Access*).
2. Перед Вами откроется окно следующего вида (Рис.1):



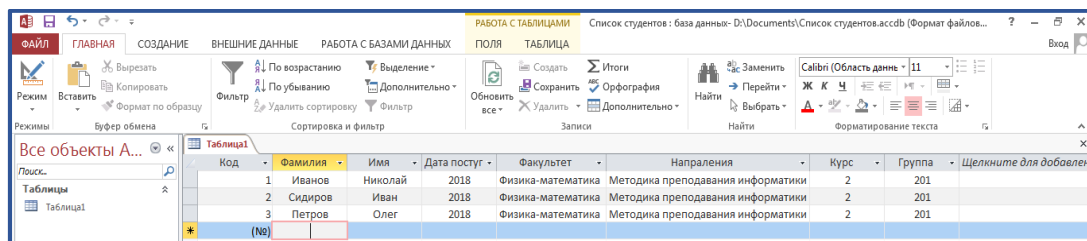
(рис.3.29.1). Окно шаблонов

3. Выберите команду Новая база данных. Затем введите имя файла - *Список студентов* и нажмите кнопку **Создать**. Перед Вами откроется окно следующего вида (рис.2).



(рис.3.29.2). Интерфейс программы

4. Заполните ее следующими данными (рис.3).



(рис.3.29.3). Заполнение таблицы

5. Сохраните таблицу под именем *Студенты*.

6. В данной таблице отсортируйте столбец “*Фамилия*” по алфавиту

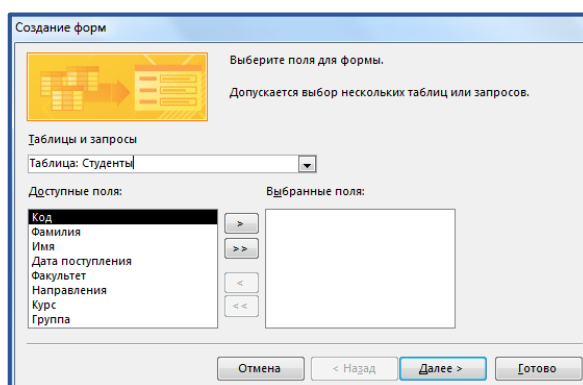
Задание 2: Создайте форму на основе двух таблиц: **Студенты**, **Оценки** для отображения и редактирования записей базы данных **Фамилия студента**.

1. Откройте базу данных **Фамилия студента**.

2. Перейдите на вкладку **Создание**.

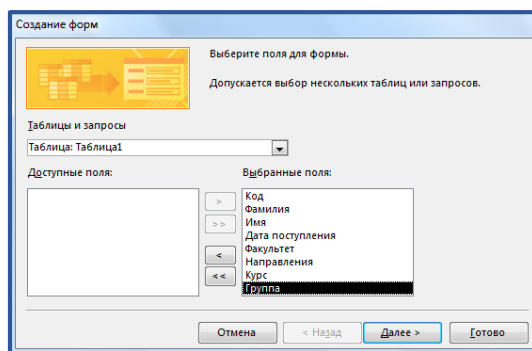
3. Выберите в группе **Формы** списка **Мастер форм**.

4. В окне **Создание форм** выберите таблицы или запросы на основе которых будем создавать формы. Здесь же указываем на поля в таблицах или запросах, которые будем размещать в форме.



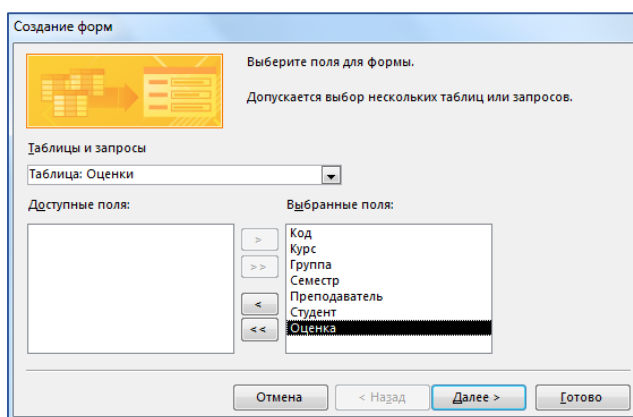
(рис.3.29.4). Заполнение таблицы

5. В диалоговом окне **Создание форм** выберите поля для формы: все поля.



(рис.3.29.5). Таблица Студенты.

6. Не спешите нажимать кнопку **Далее**. В списке **Таблицы и запросы** выберите таблицу **Оценки** и из окна **Доступные поля** выберите все поля таблицы **Оценки**. Нажмите **Далее**.



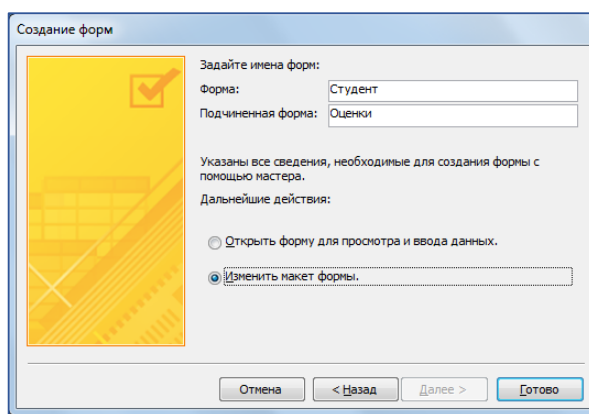
(рис.3.29.6). Создание форм

7. Выберите вид представления данных: **Подчиненные формы**. (данные главной таблицы и данные связанной с ней таблицы размещаются в одной форме).

8. На следующем шаге выберите внешний вид подчиненной формы - **Ленточный**.

9. Далее выберите требуемый стиль оформления (по своему усмотрению).

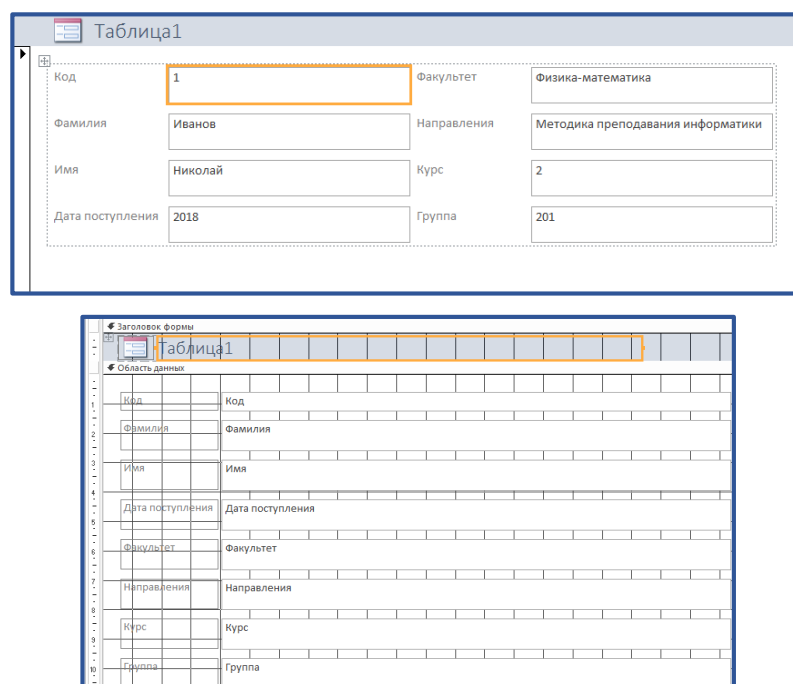
10. Введите имена форм: **Студенты** - для главной формы, **Оценки** - для подчиненной. Выберите пункт **«Изменить макет формы»**.



(рис.3.29.7). Создание форм

Нажмите на кнопку *готово*.

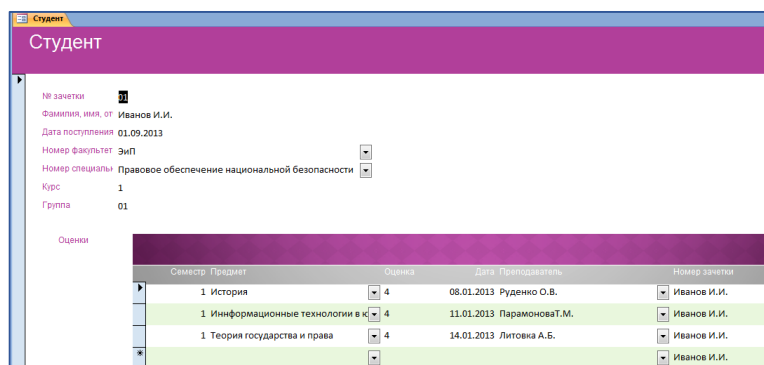
11. Экранная форма открывается в **Конструкторе форм** (*режим Конструктор*).



(рис.3.29.8). Режим конструктора

12. Для редактирования формы и всех ее элементов можно воспользоваться окном **Свойств**. Вызвать окно **Свойств** можно через контекстное меню.

13. Закройте форму и откройте её двойным щелчком по имени формы **Студенты**. Форма откроется в режиме **Формы** и будет выглядеть следующим образом:



(рис.3.29.9). Вид формы Студенты

14. Самостоятельно:

• Создайте конспект с описанием элементов окна **Свойств**. Информацию об элементах окна **Свойств** возьмите из справочной системы. Текст конспекта сохраните в файле **Конспект Форма**.

- Пример фрагмента конспекта:

Все свойства в окне разбиты на группы:

- ✓ **Макет** - *расположение, шрифт, цвет и прочее*, связанное с внешним видом объекта;

- ✓ **Данные** - в этом разделе важнейшее свойство – *Данные* или *Источник записей* - для объектов, связанных с редактированием каких-либо данных;

- ✓ **События** - методы, т.е. процедуры (*программы*), выполняющиеся для объекта при наступлении определенных событий (*Загрузка, Открытие, До обновления, После обновления и пр.*);

- ✓ **Другие** - прочие свойства. И так далее.

• Создавая конспект, обратите внимание на то, что в форму можно добавлять объекты, размещённые в группе **элементы управления** вкладки **Конструктор** (*появляется при переходе в режим Конструктор*):



(рис.3.29.10). Элементы управления

- Опишите элементы управления в своём конспекте.
- Сохраните файл базы данных.



Контрольные вопросы и задания

1. Таблица. Её структурные элементы. Способы создания и работы с ней.
2. Поле таблицы. Типы данных поля. Свойства поля.
3. Понятие ключевого поля.
4. Запись в таблице. Сортировка и фильтрация записей в таблице.
5. Связи между таблицами. Типы связей.
6. Главная и подчиненная таблицы. Очередность их заполнения
7. Форма. Назначение. Структура формы. Способы создания. Работа с формой.
8. Элементы управления, используемые в форме. Построитель выражений.
9. Управляющая форма. Назначение. Свойство Источник записей формы.

30-§. СОЗДАНИЕ ЗАПРОСОВ И ОТЧЕТОВ В СУБД MS. ACCESS.

Цель работы: Дать возможность приобрести практические навыки в создании *запросов* и *отчетов* в программе *MS Access*.

Краткая теоретическая информация:

При работе с таблицами можно в любой момент выбрать из базы данных необходимую информацию с помощью запросов.

Запрос - это обращение к БД для поиска или изменения в базе данных информации, соответствующей заданным критериям.

С помощью *MS Access* могут быть созданы следующие типы запросов: *запросы на выборку*, *запросы на изменение*, *перекрестные запросы*, *запросы с параметром*.

Одним из наиболее распространенных запросов является *запрос на выборку*, который выполняет отбор данных из одной или нескольких таблиц по заданным пользователем критериям, не приводящий к изменениям в самой базе данных.

Отчеты представляют собой наилучшее средство представления информации из базы данных в виде *печатного документа*. По сравнению с другими методами вывода данных на печать отчеты обладают двумя принципиальными преимуществами:

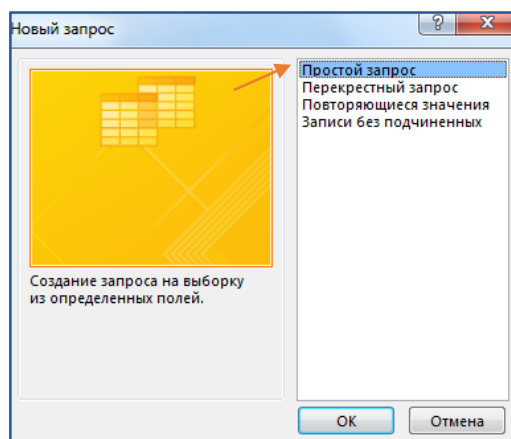
- ✚ отчеты предоставляют широкие возможности для группировки и вычисления промежуточных и общих итогов для больших наборов данных;

- ✚ отчеты могут быть использованы для получения красиво оформленных материалов, наклеек и других документов.

В основном *отчеты* применяются для просмотра «картины в целом», поэтому часто базируются на *многотабличных запросах*.

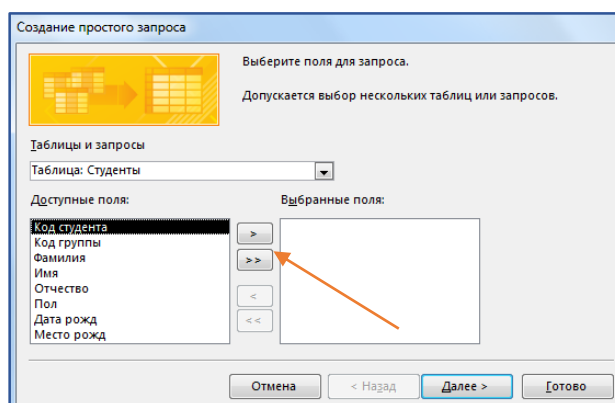
Задание: Создайте запрос к базе данных **Фамилия студента**, содержащий информацию о студентах Ташкентского государственного педагогического университета, сдавших предмет «*Информационные технологии в образовании*», на *отлично*. Данные в запросе отсортируйте по полю **Ф.И.О.** Запрос создайте на основе данных из таблиц *Студенты*, *Оценки* и *Дисциплин* под управлением **Мастера запросов**.

1. Для создания запроса на вкладке **Создание**, в группы **Запросы**, нажмите кнопку *Мастер Запросов*. В Окне **Новый запрос** выберите **Простой запрос**.



(рис.3.30.1). Новый (простой) запрос.

2. На первом шаге создания запроса под управлением **Мастера запросов** выберите таблицы и поля, которые нужно включить в запрос. Поля для формирования запроса могут размещаться в нескольких таблицах. В нашем случае выберите все поля из таблицы **Студенты**. (29-§. Создание формы базы данных в среде Ms. Access)

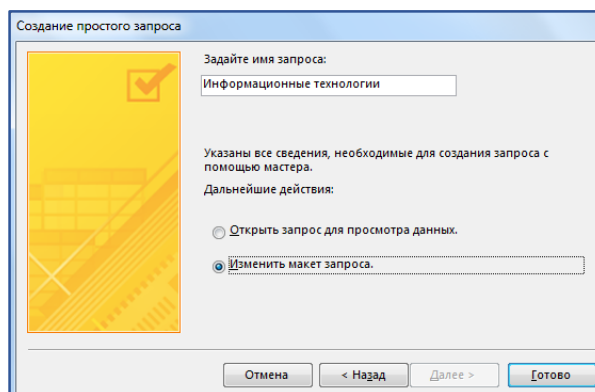


(рис.3.30.2). Создание простого запроса.

В таблице *Оценки* выберите первые 4 поля, в таблице *Дисциплин* все поля. Нажимаем Далее.

3. На втором шаге выберите вид отчёта: *Подробный* или *Итоговый отчет*. Мы хотим создать *Подробный отчет*. Нажимаем Далее.

4. На шаге 3 задайте название запроса **Информационные технологии** (5) и выберите возможность *просмотра данных в сформированном запросе* или же *сделать изменения в макете запроса*: Изменить макет запроса. Выберите последний вариант и нажмите кнопку готово. Запрос открывается в конструкторе запросов.



(рис.3.30.3). Создание простого запроса

5. Модифицируйте запрос для задания условия отбора данных и упорядочения студентов в алфавитном порядке их фамилий. Для этого:

- ✓ колонке поля **Ф.И.О.** задайте сортировку по возрастанию,
- ✓ в поле **Семестр** строки **Условие отбора** введите **1** (*первый семестр*),
- ✓ в поле **Оценка** строки **Условие отбора** задайте условие **5** (*отлично*),
- ✓ в поле **Названия предмета** строки **Условие отбора** введите

«Информационные технологии в образовании».

6. Закройте окно конструктора и выполните запрос командой **«Открыть»** или двойным щелчком мышью по имени запроса. Результат отбора данных будет показан на экране в виде таблицы.

7. **Самостоятельно:** создайте запрос с именем *Информационные технологии (4 или 5)*, содержащий информацию о студентах физико-математического факультета, сдавших предмет *«Информационные технологии в образовании»* на *хорошо* и *отлично*.

В условии отбора оценок используйте конструкцию **OR**, введите ограничение на поле *номер факультета* - для вывода списка студентов физико-математического факультета.

8. Сохраните файл базы данных и покажите результат преподавателю.

Создание отчетов в СУБД MS Access

1. Запустите **Access** и откройте базу данных **Студенты**. (*29-§. Создание формы базы данных в среде Ms. Access*).

Отчеты предназначены для отображения данных из таблиц и запросов базы данных в *«бумажном»* варианте. Отличаются от форм тем, что позволяют легко выполнить группировку данных и подведение итогов.

2. Создание отчетов с использованием мастеров.

Для создания отчетов на вкладке **Создание**, в группы **Отчеты**, нажмите кнопку **Мастер отчетов**. Выберите источником данных таблицу **Студенты**. Выберите для включения в запрос поля *Группа, Фамилия, Имя, Адрес, Телефон* в указанном порядке. Не добавляя уровни группировки, отсортируйте поля по

возрастанию номера группы, затем по возрастанию фамилии, затем имени. Выберите макет *Табличный, любой стиль*, задайте имя **Студенты**.

Отчет открывается в режиме просмотра. Обратите внимание на содержание панели инструментов режима просмотра (*она называется «Предварительный просмотр»*).

Войдите в режим конструктора отчетов, ознакомьтесь с содержанием бланка отчета. Найдите области заголовков, примечаний, колонтитулов. Обратите внимание, что появилась панель элементов управления. Это означает, что инструменты конструктора отчетов такие же, как инструменты конструктора форм. Создание отчета аналогично созданию форм. В области заголовка располагается общий заголовок отчета, в верхнем колонтитуле – заголовки полей, они будут повторены на каждой странице отчета. В области нижнего колонтитула – текущая дата и номера страниц отчета, они тоже будут повторены на каждой странице. Посмотрите, какого типа эти элементы. Для заголовков используется элемент **Надпись**, для отображения данных элемент **Поле**. Элемент **Линия** отделяет область верхнего колонтитула.

3. Группировка данных.

Для объединения записей в группы используется команда **Сортировка и группировка** из меню **Вид**. В отчете **Студенты** записи о студентах должны быть сгруппированы по номерам групп. Используйте этот пункт меню, чтобы добавить в отчет область **Заголовок группы** для номеров групп (*значение Да*), и задайте порядок сортировки по возрастанию. Перенесите в область **Заголовок группы** надпись **Группа** из верхнего колонтитула, и поле **Группа** из области данных. Отделите тонкой линией сверху данные о различных группах.

Просмотрите результат в режиме просмотра.

4. Нумерация записей в отчетах.

Для нумерации записей в группе или по всему отчету, необходимо добавить в область данных отчета свободное поле, и задать значения свойства **Данные** равным формуле **=1**. Для свойства **Сумма с накоплением** нужно

установить значение «**Для группы**». Если нужна сквозная нумерация по всему отчету, свойство **Сумма с накоплением** должно иметь значение «**Для всего**». Просмотрите результат в режиме просмотра.

5. Создание отчетов с использованием мастеров. Группировка данных.

Используйте **Мастер отчетов** для построения отчета для таблицы **Сессия**. Выберите для включения в запрос поля *Группа, Фамилия, Имя, и оценки* за три экзамена. Добавьте уровень группировки (*по группе*), отсортируйте по возрастанию фамилии, затем имени. Выберите макет **Ступенчатый**, любой стиль, задайте имя **Сессия**.

6. Подведение итогов в отчетах.

Для подведения итогов по группам используется команда **Сортировка и группировка** из меню **Вид**. В отчете записи о студентах уже сгруппированы по номерам групп. Добавьте в отчет область **Примечание группы** для номера группы (*значение Да*). В область примечаний добавьте вычисляемые поля, чтобы вычислить средние значения оценок по предметам для каждой группы (*функция Avg()*). Добавьте надпись: «*Средние по предметам*». Отделите тонкой линией снизу данные о различных группах.

Просмотрите результат в режиме просмотра.

7. Группировка данных и подведение итогов при создании отчетов с использованием мастера.

Мастер может автоматически подвести итоги при группировке данных. Создайте **новый отчет** на данных таблицы **Сессия**. На шаге **Сортировка** нажмите кнопку **Итоги** и подведите итоги как средние оценки по экзаменам. Покажите **Только Итоги**. Сохраните отчет с именем **Итоги**.

Просмотрите результат в режиме просмотра.

Войдите в конструктор. Удалите поле, вычисляющее количество записей, текст надписи **Avg** замените на *Среднее по предметам*. Удалите поле **Фамилия** из области данных и надпись **Фамилия** из колонтитула. Запомните, как расположены элементы отчета.

8. Создание отчетов на запросах. Создание отчета в режиме конструктора.

Конструктор отчетов во многом схож с конструктором форм. Оба конструктора имеют одинаковую **Панель элементов**. Создание отчета аналогично созданию форм.

Создайте отчет на запросе **Средние по группам** в режиме **конструктора**. Названия предметов должны быть в верхнем колонтитуле. Поле и название группа должно быть в области заголовка группы. Средние оценки за экзамены должны быть в области данных.

Разместите в области верхнего колонтитула **поле** для вывода даты. Свойство **Данные** этого поля задайте с помощью встроенной функции **Now()** из списка функций. Свойство **Формат** этого поля должно быть форматом отображения даты.

В нижнем колонтитуле разместите номера страниц отчета, используя пункт меню **Вставка - Номера страниц**.



Контрольные вопросы и задания

1. Для чего используются запросы?
2. Способы создания запросов.
3. Можно ли создавать запросы на основе нескольких таблиц?
4. Запишите общий алгоритм создания отчета.
5. Для чего используется конструктор отчетов?
6. Запишите алгоритм создания диаграммы в отчете.
7. В каком случае в Мастере отчетов появляется кнопка *Итоги*?
8. Какие статистические функции можно вычислять в итогах?

ЛАБОРАТОРНЫЕ РАБОТЫ

31-§. РЕДАКТИРОВАНИЕ И ФОРМАТИРОВАНИЕ ДОКУМЕНТОВ В ТЕКСТОВОМ РЕДАКТОРЕ MS WORD

Цель работы: В результате выполнения лабораторной работы студент должен иметь *представление о функциональных возможностях текстового редактора Microsoft Word. Приобрести навыки составления, редактирования и форматирования* электронных документов в среде *Microsoft Word*.



Краткая теоретическая информация:

Редактирование текста

К средствам редактирования текста относят: *копирование, вставка, вырезание фрагментов текста, поиск заданных фрагментов текста, автоматический перенос слов, проверка грамматики и орфографии* и др.

Копирование - *помещение выделенного блока текста в новое место документа с одновременным сохранением этого блока на старом месте.*

Перемещение - *помещение выделенного блока текста в новое место документа с одновременным удалением этого блока на старом месте.*

Для копирования необходимо выделить нужный фрагмент, щелкнуть мышью на кнопке *Копировать* , переместить курсор в то место текста, куда нужно скопировать фрагмент, и щелкнуть на кнопке *Вставить* . Можно использовать также меню **Главная**, команды «**Копировать**» и «**Вставить**», контекстное меню, сочетание клавиш **Ctrl+C** (*Ctrl+Insert*) и **Ctrl+V** (*Shift+Insert*). Для перемещения блока текста необходимо выделить нужный фрагмент, щелкнуть мышью на кнопке «**Вырезать**» переместить курсор в то место текста, куда нужно скопировать фрагмент, и щелкнуть на

кнопке «**Вставить**». Можно использовать также меню **Главная**, команды «**Вырезать**» и «**Вставить**», контекстное меню, сочетание клавиш **Ctrl+X** (*Shift+Delete*), **Ctrl+V** (*Shift+Insert*).

Быстрое выделение текста:

- ✓ **Слово** - двойной щелчок в нем;
- ✓ **Предложение** - щелчок в нем с нажатой клавишей **Ctrl**;
- ✓ **Абзац** - тройной щелчок в нем;
- ✓ **весь документ** – **Главная** – *Выделить* - *Выделить все* или **Ctrl+A**.

Для выделения фрагмента текста с помощью мыши необходимо *установить указатель* в начало фрагмента, нажать левую кнопку мыши и, не отпуская кнопки, *перетянуть указатель* в конец фрагмента.

Нумерация страниц

Выполнить команду **Вставка - Номер страницы**. В появившемся окне можно выбрать положение номера страницы, формат номера. Чтобы удалить или изменить номер страницы, нужно дважды на нем щелкнуть.

Колонтитулы

Колонтитул - это текст или рисунок (*номер страницы, дата печати документа, эмблема организации, название документа, имя файла, фамилия автора и т.п.*), который печатается внизу или вверху каждой страницы документа.

В зависимости от места расположения (*на верхнем или на нижнем поле страницы*) колонтитулы бывают верхними и нижними.

Создание колонтитула

1. Выбрать команду **Вставка -Верхний** (*нижний*) колонтитул.
2. В поле колонтитула ввести текст или вставить необходимый текст с помощью кнопок (*Конструктор*) и по окончании нажать кнопку **Заккрыть**.

Лабораторное задание 1. Освойте основные приемы работы в текстовом редакторе **Ms Word**.

Порядок выполнения работы:

1. Запустите программу *Microsoft Word*, пользуясь меню **Пуск-Все программы-Microsoft Word**. На экране появится окно программы *Ms Word*.

2. Изучите структуру и элементы окна. Для этого необходимо нажать комбинацию клавиш **Shift+F1** и при помощи курсора выделяйте нужные элементы.

3. *Сверните* и *разверните* окно программы.

4. Создайте новый документ и затем *сверните*, и *разверните* окно документа.

5. Научитесь *устанавливать* и *убирать* панели инструментов и линейку при помощи команды **Вид** и с помощью контекстного меню.

6. Установите **Линейку** (*Вид-Линейка*).

7. Изучите содержимое строки состояния. *Выключите* и *включите* отображение строки состояния.

8. Создайте новый документ. В заголовке окна программы появится имя нового документа. Теперь в окне программы **Ms Word** открыто два документа: Создайте еще один новый документ. Научитесь переключаться между окнами документов и упорядочивать окна всех документов с помощью меню **Вид**.

9. Закройте окна всех документов.

Лабораторное задание 2. Форматирование информации в текстовом редакторе **Ms Word**. Изучите команду **Формат**, ее подкоманды **Шрифт**, **Абзац**, **Список**.

Создайте новый документ, содержащий *копию текста*, изображенного в таблице (3.31.1).

Строка	Шрифт		
Заголовок	Times New Roman, 14, полужирный		
Подзаголовок	Times New Roman, 14, полужирный курсив		
Основной текст	Times New Roman, 14		
Требования к формату абзацев			
Строка	Абзац		
	Выравнивание	Отступы, см	Интервалы, см





Заголовок	По центру	Слева - 0 Справа - 0 Первая строка - 0	Перед - 6 После - 6 Межстрочный - 1
Подзаголовок	По левому краю	Слева - 0 Справа - 0 Первая строка - 1 отступ 1	Перед - 3 После - 3 Межстрочный - 1
Основной текст	По ширине	Слева - 0 Справа - 0 Первая строка - 1	Перед - 0 После - 0 Межстрочный - 1

(табл.3.31.1). Шрифт, абзац

Основы форматирования в Ms Word

Шрифт.

Настройка формата выделенных символов осуществляется в диалоге Главная-Шрифт и включает такие характеристики:

1. **Шрифт:** (*Arial, Times, Courier, ...*);
2. **Размер:** (*от 1 до 1638*)
3. **Начертание:** (**Полужирный**, *Курсив*, Подчеркнутый);
4. **Зачеркнутый**
5. **Подстрочный знак:** (*ctrl+=*) 
6. **Надстрочный знак:** (*ctrl+sift++*) 
7. **Текстовые эффекты:** A A A A A
8. **Цвет выделения текста** 
9. **Цвет шрифта:** 
10. **Увеличить размер шрифта:** (*ctrl+!*)
11. **Уменьшить размер шрифта:** (*ctrl+()*)
12. **Регистр:** 
13. **Очистить все форматирование:** 

Абзац.

Формат абзаца (*меню Главная-Абзац*) включает такие параметры:

1. **Способ выравнивания:**

влево,

вправо,

по центру,

по ширине.

2. **Интервал:** (*Настройка интервалов между строками и абзацами*)
3. **Заливка:** (*Изменение цвета фона для выделенного текста, абзаца или ячейки таблицы*).
4. **Границы:** (*Добавление или удаление границ выделенных ячеек*).
5. **Маркеры:** (*Создание маркированного списка*).
6. **Нумерация:** (*Создание нумерованного списка*).
7. **Многоуровневый список:**
8. **Уменьшить отступ:** (*уменьшение расстояния от поля до абзаца*).
9. **Увеличить отступ:** (*увеличение расстояния от поля до абзаца*).
10. **Сортировка:** (*Сортировка по алфавиту или по числовым значениям*).
11. **Отобразить все знаки:** “¶”.

Указания:

Для вставки специального символа “¶” примените команду **Вставка-Символ-Другие символы**. Для нумерации строк необходимо строки выделить и нажать на кнопку нумерация в группе **Абзац**.

Задайте следующие *параметры страницы* с помощью меню **Макет-Параметры страницы** или линейки: *левое - 3,5; правое - 2,5; верхнее - 2; нижнее - 2*.

Для вставки рисунка перенесите текущую позицию редактирования в начало текста и создайте новый абзац. *Вставьте рисунок* при помощи команды **Вставка-Рисунки**.

Сохраните документ в своей папке под именем “*Лабораторное задание 2*”. Закройте документ и откройте его снова.

Лабораторное задание 3. Создайте многоуровневый список, указанный ниже:
Программное обеспечение ЭВМ.

1. Операционные системы

- 1.1. Windows
- 1.2. Mac.OS
- 1.3. Linux

2. Системы программирования

- 2.1. Pascal
- 2.2. Java
- 2.3. C++
- 2.4. Python

3. Прикладные программы

3.1. Текстовые редакторы

- 3.1.1. Microsoft Word
- 3.1.2. Word PAD
- 3.1.3. Блокнот

3.2. Электронные таблицы

- 3.2.1. Excel
- 3.2.2. Lotus
- 3.2.3. Quatropro

3.3. Системы управления базами данных

- 3.3.1. Foxpro
- 3.3.2. Access
- 3.3.3. Oracle

Методические указания.

Для построения этого списка наберите первую строку и выделите ее. Выполните команды **Главная-Абзац-Многоуровневый** и выберите нужный вид списка и нужную нумерацию. Установите курсор в конец первой строки и нажмите клавишу **Enter**. Добавленная строка будет иметь тот же уровень вложенности, что и предыдущая. Для увеличения уровня вложенности нажмите клавишу **Tab**, для уменьшения - **Shift+Tab**. Последовательно наберите нужные строки, устанавливая нужный уровень вложенности. В случае, если уровень вложенности будет увеличиваться не последовательно, уменьшите размер табуляции по умолчанию до **0,5см**.

Этот список можно построить и иначе. Для этого необходимо набрать только текст, нажимая в конце каждой строки клавишу **Enter**. Выделяя строки, находящиеся ниже первого уровня, сдвигаем их вправо на одну или две

позиции табулятора (*в зависимости уровня вложенности*) с помощью кнопки «**Увеличить отступ**» в группе **Абзац** или с помощью клавиши **Tab**. Затем выделите весь список и выполните команды **Список-Многоуровневый**.

Выберите нужную нумерацию и нажимаем кнопку **ОК**. В случае, если уровень вложенности не будет нужным, уменьшите размер табуляции по умолчанию до **0,5см**. повторите предыдущие действия.

Построить многоуровневый список можно и не используя табуляцию. В этом случае строки каждого уровня нужно набирать с помощью подчиненных стилей, например, **Заголовок 1**, **Заголовок 2** и **Заголовок 3**.



Контрольные вопросы и задания

1. Создайте в программе *Microsoft Word* свою *автобиографию*, состоящую минимум из трех абзацев. Создайте *пароль* для открытия вашего документа.
2. Для чего предназначена программа *Microsoft Word*?
3. Как вставить *пустую строку* в тексте? Как удалить *пустую строку* в тексте?
4. Назовите основные *этапы создания* текстового документа.
5. Чему равен *один пункт* размера шрифта?
6. Какие *виды списков* существуют в MS Word?
7. Каким образом можно *скопировать* текст?
8. Каким образом можно *переместить* текст?
9. Как вставить *номер страницы*?
10. Что такое *колонтитул*? Как его добавить в документ?

32-§. ТЕХНОЛОГИЯ ОБРАБОТКИ ДАННЫХ В ЭЛЕКТРОННЫХ ТАБЛИЦАХ.

Цель работы: **Формирование** *основных умений и навыков работы с табличным процессором Ms Excel.*

Краткая теоретическая информация:

Сортировка данных в Ms Excel – инструмент для представления информации в удобном для пользователя виде.

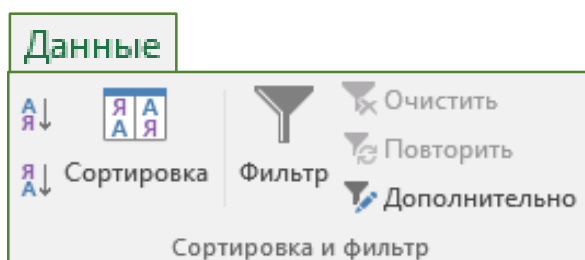
Сортировка данных в Ms Excel может производиться:

- ✚ в алфавитном порядке - от **А** к **Я** или от **Я** к **А**;
- ✚ по значениям - по **убыванию** или **возрастанию**;
- ✚ по **дате** и **времени**.

При этом сортировать данные в Ms Excel можно в *одном* или *нескольких* столбцах одновременно и, кроме этого, можно выполнять сортировку по *стискам*, которые создаются самим пользователем, а также по *цвету ячейки* или *текста*.

Если результаты сортировки данных не отвечают запросу и пользователь хотел бы отменить сортировку, то самым простым способом будет стандартная операция отмены последних действий. Необходимо просто нажать на соответствующую кнопку на **панели быстрого доступа** программы или использовать сочетание клавиш **Ctrl+Z**. Бывает так, что изменения в файле были сохранены и *отменить действие* невозможно.

Если после *сложных манипуляций* с таблицей ее нужно будет вернуть в первоначальный вид, перед выполнением сортировки следует специально создать *дополнительный столбец*, в котором будет отражена нумерация строк. После завершения сложного анализа *числовых* и *текстовых данных*, представленных в таблице, чтобы отменить все сделанные операции достаточно будет установить сортировку по созданному столбцу.



(рис.3.32.1). Сортировка и фильтр

Фильтрация в программе *Excel* является удобным инструментом, который позволяет быстро находить нужную информацию в большом массиве данных.

Фильтрация данных в *Ms Excel* позволяет отобразить среди большого объема информации только ту, что в данный момент *необходима*, а остальные *скрываются* до тех пор, пока не будет отменен **фильтр**.

Фильтры бывают двух типов: обычный фильтр (*его еще называют автофильтр*) и расширенный фильтр. Для применения автофильтра нужно нажать ту же кнопку, что и при сортировке - "Сортировка и фильтр" и выбрать пункт "Фильтр" (*перед этим должен быть выделен диапазон ячеек*).

1. **Автофильтр** - для отбора записей по значению ячейки, по формату или в соответствии с простым критерием отбора.

2. **Расширенный фильтр** - для фильтрации данных с помощью сложного критерия отбора.

Сделать **фильтр** довольно просто, нужно выполнить следующие последовательные шаги:

1. выделить *шапку таблицы*, где необходимо искать требуемое значение;
2. активировать вкладку "Данные";
3. в ней найти иконку в виде *воронки* и нажать на неё;
4. Далее начать искать по нужным условиям требуемое значения.

Если данные, которые нужно *отфильтровать*, должны иметь сложные условия, можно использовать диалоговое окно "*Расширенный фильтр*".

Основное отличие *фильтра* от *упорядочивания* - это то, что во время фильтрации записи, не удовлетворяющие условиям отбора, временно скрываются (*но не удаляются*), в то время, как при сортировке показываються все записи списка, меняется лишь их порядок.

Лабораторное задание 1. Сортировка в *Ms Excel*

По одному столбцу.

Чтобы отсортировать данные по одному столбцу, выполните следующие действия:

1. Создайте таблицу в соответствии с образцом, приведенным на рисунке. Сохраните ее под названием «*Сортировка данных*».

	Успеваемость студентов					
	Ф.И.О.	Лабораторная работа №1	Лабораторная работа №2	Лабораторная работа №3	Всего	Проценты
4	Рахимова Карина	3	4	5	12	6,6
5	Абдуганиева Юлдуз	3	3	4	10	5,5
6	Олимбекова Адиба	4	4	4	12	6,6
7	Абрамян Алексан	5	5	5	15	8,25
8	Шавкатова Нигора	4	3	4	11	6,05
9	Касымов Дилмурод	4	4	4	12	6,6
10	Иминова Зайнаб	2	3	4	9	4,95
11	Хамидов Днёр	3	4	3	10	5,5
12	Раззаков Нодир	4	4	4	12	6,6
13	Садиллаева Робия	3	4	3	10	5,5
14	Насриддинова Шахло	4	4	4	12	6,6
15	Зокиров Элёр	4	4	4	12	6,6

(рис.3.32.2). Сортировка по одному столбцу

2. Выделите любую ячейку в столбце, по которому нужно выполнить сортировку.

3. Для сортировки в порядке возрастания, откройте вкладку «*Главная*»-«*Редактирование*»-«*Сортировка и фильтр*»-«*Настраиваемая сортировка*» или на закладке: «*Данные*»-«*Сортировка и фильтр*»-«*Сортировка*» и кликните по иконке **АЯ** (AZ).

4. Результат:

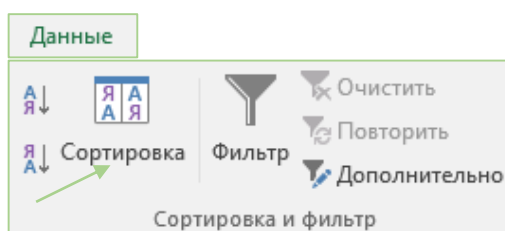
4	Абдуганиева Юлдуз
5	Абрамян Алексан
6	Зокиров Элёр
7	Иминова Зайнаб
8	Касымов Дилмурод
9	Насриддинова Шахло
10	Олимбекова Адиба
11	Раззаков Нодир
12	Рахимова Карина
13	Садиллаева Робия
14	Хамидов Диёр
15	Шавкатова Нигора

(рис.3.32.3). Результат сортировки

Примечание: Для сортировки в порядке убывания кликните иконку **ЯА** (ZA).
По нескольким столбцам.

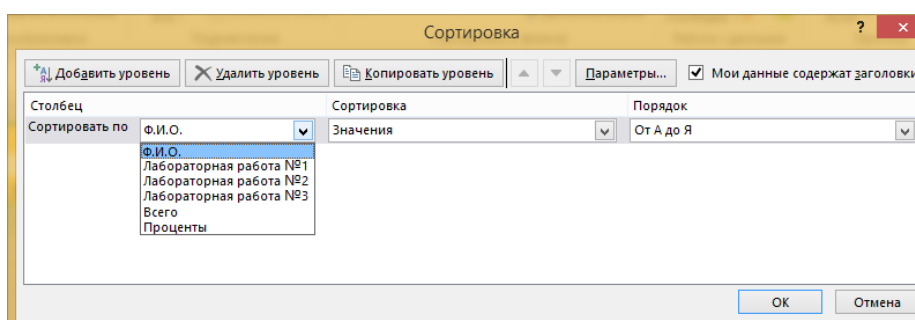
Чтобы отсортировать данные по нескольким столбцам, выполните следующие действия:

1. На вкладке **Данные** (*Data*) нажмите **Сортировка** (*Sort*). Откроется диалоговое окно **Сортировка** (*Sort*).



(рис.3.32.4). Диалоговое окно Сортировка

2. В выпадающем списке «**Сортировать по** (*Sort by*)» выберите столбец для сортировки (в нашем примере это **Ф.И.О.**).



(рис.3.32.5). Выпадающий список

3. Нажмите кнопку «**Добавить уровень** (*Add Level*)».
 4. В выпадающем списке «**Затем по** (*Then by*)» выберите следующий столбец для сортировки (выбираем **Лабораторная работа №1**).

5. Нажмите **ОК**. Результат записи отсортированы сначала по столбцу *Ф.И.О*, затем по столбцу *Лабораторная работа №1*.

6. Результаты *выполнения задания* покажите преподавателю!

Лабораторное задание 2.

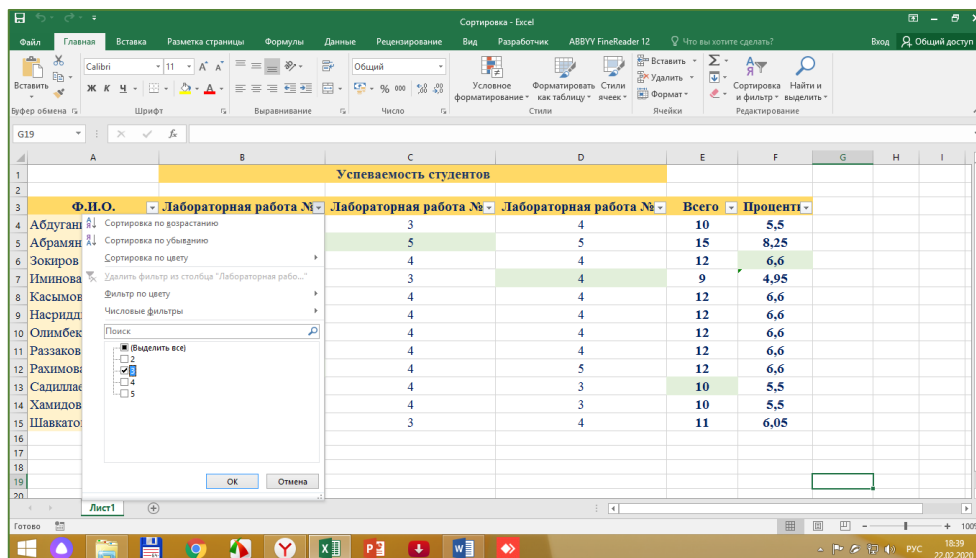
Ms Excel фильтрация (выборка) данных.

1. Установите курсор-рамку внутри таблицы данных (рис.3.32.6).
2. Выполните команду меню *Данные - Фильтр - Автофильтр*.
3. Снимите выделение в таблицы.

	A	B	C	D	E	F	
1		Успеваемость студентов					
2							
3	Ф.И.О.	Лабораторная работа №1	Лабораторная работа №2	Лабораторная работа №3	Всего	Проценты	
4	Абдуганиева Юлдуз	3	3	4	10	5,5	
5	Абрамян Алексан	5	5	5	15	8,25	
6	Зокиров Элёр	4	4	4	12	6,6	
7	Иминова Зайнаб	2	3	4	9	4,95	
8	Касымов Дилмурод	4	4	4	12	6,6	

(рис.3.32.6). Ms Excel, фильтрация

4. У каждой ячейки заголовка таблицы появилась кнопка "Стрелка вниз" (она не выводится на печать), нажав на которые можно настроить параметры фильтра.



(рис.3.32.7). Параметры фильтра.

5. Поля, по которым установлен фильтр, отображаются со значком воронки. Если подвести указатель мыши к такой воронке, то будет показано условие фильтрации.

	A	B	C	D	E	F	
1		Успеваемость студентов					
2							
3	Ф.И.О.	Лабораторная работа №	Лабораторная работа №	Лабораторная работа №	Всего	Проценти	
6	Зокпиров Элёр	4	4	4	12	5,6	
8	Касымов Дилмурод	4	4	4	12	5,6	
9	Насриддинова Шахло	4	4	4	12	6,6	
10	Олимбекова Адлиба	4	4	4	12	6,6	
11	Раззаков Нодир	4	4	4	12	6,6	
12	Рахимова Карина	3	4	5	12	6,6	
16							

(рис.3.32.8). Условие фильтрации

6. Для формирования более сложных условий отбора предназначен пункт "Текстовые фильтры" или "Числовые фильтры". В окне "Пользовательский автофильтр" необходимо настроить окончательные условия фильтрации.

7. При использовании расширенного фильтра критерии отбора задаются на рабочем листе.

8. Результаты выполнения задания покажите преподавателю!

Задание к лабораторной работе:

1. Скопируйте и вставьте на свободное место шапку списка. В соответствующем поле (полях) задайте критерии фильтрации.

2. Выделите основной список.

3. Нажмите кнопку "Фильтр" на панели "Сортировка и фильтр" ленты "Данные".

4. На той же панели нажмите кнопку "Дополнительно".

5. В появившемся окне "Расширенный фильтр" задайте необходимые диапазоны ячеек.

6. В результате отфильтрованные данные появятся в новом списке.

7. Расширенный фильтр удобно использовать в случаях, когда результат отбора желательно поместить отдельно от основного списка.



Контрольные вопросы и задания

1. Как выполнить сортировку записей?
2. Как отсортировать данные в списке?
3. Что такое пользовательский порядок сортировки?

4. Описать технологию отбора записей с помощью *Автофильтра*.
5. Когда нужно для отбора данных использовать *Расширенный фильтр*?
6. Как применить *Автофильтр* к списку?
7. Как отменить результаты фильтрации?
8. Сколько *условий* можно наложить на один столбец с помощью команды *Автофильтр*?
9. Каков порядок применения *Расширенного фильтра*?
10. Как отменить результат применения *Расширенного фильтра*?

33-§. СОЗДАНИЕ ОБУЧАЮЩИХ ПРЕЗЕНТАЦИЙ.

Цель работы: *Формирование умений и навыков, носящих в современных условиях общенаучный и обще интеллектуальный характер. Развитие у студентов теоретического, творческого мышления, индивидуального стиля.*

Краткая теоретическая информация:

Информатизация образования в наше время пронизывает все формы обучения. Компьютер стал неотъемлемой частью образовательного процесса. Одним из средств наглядного представления информации, сопровождающих выступление или знакомство с каким-либо объектом, является презентация.

Слово «презентация» в переводе с английского языка означает «представление», «показ», т.е. презентация - это красивый, наглядный показ какой-либо информации.

Компьютерная презентация состоит из отдельных кадров, которые называются *слайдами*. На каждом слайде можно поместить произвольную *текстовую, графическую и аудио информацию*, а также видеоклипы. Объекты на слайде могут быть анимированы. Презентацию можно показывать прямо на компьютере или выводить на большой экран через мультимедийный проектор.

Для создания компьютерных презентаций предназначены специальные программы. Одна из самых популярных программ - **PowerPoint**.

Функции презентации:

- ✚ Привлечь внимание, заинтересовать;
- ✚ Продемонстрировать серьезное отношение к делу, профессионализм;
- ✚ Дать самые важные ответы;
- ✚ Придать проекту эмоциональную окраску.

Структура презентации

- ✚ Подготовка презентации;
- ✚ Проведение;
- ✚ После презентационные исследования.

Правила выбора цветовой гаммы

- Цветовая гамма должна состоять не более чем из двух-трех цветов.
- Существуют не сочетаемые комбинации цветов.

Психологические свойства цветов

Цвет	Психологические свойства
<i>Красный</i>	возбуждающий, активный, богатый ассоциациями
<i>Голубой</i>	спокойный, содействует сентиментальному настроению, успокаивает нервную систему
<i>Желтый</i>	теплый, веселый, действует возбуждающе, вызывает желание действовать
<i>Оранжевый</i>	веселый, вызывает радость, усиливает активность
<i>Зеленый</i>	спокойный, создает прекрасное настроение, богат многочисленными ассоциациями
<i>Фиолетовый</i>	в равной мере привлекает и отталкивает, полон жизни, в некоторых случаях вызывает грусть
<i>Черный</i>	действует угнетающе, вызывает печаль
<i>Белый</i>	действует утомляюще, большие пространства, окрашенные исключительно в белый цвет, вызывают ощущение усталости

Правила общей композиции

- На полосе не должно быть больше семи значимых объектов, так как человек не в состоянии запомнить за один раз более семи пунктов чего-либо.
- Дизайн должен быть простым, а текст - коротким.
- Крупные объекты в составе любой композиции смотрятся довольно неважно. Аршинные буквы в заголовках, кнопки навигации высотой в 40

пикселей, верстка в одну колонку шириной в 600 точек, разделитель одного цвета, растянутый на весь экран - все это придает дизайну непрофессиональный вид.

- Не стоит забывать, что на каждое подобное утверждение есть сотни примеров, доказывающих обратное. Поэтому приведенные утверждения нельзя назвать общими и универсальными правилами дизайна, они верны лишь в определенных случаях.

Лабораторное задание. Создание теста в программе **PowerPoint** с использованием приложения **VBA** (*Visual Basic for Application*).

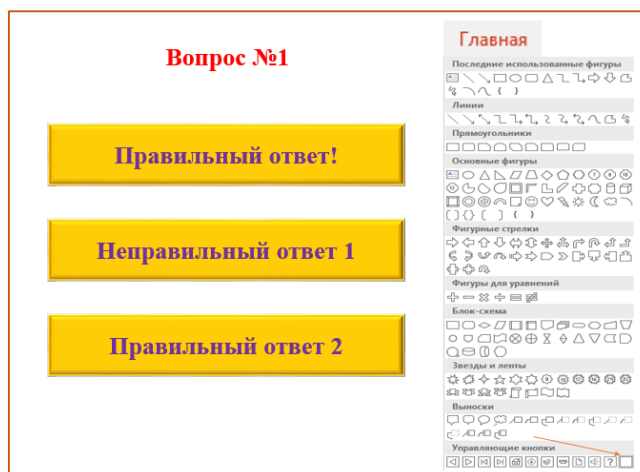
Очень эффектно при использовании **PowerPoint**, особенно в образовательном процессе, использование тестов. Например, задан вопрос с несколькими вариантами ответов. При правильном ответе есть возможность перейти к следующему слайду. При неверном ответе получить сообщение «**Неверно, попробуйте еще раз!**». Такие тесты можно создавать в **PowerPoint** с помощью **Visual Basic**.

Создание слайдов с вопросом и несколькими вариантами ответов

1. Первый шаг заключается в создании нескольких слайдов с вопросами. Введите первый вопрос в заголовке заполнителя. Перейдите в автофигуры и в разделе «*Управляющие кнопки*» выберите «*Управляющая кнопка: настраиваемая*»:

Щелкните по слайду. Появится диалоговое окно «*Настройка действия*» и кнопка. Окно закройте, щелкнув «*Отмена*», а кнопку разместите в нужное место, задав ей размер с учетом того, что туда будет вставлен текст. Щелкните по кнопке правой кнопкой мыши для вызова контекстного и выберите «*Изменить текст*».

Появится курсор ввода текста. Введите туда правильный вариант ответа. Продублируйте кнопку два раза и измените текст на неправильные варианты ответа. Таким образом сделайте еще два слайда. Должно получиться несколько слайдов примерно такого вида:



2. Когда слайды с *вопросами* и *ответами* готовы, переходите к созданию макросов. Для этого непосредственно в программе PowerPoint используйте редактор **Visual Basic**. В PowerPoint 2013 для перехода в редактор необходимо сначала открыть пункт меню - **«Разработчик»** (*по умолчанию, он не отображается.*) Для этого проделайте следующие шаги.

a. Откройте меню *«Настройка панели быстрого доступа»* - *«Другие команды»*:

b. В диалоговом окне *«Параметры PowerPoint»* выберите пункт *«Основные вкладки»* и ставьте галочку показывать вкладку **«Разработчик»** на ленте:

c. Переходите во вкладку **«Разработчик»** и выбирайте **«Visual Basic»**. Редактор откроется в новом окне. Создайте три макроса. Щелкните *«Insert»* - *«Module»* и вставьте код для **неправильного ответа**:

```
Sub Wrong()
```

```
MsgBox ("Ответ неверный. Попробуйте еще раз") End Sub
```

Далее - фрагмент кода для **правильного ответа**:

```
Sub Right()
```

```
MsgBox ("Все верно")
```

```
SHdeShowWindows( 1). View. Next End Sub
```

Фрагменты схожи, только во втором есть переход к следующему слайду. Для того, чтобы презентация не завершилась выходом на последнем слайде, добавьте для правильного ответа код:

Sub RightLast()

Msg Box ("Cong ra tula tions!")

End Sub

3. Переходите из редактора **VBA** в **PowerPoint**. Щелкните по значку «Смотреть (*View*) Microsoft PowerPoint».

Редактор Visual Basic оставьте открытым в фоновом режиме. В **PowerPoint** сохраните презентацию, это коснется и макросов. Сохраните как презентацию с поддержкой макросов.

Переходите к кнопке с правильным ответом. Щелкните правой кнопкой мыши на кнопку и выберите «**Гиперссылка**» из контекстного меню. Появится диалоговое окно «*Настройка действия*». Выберите «*Запуск макроса*» - «*Right (Правильно)*». Таким образом, подключили фрагмент кода для правильного ответа к кнопке.

Такую же процедуру проделайте для всех кнопок «*Неправильный ответ*», только подключайте макрос «*Wrong (Неправильный)*».

Прежде чем перейти к оставшимся слайдам и назначить соответствующие макросы на кнопках этих слайдов, проверьте работоспособность кнопок на первом слайде. Для этого запустите презентацию в *режим просмотра* и нажмите на неправильный ответ. Должно появиться сообщение «**Ответ неверный. Попробуйте еще раз**». Нажмите **ОК**, чтобы закрыть окно:

Проверьте другие неправильные ответы на слайде, чтобы убедиться, что они также работают. Наконец, выберите правильный ответ. На этот раз текстовое поле сообщение должно означать, что вы выбрали правильный ответ. Нажмите кнопку **ОК**, и *слайд-шоу* автоматически перейдет к следующему слайду.

Если первый слайд работает нормально, то проверьте остальные. На последнем слайде, назначьте для правильного ответа макрос «*RightLast*», чтобы по завершении слайда остаться на последнем слайде. Для правильного

ответа на последнем слайде должны получить сообщение «**Поздравляем!**» и остаться на этом слайде после нажатия кнопки **ОК**.

4. Для предотвращения пропуска в тесте вопросов без ответов выберите «Показ слайдов» - «Настройка демонстрации». В диалоговом окне «Настройка демонстрации» выберете «автоматический (полный экран)». Нажмите кнопку **ОК**. Эта установка дает возможность выйти из презентации только по клавише **ESC**.

Теперь перейдите в режим слайд-шоу и проверьте тест. Убедитесь, что все работает правильно.

5. Результаты выполнения задания покажите преподавателю!

Обратите внимание! При запуске файла с макросом, можно получить предупредительное сообщение о том, что файл содержит макросы. Это сделано в целях безопасности. Для того, чтобы макросы работали, необходимо выбрать "**Включить это содержимое**".



Контрольные вопросы и задания

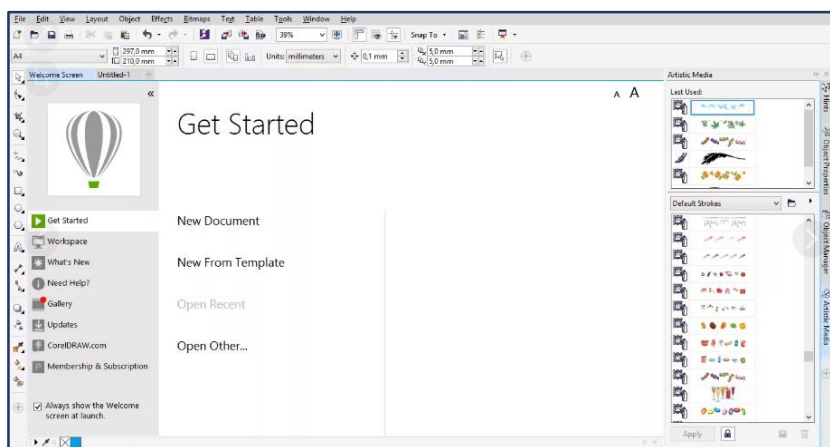
1. Что такое компьютерная презентация?
2. С каким расширением по умолчанию сохраняется файл презентации в MS Power Point 2013?
3. Где располагается и как настраивается панель быстрого доступа в окне MS Power Point 2013?
4. Что такое слайд? Из чего он состоит?
5. Как добавить на слайд текстовую надпись?
6. Как изменить положение текстовой надписи на слайде?
7. Как настроить автоматическую смену слайдов во время полноэкранный демонстрации презентации?
8. Какие действия можно настроить для объектов на слайдах?
9. Как создаются управляющие кнопки? Для чего их можно использовать?

34-§. НАСТРОЙКА ИНТЕРФЕЙСА CORELDRAW И СОЗДАНИЕ ИЗОБРАЖЕНИЕ.

Цель работы: знакомство с рабочей средой и интерфейсом пользователя, освоение функций элементов стандартной панели инструментов, приобретение навыков выравнивания объектов с помощью сетки и направляющих, освоение навыков изображения «примитивов» - прямоугольников, эллипсов, многоугольников, спиралей, сеток.

Краткая теоретическая информация:

Полностью интегрированные приложения и дополнительные подключаемые модули **CorelDRAW X7** позволяют работать в любом аспекте графического дизайна: от векторной графики и верстки страниц до редактирования фотографий, преобразования растровых изображений в векторные и проектирования веб-сайтов.



(рис.3.34.1). Основные компоненты экрана CorelDraw

Рабочее пространство CorelDRAW X7

Строка заголовка

Строка меню.

Линейки.

Стандартная панель инструментов

Панель свойств

Набор инструментов.

Окно рисования

Страница рисования.

Палитра документа.

Навигатор документов.

Строка состояния

Навигатор

Шаблоны.

Шаблоны позволяют быстро создать новый проект. Шаблоны можно выбирать, просматривать и искать по имени, категории, ключевым словам или примечаниям создателя.

Рисование линий.

Инструменты для рисования, доступные в выпадающем меню Кривая, позволяют создавать изогнутые и прямые линии, а также линии, состоящие как из изогнутых, так и из прямых сегментов. Сегменты линий соединяются с помощью узлов, которые изображаются небольшими квадратами

Рисование фигур.

В программе CorelDRAW доступны разнообразные инструменты для рисования фигур.

Текст.

В рисунки можно добавлять текст двух типов - простой и фигурный. Можно также загрузить готовый текст из внешнего файла или вставить текст из буфера обмена.

Палитра цветов.

CorelDraw предназначена для выбора цвета контуров и заливок. Для выбора цвета заливки щелкните палитру *левой кнопкой мыши*, а для выбора цвета контура-*правой кнопкой мыши*.



(рис.3.34.2). Палитра цветов программы CorelDraw

Для выравнивания объектов в документе используются средства выравнивания объектов: *сетку и направляющие*. Оба средства обладают возможностью привязки. Если привязка включена, *горизонтальное, вертикальное* или *наклонное выравнивание объектов* осуществляется автоматически.

Сетка.

Набор пунктирных линий, равномерно расположенных на экране. Для изменения шага и других параметров сетки можно воспользоваться окном диалога **Параметры**. Чтобы открыть это окно, необходимо выбрать команду Установка сетки в меню **Просмотр**.

Направляющие.

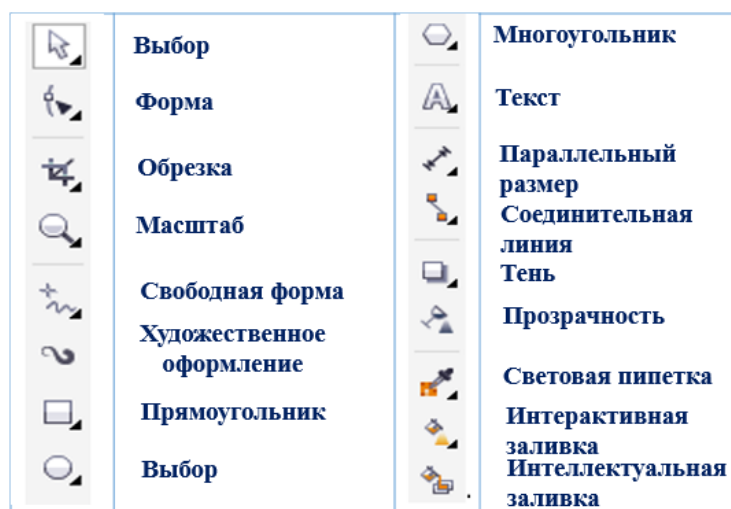
Горизонтальные, вертикальные и/или наклонные линии, которые используются для выравнивания объектов, но не выводятся на печать. Чтобы отобразить направляющие, нужно щелкнуть по линейке и перетащить указатель в окно документа. Перетаскивание будет сопровождаться появлением пунктирной голубой линии, предназначенной для выравнивания объектов. Для вывода направляющих можно воспользоваться окном диалога **Параметры**. Чтобы открыть это окно, нужно выбрать команду Установка, направляющую в меню **Просмотр**.

Линейки.

Линейки можно вывести слева от окна документа и над ним для упрощения задания положения и размеров объектов. Чтобы изменить положение начала координат $(0,0)$, необходимо щелкнуть кнопку, расположенную на пересечении линеек, а затем перетащить перекрестье линеек на нужное место. После отпускания кнопки мыши начало координат переместится в указанное место. Чтобы убрать линейки с экрана, нужно выбрать команду Линейки в меню **Вид**.

Стандартная Панель инструментов - основной элемент графического интерфейса CorelDraw.

Pick Tool (*Выбор*) - один из наиболее важных инструментов, который находится на панели **Toolbox** (*инструменты*).



(рис.3.34.3). Панель инструментов программы CorelDraw

При помощи этого инструмента можно выделять и различным образом преобразовывать выделенные объекты. Для выделения надо щелкнуть левой кнопкой мыши на выбранный вами объект, после чего вокруг него должны появиться восемь точек-маркеров, расположенных прямоугольником. Нажав и удерживая левую кнопку мыши и перемещая угловой маркер, можно изменять размер выделенного объекта, т.е. масштабировать его. Средними маркерами растягивают объект. Повторным нажатием на объекте маркеры изменяются на стрелочки, и теперь можно поворачивать объект, «зацепив» угловые стрелочки относительно его центра или центра, который можно задать, переместив небольшой кружок, появившийся вместе со стрелками. При помощи средних стрелочек объект наклоняют.

Все нарисованные линии векторной графики описываются координатами узлов и математическими уравнениями соединяющих их отрезков кривых.

Инструмент **Shape Tool** (*Форма*). Его основное назначение - редактировать узлы. Некоторые кнопки инструментов снабжены треугольником в нижнем правом углу. Это указатель, что с кнопкой связан не один, а несколько инструментов. Чтобы увидеть их все, после выполнения щелчка следует задержать кнопку мыши в нажатом состоянии - на экране раскроется панель конкретного инструмента.

Инструмент **Knife** (*Нож*) разделяет объект на части. Его подводят к контуру объекта и последовательно щелкают левой кнопкой в нужных точках. Если на панели инструментов включена кнопка, точки разрыва будут соединены новой линией и образуют новые замкнутые контуры. При выключенной кнопке образуются разомкнутые контуры.

Инструмент **Eraser** (*Ластик*) стирает часть объекта, при этом создаются новые узлы и объект становится значительно сложнее. Размер полосы стирания регулируется кнопкой панели атрибутов.

Инструмент **Zoom Tool** (*Масштаб*), при помощи которого можно изменять масштаб изображения. Использовать этот инструмент очень просто: левой кнопкой мыши щелкнуть столько раз, сколько требуется, или, удерживая левую кнопку мыши, выделить ту область, которую нужно приблизить.

Инструмент **Rectangle Tool** (*Прямоугольник*). С его помощью рисуются прямоугольники. Чтобы сделать это, нужно щелкнуть левой кнопкой мыши в точке расположения одного из углов и, продолжая удерживать левую кнопку, перетащить мышь по диагонали к противоположному углу прямоугольника. При нажатой клавише **Ctrl** размеры сторон автоматически выравниваются, и у вас в итоге получается правильный квадрат. При нажатой клавише **Shift** прямоугольник рисуется от центральной точки.

Инструмент **Ellipse Tool** (*Эллипс*). Этот инструмент позволяет рисовать эллипсы, дуги, сектора и может быть использован для рисования заготовок практически любых графических объектов.

Для того чтобы нарисовать эллипс, необходимо щелкнуть кнопкой мыши в точке одного из углов воображаемой прямоугольной области, охватывающей эллипс, и начать перетаскивать мышь в точку расположения другого угла.

Потом отпустить кнопку мыши, и в строке состояния отобразятся различные параметры нарисованного вами эллипса, например, координаты центра, высота, ширина, и т.п.

Инструмент **Spiral Tool** (*Спираль*) предназначен для рисования спирали.

Перед тем, как приступить к рисованию спирали, надо настроить ее параметры на верхней панели управления - панели атрибутов. В CorelDraw существует два вида спиралей:

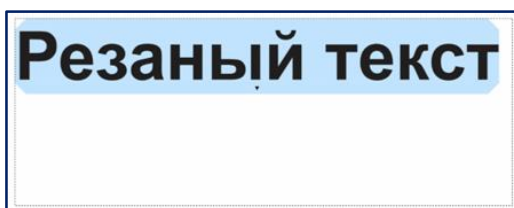
- *логарифмическая спираль*, у нее каждый следующий виток все дальше отстоит от предыдущего, причем степень нарастания шага спирали зависит от коэффициента расширения, управляемого ползунком, расположенным на панели атрибутов;
- *симметричная спираль*, каждый виток которой отстоит на одинаковое расстояние.

Инструмент **Polygon Tool** (*Многоугольник*) рисует самый сложный из всех объектов-«примитивов» - многоугольник. Этот же инструмент после изменения некоторых его параметров может рисовать различные звезды и звездообразные фигуры.

На панели атрибутов (*свойств*) определенного объекта представлены элементы управления, содержащие параметры модели объекта, и кнопки, позволяющие выполнять стандартные действия над объектами этого класса.

Лабораторное задание. Создание резаного текста в CorelDRAW

1. Ввести текст с помощью инструмента «Текст» (*F8*), сделать его полужирным (*Ctrl+B*) и разместить по центру (*Ctrl+E*).



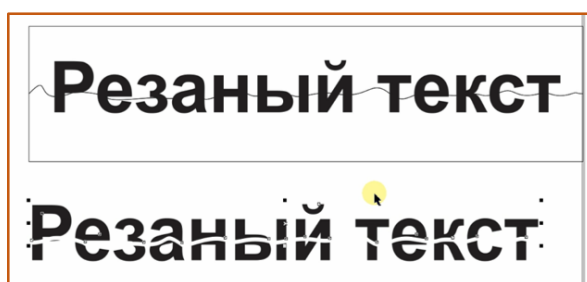
2. Преобразовать текст в кривую (*Ctrl+Q*).
3. Выбрать инструмент «Прямоугольник» (*F6*) и взять текст в прямоугольник.
4. Поставить прямоугольник на задний план страницы (*Порядок-На задний план страницы, или Ctrl+End*).

5. С помощью инструмента **«Нож»** (*Обрезка Нож*) или **«Knife»** разрезать текст.

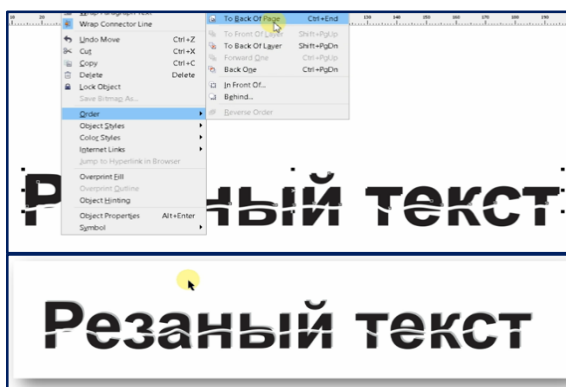
6. Строка меню–*Окно-Окна настройки-Формирование-Пересечение* (*Window-Dockers-Shaping-Intersect*).

7. Выбрать текст и нажать **«Пересечь»** (*«Intersect with»*), затем выбрать верхнюю часть прямоугольника. Снова выбрать текст и нажать **«Пересечь»** (*«Intersect with»*), затем выбрать нижнюю часть прямоугольника.

8. Перетащить отделённые верхнюю и нижнюю части текста. Получается следующее:



9. Убрать текст, который расположен в прямоугольнике. Дублировать верхнюю часть текста, расположить её чуть выше основной части для придания ей тени. Новый слой верхней части текста поставить на задний план страницы (*Порядок-На задний план страницы, или Ctrl+End*) и придать ему цвет **«Черный 30%»**. То же самое сделать и с нижней частью текста.



(рис.3.34.4). Результат работы



Контрольные вопросы и задания

1. Загрузить редактор CorelDRAW.

1. Изучить назначение кнопок палитры инструментов и их особенности.
2. Изучить назначение кнопок панели атрибутов, экранной палитры цветов.
3. Объяснить сведения, выводимые в строке состояния.
4. Получить изображения четырех типов прямоугольников. Выполнить для этого соответствующие операции.
5. Получить закругление одного и нескольких углов прямоугольника.
6. Построить изображение эллипса, сектора, дуги.
7. Выполнить построение многоугольников различных типов и их модификаций.

35-§. СОЗДАНИЕ БАЗЫ ДАННЫХ В MS. ACCESS.

Цель работы: *Освоение приемов проектирования базы данных, описания структуры таблиц и связей между ними.*

Краткая теоретическая информация:



Реляционная система управления базами данных (СУБД) корпорации Microsoft. Входит в состав пакета Microsoft Office. Имеет широкий спектр функций, включая связанные запросы, связь с внешними таблицами и базами данных. Благодаря встроенному языку **VBA**, в самом Access можно писать приложения, работающие с базами данных.

Пользовательский интерфейс Microsoft Office Access в версии 2013 был изменен в пользу упрощения восприятия информации и ускорения нахождения тех или иных инструментов. Достоинством Microsoft Access является то, что он имеет простой графический интерфейс, который позволяет не только создавать базу данных, но и разрабатывать приложения.

1. **Лента.** Новый тип интерфейса пользователя, пришедший на смену традиционным меню и панелям инструментов.

2. **Панель быстрого доступа.** На ней расположены кнопки наиболее часто применяемых действий. Расположение панели и наличие на ней кнопок можно настраивать по желанию пользователя.

3. **Область навигации** (*Область переходов*). В версии Access данная область заменила собой окно базы данных предыдущих версий.

По умолчанию область навигации (*область переходов*) отображается при открытии базы данных в Office Access 2013, заменяя окно базы данных, использовавшееся в более ранних версиях Microsoft Access. Если для выполнения определенной задачи в предыдущей версии Access предназначалось окно базы данных, в новой версии используется область переходов. Например, *если нужно выполнить отчет или ввести данные непосредственно в новую строку, следует открыть таблицу из области переходов.*

Выбор представления для того, чтобы работать с информацией на должном уровне, необходим набор инструментов. Основными и наиболее часто используемыми являются следующие представления объектов базы данных: **Режим таблицы** (*Datasheet View*), **Режим формы** (*Forms View*), **Конструктор** (*Design View*), **Режим макета** (*Layout View*). **Режим таблицы** (*Datasheet View*) дает возможность вводить информацию в таблицу, редактировать ее структуру, добавлять новые поля.

Лабораторное задание. Создание таблиц в *режиме конструктора*. Создание форм.

Сортировка и отбор данных.

1. Откройте программу **Ms Access**, создайте новую базу данных, сохраните в своей папке с именем **Студент**. Изучите *окно базы данных*, найдите вкладки объектов, ознакомьтесь с *формой представления данных* и панелью инструментов.

2. **В режиме конструктора** создайте макет таблицы, содержащей следующие поля:

Код студента	Фамилия	Имя	Группа	Адрес	Телефон	Фото	Примечания

Имена полей задайте также как в (табл.3.35.1).

Свойства полей:

№	Тип	Подпись Поля	Размер поля	Подпись	Маска ввода	
Код студента:	<i>Счетчик</i>	<i>№</i>	<i>индексированное, без повторений.</i>			
Фамилия:	<i>Краткий текст</i>		<i>50</i>	<i>обязательное, без пустых строк, индексированное.</i>		
Имя:	<i>Краткий текст</i>		<i>25</i>	<i>обязательное, без пустых строк.</i>		
Группа:	<i>Краткий текст</i>		<i>7</i>	<i>№ Группы</i>	<i>КОМ-999</i>	<i>обязательное, без пустых строк, индексированное.</i>
Адрес:	<i>Краткий текст</i>		<i>200</i>	<i>не обязательное, не индексированное.</i>		
Телефон:	<i>Краткий текст</i>		<i>8</i>		<i>99999-99-99</i>	<i>не обязательное, не индексированное.</i>
Фото:	<i>OLE – объект.</i>					
Примечания:	<i>Длинный текст.</i>					

(табл.3.35.1). Свойства полей

Закройте таблицу, присвойте имя **Студенты**.

3. Для ввода данных в таблицу создайте автоформу в столбец с именем **Формуляр**. Она отображает все поля таблицы. Обратите внимание, как используются подписи полей. Откройте форму и введите *10-12 записей*, проверяя правильность описания полей. В данных должны быть *повторения фамилий, номеров групп (используйте три номера группы)*. Для ввода фото используйте вставку объекта из **Clip Gallery**. Если необходимо сделать

исправления, *откройте таблицу в режиме конструктора* и внесите необходимые изменения в макет таблицы.

4. Войдите в режим *конструктора форм*. Отредактируйте формуляр: текстовые поля расположены в столбце слева один под другим, фотографию и примечания поместите справа один под другим. Уменьшите размер полей фотографии и примечаний, а также измените в макете формы свойство рамки объекта **«Установка размеров»** на значение *«По размеру рамки»*. Окно свойства вызывается нажатием правой кнопки мыши на активном элементе формы.

5. Для просмотра *записей таблицы* создайте с использованием **мастера форм** ленточную форму, не включая в нее фотографии и примечания. Задайте имя **Ленточная форма**.


6. Выполните в этой форме операции, характерные для баз данных.

а. **Сортировка записей** выполняется по значению одного поля, активного в данный момент.

Выполните сортировку записей:

- по фамилиям,
- по номерам групп,
- по номерам телефонов.


б. Отбор записей выполняется с использованием фильтров. Чтобы задать условия отбора, используется кнопка .

Чтобы выполнить отбор или отменить условия отбора, используется кнопка . В бланке отбора можно задать условия, выполняемые одновременно (*задаются в одной строке бланка*) или условия, выполняемые порознь (*задаются через закладку «ИЛИ»*). Условия отбора могут быть выбраны из списка или записаны в виде отношений или логических выражений.

Выполните последовательно отбор:

- Всех студентов, учащихся в одной группе (*например, 201*).

- Всех студентов, учащихся в двух других группах (*например, 202 и 203*).
- Студента, *например, Ахмедова*, учащегося в указанной группе.
- Всех студентов, фамилии которых начинаются на букву «А» или «Б». Используется маска *, например, **А*** или **Б***.

с. Пункт меню **Записи** – *Расширенный фильтр* позволяет выполнить отбор записей с использованием бланка фильтра (*запроса*). В бланке фильтра будет добавлена таблица **Студенты**. Для фильтра можно задать поля, по которым выполняется *отбор, сортировки по значениям полей, и условия отбора*. Для отбора данных поле таблицы двойным щелчком переносится в нижнюю часть бланка. В строке «**Сортировка**» выбирается способ сортировки по указанному полю. В строке «**Условие отбора**» накладывается произвольное условие на значения поля. Если условия записаны в одной строке бланка, они накладываются друг на друга, то есть должны быть выполнены одновременно. Для записи условий, выполняемых *врозь*, используется строка бланка «**ИЛИ**». Для записи условий используется построитель выражений, вызываемый правой кнопкой мыши в области условия . Вызовите построитель выражений и ознакомьтесь с его окном. Для сравнения текстовых строк используется операция **Like**, например, *Like Ахмедов, или Like А**.

d. Выполните последовательно отбор данных, как в пункте **b** Каждый результат отбора сохраните (*меню Файл - Сохранить как*) в форме отчета с именами, соответственно, *Фильтр 1 – Фильтр 4*.

7. Результаты работы показать преподавателю.



Контрольные вопросы и задания

1. Какое расширение имеет файл *СУБД MS Access*?
2. Объекты, с которыми работает СУБД и их назначение.
3. Режимы работы с объектами СУБД.

4. Типы данных для полей
5. Типы создаваемых запросов в Access.
6. Когда приложение MS Access создает файл для работы с приложением?
7. Основными объектами СУБД MS Access являются?
8. Какие типы данных используются в СУБД MS Access для хранения *больших объемов текста*?
9. Какие типы данных используются в СУБД MS Access для хранения *графических объектов*?

ГЛАВА 4. КОМПЬЮТЕРНЫЕ СЕТИ.

МОДУЛЬ 4

36-§. КОМПЬЮТЕРНЫЕ СЕТИ: ПОНЯТИЕ И ВОЗМОЖНОСТИ.

Ключевые слова: компьютерная сеть, локальные сети, корпоративная сеть, глобальная сеть, трафик, Интернет, сервер, клиент.

Компьютерная сеть – это совокупность компьютерного и сетевого оборудования, соединенного с помощью каналов связи в единую систему.

Для создания компьютерной сети потребуются следующие компоненты:

- компьютеры, имеющие возможности для подключения к сети (например, сетевая карта, которая есть в каждом современном ПК);
- передающая среда или каналы связи (кабельные, спутниковые, телефонные, волоконно-оптические и радиоканалы);
- сетевое оборудование (например, коммутатор или роутер);
- сетевое программное обеспечение (как правило, входит в состав операционной системы или поставляется вместе с сетевым оборудованием).



(рис4.36.1). Компьютерная сеть

Компьютерные сети принято подразделять на два основных вида: глобальные и локальные.

Локальная сеть (*локальная вычислительная сеть или LAN*) - компьютерная сеть, покрывающая обычно относительно небольшую территорию или небольшую группу зданий (дом, офис, университет).

Локальные сети обладают замкнутой инфраструктурой до выхода на поставщиков услуг интернета. Термин “*локальная сеть*” может описывать и маленькую офисную сеть, и сеть университета, занимающего несколько гектаров. Применительно к организациям, предприятиям, фирмам используется термин **корпоративная сеть** – локальная сеть отдельной организации (*юридического лица*) независимо от занимаемой ею территории.

Корпоративная сеть - *коммуникационная система, принадлежащая и/или управляемая единой организацией в соответствии с правилами этой организации.*

Локальные – сети (*вычислительные*), абоненты которых сосредоточены на расстоянии 10 – 15км.

Преимущества локальных сетей:

Разделение ресурсов - позволяет экономно использовать ресурсы в информационной системе. Например, производить печать со всех компьютеров на одном принтере, использовать один дисковод DVD и т.д.

Разделение данных - позволяет иметь доступ с разных рабочих мест к файлам, которые расположены на других компьютерах. Благодаря разделению данных можно организовать работу нескольких пользователей по созданию общего документа.

Разделение программных средств - позволяет пользователям использовать программы, установленные на других компьютерах.

Топология локальных сетей

Под топологией вычислительной сети понимается способ соединения ее отдельных компонентов (*компьютеров, серверов, принтеров и т.д.*).
Различают три основные топологии:

- ✓ топология типа звезда;
- ✓ топология типа кольцо;
- ✓ топология типа общая шина (*линейная*).

При использовании топологии типа **звезда** информация между клиентами сети передается через единый центральный узел. В качестве центрального узла может выступать сервер или специальное устройство - концентратор (*Hub*).



(рис.4.36.2). Топология «звезда» (*Star*)

Преимущества данной топологии состоят в следующем:

1. Высокое быстродействие сети, так как общая производительность сети зависит только от производительности центрального узла.
2. Отсутствие столкновения передаваемых данных, так как данные между рабочей станцией и сервером передаются по отдельному каналу, не затрагивая другие компьютеры.

Однако помимо достоинств у данной топологии есть и недостатки:

1. Низкая надежность, так как надежность всей сети определяется надежностью центрального узла. Если центральный компьютер выйдет из строя, то работа всей сети прекратится.
2. Высокие затраты на подключение компьютеров, так как к каждому новому абоненту необходимо ввести отдельную линию.

При топологии типа **кольцо** все компьютеры подключаются к линии, замкнутой в кольцо. Сигналы передаются по кольцу в одном направлении и проходят через каждый компьютер.

Передача информации в такой сети происходит следующим образом. Маркер (*специальный сигнал*) последовательно, от одного компьютера к другому, передается до тех пор, пока его не получит тот, которому требуется передать данные. Получив маркер, компьютер создает так называемый "*пакет*", в который помещает адрес получателя и данные, а затем отправляет этот пакет по кольцу. Данные проходят через каждый компьютер, пока не окажутся у того, чей адрес совпадает с адресом получателя.

После этого принимающий компьютер посылает источнику информации подтверждение факта получения данных. Получив подтверждение, передающий компьютер создает новый маркер и возвращает его в сеть.



(рис.4.36.3). Топология «Кольцо» (Ring)

Преимущества топологии типа кольцо состоят в следующем:

1. Пересылка сообщений является очень эффективной, т.к. можно отправлять несколько сообщений друг за другом по кольцу. Т.е. компьютер, отправив первое сообщение, может отправлять за ним следующее сообщение, не дожидаясь, когда первое достигнет адресата.

2. Протяженность сети может быть значительной. Т.е. компьютеры могут подключаться к друг к другу на значительных расстояниях, без использования специальных усилителей сигнала.

К недостаткам данной топологии относятся:

1. Низкая надежность сети, так как отказ любого компьютера влечет за собой отказ всей системы.

2. Для подключения нового клиента необходимо отключить работу сети.

3. При большом количестве клиентов скорость работы в сети замедляется, так как вся информация проходит через каждый компьютер, а их возможности ограничены.

4. Общая производительность сети определяется производительностью самого медленного компьютера.

При топологии типа **общая шина** (*линейная*) все клиенты подключены к общему каналу передачи данных. При этом они могут непосредственно вступать в контакт с любым компьютером, имеющимся в сети.

Передача информации в данной сети происходит следующим образом. Данные в виде электрических сигналов передаются всем компьютерам сети. Однако информацию принимает только тот компьютер, адрес которого соответствует адресу получателя. Причем в каждый момент времени только один компьютер может вести передачу данных.

Преимущества топологии общая шина:

1. Вся информация находится в сети и доступна каждому компьютеру.

2. Рабочие станции можно подключать независимо друг от друга. Т.е. при подключении нового абонента нет необходимости останавливать передачу информации в сети.

3. Построение сетей на основе топологии общая шина обходится дешевле, так как отсутствуют затраты на прокладку дополнительных линий при подключении нового клиента.

4. Сеть обладает высокой надежностью, т.к. работоспособность сети не зависит от работоспособности отдельных компьютеров.

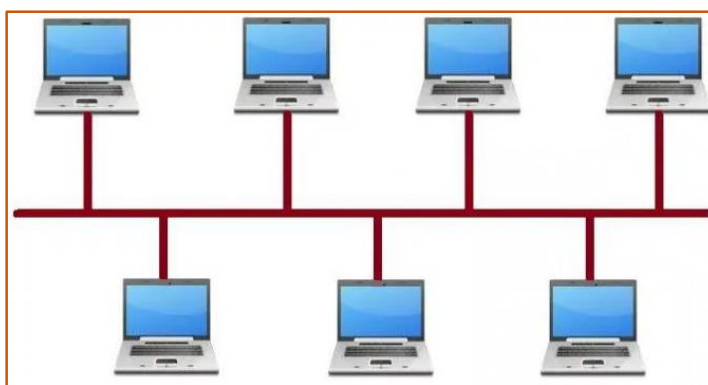
К недостаткам топологии типа общая шина относятся:

1. Низкая скорость передачи данных, т.к. вся информация циркулирует по одному каналу (*шине*).

2. Быстродействие сети зависит от числа подключенных компьютеров. Чем больше компьютеров подключено к сети, тем медленнее идет передача информации от одного компьютера к другому.

3. Для сетей, построенных на основе данной топологии, характерна низкая безопасность, так как информация на каждом компьютере может быть доступна с любого другого компьютера.

Самым распространенным типом сети с топологией общая шина является **сеть стандарта** со скоростью передачи информации *10 - 100 Мбит/сек.*



(рис.4.36.4). Топология «шина» (Bus)

Типы локальных сетей

Существует две модели локальных вычислительных сетей:

- ✚ одноранговая сеть;
- ✚ сеть типа клиент-сервер.

В одноранговой сети пользователю, работающему за любым компьютером *доступны ресурсы всех других компьютеров сети.* Например, сидя за одним компьютером, можно редактировать файлы, расположенные на другом компьютере, печатать их на принтере, подключенном к третьему, запускать программы на четвертом.

В сетях типа клиент-сервер имеется один (или несколько) главных компьютеров - серверов. Серверы используются для хранения всей информации в сети, а также для ее обработки. В качестве достоинств такой модели следует выделить:

- ✓ высокое быстродействие сети;
- ✓ наличие единой информационной базы;
- ✓ наличие единой системы безопасности.

Однако у данной модели есть и *недостатки*. Главный недостаток заключается в том, что стоимость создания сети типа *клиент-сервер* значительной выше, за счет необходимости приобретать специальный сервер. Также к недостаткам можно отнести и наличие дополнительной потребности в обслуживающем персонале - *администраторе сети*.

Глобальная сеть - это объединение компьютеров, расположенных в разных точках планеты для совместного использования информационных ресурсов человечества

Глобальная сеть (*Wide Area Network - WAN*) охватывает большие географические регионы и состоит из множества локальных сетей. С глобальной сетью, которая состоит из нескольких тысяч сетей и компьютеров, знакомы все - это **Интернет**.



(рис.4.36.5). Глобальная сеть «WAN» (Интернет)

Компьютер, предоставляющий свои ресурсы для общего использования другим компьютерам сети, называется **сервером**; а компьютер, обращающийся к совместно используемым ресурсам на сервере – **клиентом**.

Существуют различные **виды серверов**: файловые (*для хранения общих файлов*), серверы баз данных, серверы приложений (*обеспечивающие*

удаленную работу программ на клиентах), web-серверы (для хранения web-контента) и другие.

Загрузка сети характеризуется параметром, называемым трафиком. **Трафик** – это поток сообщений в сети передачи данных. Под ним понимают количественное измерение числа проходящих по сети блоков данных и их длины, выраженное в битах в секунду. Например, скорость передачи данных в современных локальных сетях может быть *100Мбит/с* или *1Гбит/с*.

В настоящее время в мире насчитывается огромное количество всевозможного сетевого и компьютерного оборудования, позволяющего организовать самые различные компьютерные сети. Все многообразие компьютерных сетей можно разделить на несколько видов по различным признакам:

По территории:

- локальные – охватывают небольшие территории и располагаются внутри отдельных офисов, банков, корпораций, домов;
- региональные – образуются путем объединения локальных сетей на отдельных территориях;
- глобальные (*интернет*).

По способу связи компьютеров:

- проводные (*компьютеры соединяются посредством кабеля*);
- беспроводные (*компьютеры обмениваются информацией посредством радиоволн. например, по технологии **WI-FI** или **Bluetooth***).

По способу управления:

- с централизованным управлением – для управления процессом обмена данных в сети выделяется одна или несколько машин (*серверов*);
- децентрализованные сети – не содержат в своем составе выделенных серверов, функции управления сетью передаются по очереди от одного компьютера другому.

По составу вычислительных средств:

- однородные – объединяют однородные вычислительные средства (компьютеры);

- неоднородные – объединяют различные вычислительные средства (например, ПК, торговые терминалы, веб-камеры и сетевое хранилище данных).

По типам среды передачи сети разделяются на оптоволоконные, с передачей информации по радиоканалам, в инфракрасном диапазоне, через спутниковый канал и т.д.

Существуют другие классификации компьютерных сетей. Как правило, системному администратору приходится иметь дело с локальными проводными сетями с централизованным, либо децентрализованным управлением.



Контрольные вопросы и задания

1. Что такое компьютерная сеть?
2. Виды компьютерных сетей.
3. Функции локальных сетей.
4. Компоненты компьютерных сетей
5. Локальные сети
6. Корпоративная сеть
7. Глобальная сеть
8. Виды серверов

37-§. ОРГАНИЗАЦИЯ СЕТИ ИНТЕРНЕТ.

Ключевые слова: *arpanet, интернет, tcp/ip, www, трафик, IP адрес, сервер, домен, электронная почта.*

Что такое интернет?

Интернет - *всемирная система, состоящая из объединенных компьютерных сетей на базе протокола TCP/IP.*

Это невероятное изобретение 20 века с последующим развитием в 21 веке. Сейчас интернет - это не просто сеть, *а целая информационная вселенная*, подчиняющаяся *техническим, социальным и государственным законам* в различных ее частях.

Зарождение Интернета принято считать с момента появления первой компьютерной сети, родиной которой в середине 60-х годов двадцатого века стала Америка. В то время еще не существовало персональных компьютеров, и крупные американские университеты могли себе позволить 1-2 больших компьютера. Компьютерное время было драгоценным ресурсом, и на него заранее записывались. Люди работали ночами, чтобы ни минуты этого времени не пропало даром.

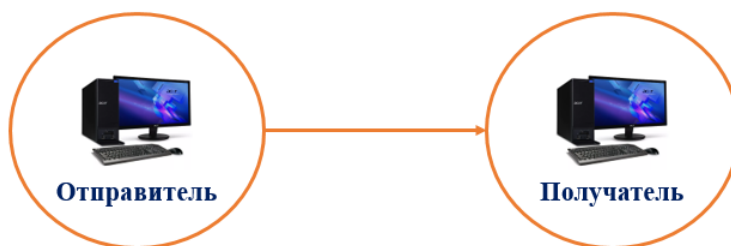
Наконец появилась идея соединить между собой компьютеры разных университетов, чтобы сделать возможным удаленное использование любого свободного в данный момент компьютера. Этот проект получил название **ARPANET**. К концу 1969 года были соединены компьютеры четырех университетов и появилась *первая компьютерная сеть*.

Очень скоро обнаружилось, что сеть в основном используется не для вычислений на удаленном компьютере, а для обмена сообщениями между пользователями. В 1972 году, когда ARPANET уже соединял 23 компьютера, была написана первая программа для обмена электронной почтой по сети. Электронную почту оценили по достоинству, что побудило целый ряд государственных организаций и корпораций к созданию собственных компьютерных сетей. Эти сети обладали тем же недостатком, что и ARPANET: они могли соединять только ограниченное число однотипных компьютеров. Кроме того, они были не совместимы друг с другом.

В середине 70-х годов для ARPANET были разработаны новые стандарты передачи данных, которые позволяли объединять между собой сети произвольной архитектуры, тогда же было придумано слово "**Интернет**". Именно эти стандарты, впоследствии получившие название протокола **TCP/IP**, заложили основу для роста глобальной компьютерной сети путем

объединения уже существующих сетей. Их важным достоинством было то, что сеть считалась в принципе не стопроцентно надежной, и предусматривались средства борьбы с ошибками при передаче данных. В 1983 году сеть ARPANET перешла на новый протокол и разделилась на две независимые сети - военную и образовательную. К этому времени сеть объединяла более тысячи компьютеров, в том числе в Европе и на Гавайских островах. Последние использовали спутниковые каналы связи.

Протоколы TCP/IP



(рис.4.37.1). Протоколы TCP/IP

TCP – определяет, как происходит передача информации,

IP – куда происходит передача

Стек протоколов TCP/IP

Уровни модели OSI	Протоколы TCP/IP	Уровни TCP/IP
Прикладной	FTR, TFTR, SMTP, SNMP, HTTP	I
Транспортный	TCP, UDP	
Сеансовый	TCP, UDP	II
Сетевой	IP, ICMP	III

(табл.4.37.1). Стек протоколов TCP/IP

Развитие Интернета получило новый импульс благодаря инициативе Национального научного фонда США (NSF) по созданию глобальной сетевой инфраструктуры для системы высшего образования (1985-88). NSF создал сеть скоростных магистральных каналов связи и выделял средства на подключение к ней американских университетов, при условии, что университет обеспечивал доступ к сети для всех подготовленных пользователей. Интернет оставался преимущественно университетской сетью до начала 90-х годов, однако NSF сразу взял курс на то, чтобы сделать его в дальнейшем независимым от

государственного финансирования. В частности, NSF поощрял университеты к поиску коммерческих клиентов. К 1988 году Интернет уже насчитывал около 56 тысяч соединенных компьютеров.

Настоящий расцвет Интернета начался в 1992 году, когда была изобретена новая служба, получившая странное название «Всемирная паутина» (*World Wide Web, или WWW, или просто «веб»*). WWW позволял любому пользователю Интернета публиковать свои текстовые и графические материалы в привлекательной форме, связывая их с публикациями других авторов и предоставляя удобную систему навигации. Постепенно Интернет начал выходить за рамки академических институтов и стал превращаться из средства переписки и обмена файлами в гигантское хранилище информации. К 1992 году Интернет насчитывал более миллиона соединенных компьютеров.

В настоящее время Интернет продолжает расти с прежней головокружительной скоростью. По оценке специалистов, количество передаваемой информации (*трафик*) в Интернете увеличивается на 30% ежемесячно. В 1999 году Интернет объединял около 60 миллионов компьютеров и более 275 миллионов пользователей, и каждый день в нем появлялось полтора миллиона новых веб-документов. Эти оценки довольно приблизительны, потому что в Интернете нет центрального административного органа, который регистрировал бы новых пользователей и новые компьютеры.

Принципы и организация сети Интернет

IP-адресация

Как в телефонной сети каждому телефону присваивается телефонный номер, так и каждому компьютеру в Интернете присваивается свой номер, который называется **IP-адресом**.

IP адрес - уникальный идентификатор, приписываемый каждому устройству, которое подключается к глобальной сети.

Каждый IP-адрес имеет длину ровно 32 бита и записывается обычно как четыре десятичных числа (*от 0 до 255*), - например, **62.76.161.102**. Это глобальная нумерация - каждый компьютер, подключенный к Интернету, имеет уникальный IP-адрес.

Чтобы передать длинное сообщение, оно разбивается на нужное число кусочков, и каждый из них снабжается адресом отправителя, адресом получателя и некоторой служебной информацией. Каждый пакет передается по Интернету независимо от всех остальных и, в принципе, они могут следовать разными маршрутами. По прибытии пакетов на место из них собирается исходное сообщение. Это называется *коммутацией пакетов*.

Одним из главных преимуществ режима коммутации пакетов это - эффективное использование общих коммуникационных ресурсов. В Интернете каждый компьютер может одновременно принимать пакеты от большого количества других компьютеров. При этом возможны перегрузки коммутационных узлов (*серверов*) в результате большого количество информации.

Сервер - это специализированный компьютер с особым программным обеспечением, который выполняет служебные функции. Собственно, в переводе с английского *to serve* значит "служить".

Однако все пакеты, пусть с небольшой задержкой, все равно дойдут до адресата в порядке своей очереди. В то же время, если в данный момент вы никакой информации не посылаете, то вы не потребляете никаких ресурсов сети, и тем самым можете находиться «*на линии*» сколь угодно долго, не создавая помех другим.

Для обеспечения жизнеспособности всей сети необходимо, чтобы обмен информацией между различными ее блоками или отдельными компьютерами велся на основе общепринятых стандартов. Набор формальных правил о том, как и в каком виде следует передавать данные между различными устройствами и программами, называется *протоколом*.



(рис.4.37.2). Сервер

TCP - Transfer Control Protocol. Протокол управления передачей. Он служит для обеспечения и установление надежного соединения между двумя устройствами и надежную передачу данных. При этом протокол TCP контролирует оптимальный размер передаваемого пакета данных, осуществляя новую посылку при сбое передачи.

Этот протокол регламентирует, как следует разбивать длинное сообщение на пакеты, как должны быть устроены пакеты, как контролировать прибытие пакетов к месту назначения, что делать в случае ошибок передачи данных, и другие детали.

Собственно, Интернет представляет собой объединение десятков тысяч отдельных сетей, которые используют протокол TCP/IP и единое пространство IP-адресов. В остальном эти сети административно и финансово независимы.

Еще одно важное достоинство коммутации пакетов - это легкость объединения в единую сеть разных по скорости каналов связи. В связи с этим скоростные характеристики подключения к Интернету могут варьироваться в очень широких пределах, и разница будет заключаться лишь в скорости получения информации. Никаких других качественных отличий или ограничений нет.

Доменная система имен и указатели ресурсов.

У большинства компьютеров в Интернете есть собственное имя, а не только IP-адрес. Служба, которая обеспечивает перевод имен компьютеров в их IP-адреса, называется *Доменной Службой Имен (DNS)*.

Домен - это иерархическое имя сайта.

Домен можно сравнивать с адресом проживания. Чтобы понять, где живет человек, нужно знать *город, улицу, номер дома и квартиры*. Так и с сайтом. Чтобы зайти, нужно знать его **адрес**.

Домены были придуманы для того, чтобы придать лицо IP-адресу сайта. Адрес сайта состоит из *букв, цифр и дефисов*. Его длина может быть от 2 до 63 знаков.



(рис.4.37.2). Домены

.uz (*англ. Uzbekistan*) - национальный домен верхнего уровня Республики Узбекистан. Доступна регистрация имени домена второго или третьего уровня в доменных зонах *.uz, .for.uz, .gov.uz, .com.uz, .co.uz* физическими, юридическими лицами, резидентами, нерезидентами Узбекистана.

Национальный домен верхнего уровня - **uz** используется как национальный домен верхнего уровня в стандартах административно-территориального деления территории государства ISO 3166 (*ISO 3166-1, ISO 3166-2, ISO 3166-2:UZ*) в качестве кода **Alpha2**, образующего основу геокода административно-территориального деления Узбекистана.

Таблица доменов первого уровня доменной зоны .uz

Домен	Регламентированные пользователи
.uz	основной домен верхнего уровня.

Таблица доменов второго уровня доменной зоны .uz, условия использования, требования, регламентированные пользователи, категории доменов.

Домен	Регламентированные пользователи
.for.uz	Бессрочный домен для пользователей ID.UZ со статусом PassportID.
.gov.uz	Государственные учреждения Узбекистана.
.com.uz	Коммерческие организации.
.ac.uz	Институты, высшие учебные заведения, учреждений культуры, ведущие научные исследования.
.edu.uz	Учебные заведения.
.int.uz	Международные организации.
.net.uz	Поставщики интернет-услуг.
.org.uz	Некоммерческие организации.
.pp.uz	Физические лица.

Службы Интернета

Интернет - это средство обмена информацией *между людьми*, а не *между компьютерами*. Это один из инструментов, который помогает людям удовлетворять потребность в общении.

Разные *службы Интернета* реализуют разные функции обмена информацией, причем службы могут возникать и исчезать по мере необходимости. Некоторые из них становятся популярными и процветают, некоторые отвечают потребностям ограниченного круга пользователей, некоторые вытесняются более совершенными конкурирующими службами. Все зависит от потребностей людей в данном способе обмена информацией. Наиболее широко используются следующие службы Интернета:

Электронная почта (*e-mail*) - в соответствии с названием, это почта – служба для адресной доставки информации, - только электронная.

Новости (*news, USENET*) - это тематические конференции, доски объявлений или газеты, где каждый подписчик одновременно может быть автором. Эта служба, так же, как и веб-форумы или списки рассылки (*похожие по функциям, но отличающиеся техническими деталями*), предназначена для

обмена сообщениями в пределах группы людей, связанных общими интересами.

FTP («эф-тэ-пэ») - также одна из старейших служб, используется для копирования файлов с компьютера на компьютер.

Web-страница.

Публикация в Интернете, как и обычная публикация, состоит из упорядоченных страниц, на которых представлен *форматированный текст* и *иллюстрации* в виде картинок. Страницы в Интернете электронные – так называемые веб-страницы.

Web-страница – это созданный документ в формате *html*, который может включать в себя видео, фото, текст, графику, гиперссылки.

Основное содержимое веб-страницы называется *контентом*. А главной целью создания этих страниц является предоставление информации посредством всемирной паутины.

Кроме того, веб-страница может включать в себя так называемые *гипертекстовые ссылки* (*гиперссылки*), соединяющие контекст, в котором они находятся, с другим контекстом в рамках того же или другого текста, находящегося на произвольном сервере сети.

Свойства сети Интернет

Internet вобрала в себя все важные *свойства информационных систем*. Отличительной чертой Internet является *интерактивность*, т.е. возможность немедленно отреагировать на обращение.

Таким образом, Internet представляет собой уникальную *коммуникационную систему*, которая может использоваться не только для какой-то одной цели. Ее привлекательность как раз и состоит в многообразии возможностей и в том новом, что она несет для различных видов деятельности.



Контрольные вопросы и задания

1. Какова история зарождения Интернет?
2. Опишите преимущества локально-вычислительных сетей.
3. Что такое глобальная сеть? Какая ее важнейшая особенность?
4. За что отвечают протоколы TCP/IP?
5. Основные службы Интернет?
6. Какова система адресации в Интернет?
7. Из чего состоит IP-адрес?
8. Какие типы электронной почты используются?
9. В каком году появилось первая сеть, как она называлась и где?
10. Теоретические основы организации сети интернет

38-§. ТЕХНОЛОГИЯ СОЗДАНИЯ WEB-СТРАНИЦ.

Ключевые слова: *тег, атрибут, текст, звук, анимация, видео, html, www, web, тело, голова, гипертекст.*

WWW (*World Wide Web*) - глобальная компьютерная сеть на сегодняшний день содержит *миллионы сайтов*, на которых размещена всевозможная информация. Люди получают доступ к этой информации посредством использования технологии Internet. Для навигации в WWW используются специальные программы - **Web-браузеры**, которые существенно облегчают путешествие по бескрайним просторам WWW. Вся информация в Web-браузере отображается в виде Web-страниц, которые являются основным элементом байтов WWW.

Что такое Web-страница?

Web-страница (*англ. Web-page*) – документ или информационный ресурс Всемирной паутины, доступ к которому осуществляется с помощью веб-браузера.

Типичная Web-страница представляет собой текстовый файл в формате HTML (*.html*). Например, *index.html*.



(рис.4.38.1). Виды веб-страниц

Web-страницы, поддерживая *технология мультимедиа*, объединяют в себе различные виды информации: *текст, графику, звук, анимацию и видео*. От того, насколько качественно и красиво сделана та или иная Web-страница, зависит во многом ее успех в Сети.

Мультимедиа - *включает в себя не только текст, но и двух - и трехмерную графику, видео и звук.*

Информация в WWW представляется в виде документов, любой из которых может содержать как внутренние перекрестные ссылки, так и ссылку на другие документы, которые сохраняются на том же самом или на любом другом сервере.

Гипертекст - *это система страниц с текстом, которые связаны ссылками.*

Web-страница может содержать стилизованный и форматированный текст, графику и гиперсвязи с разными ресурсами Internet. Чтобы реализовать все эти возможности, был разработан специальный язык, названная **HyperText Markup Language (HTML)**, то есть, язык разметки гипертекста.

HTML – *это язык гипертекстовой разметки, который получил очень широкое распространение в сети Интернет.*

Документ, написанный на HTML, представляет собой текстовый файл, который содержит собственно текст, несущий информацию читателю, и флаги

разметки. Последние представляют собой определенные последовательности символов, которые являются инструкциями для программы просмотра; в соответствии с этими инструкциями программа располагает текст на экране, включает в него рисунки, которые сохраняются в отдельных графических файлах, и формирует гиперсвязи с другими документами или ресурсами Internet. Таким образом, файл на языке HTML приобретает вид WWW-документа только тогда, когда он интерпретируется программой просмотра.

Web-страницы могут существовать в любом формате, но в качестве стандарта принят Hyper Text Markup Language - язык разметки гипертекстов, предназначенный для создания форматированного текста, насыщенного изображениями, звуком, анимацией, видеоклипами и гипертекстовыми ссылками на другие документы, разбросанные как по всему Web-пространству, так и находящиеся на этом, же сервере или являющиеся составной частью этого Веб-проекта.

Можно работать на Web без знания **языка HTML**, поскольку тексты HTML могут создаваться разными специальными редакторами и конвертерами. Но писать непосредственно на HTML нетрудно. Возможно, это даже легче, чем изучать HTML-редактор или конвертер, которые часто ограничены в своих возможностях, содержат ошибки или проводят плохой HTML код, который не работает на разных платформах.

Язык HTML существует в нескольких вариантах и продолжает развиваться, но конструкции HTML скорее всего будут использоваться и в дальнейшем. Изучая HTML и познавая его глубже, создавая документ в начале изучения HTML и расширяя его насколько возможно, пользователь имеет возможность создавать Web-страницы, которые могут быть просмотрены многими браузерами Web, как сейчас, так и в будущем.

Браузер – это программа, которая позволяет просматривать содержимое веб-страниц.



(рис.4.38.2). Виды браузеров

Работа по HTML - это способ усвоить особенности создания документов в стандартизированном языке, используя расширения, только если это действительно необходимо.

HTML был ратифицирован World Wide Web Consortium. Он поддерживается несколькими широко распространенными браузерами, и, возможно, станет основанием почти всего программного обеспечения, которое имеет отношение к **Web**.

Структура HTML документа

Поскольку HTML-документы записываются в **ASCII-формате**, то для ее создания может использоваться любой текстовый редактор.

Обычно HTML-документ - это файл с расширением **.html** или **.htm**, в котором текст размечен HTML-тегами (*англ. tag - специальные встроенные указания*). Средствами HTML задаются синтаксис и размещение тегов, в соответствии с которыми браузер отображает содержимое Веб-документа. Текст самих тегов Веб-браузером не отображается.

Тег - это метка, размечающая и каталогизирующая информацию для облегчения процесса поиска.

Все теги начинаются символом “<” и заканчиваются символом “>”. Обычно имеется пара тегов - **стартовый** (*открывающий*) и **завершающий** (*закрывающий*), между которыми помещается размечаемая информация: например, `<p>Информация</p>`. Здесь стартовым тегом является тег `<p>`, а завершающим - `</p>`.

Завершающий тег отличается от стартового лишь тем, что у него перед текстом в скобках $\langle \rangle$ стоит символ «/» (*слэш*). Браузер, читающий HTML-документ, отображает его в окне, используя структуру HTML-тегов. В каждом HTML-документе должны присутствовать три главных части:

1. Объявление HTML;
2. Заголовочная часть;
3. Тело документа.



(рис.4.38.3). Структура Web-страницы

Объявление HTML

$\langle \text{HTML} \rangle$ и $\langle / \text{HTML} \rangle$. Пара этих тегов сообщает программе просмотра (*браузеру*) что между ними заключен документ в формате HTML, причем первым тегом в документе должен быть тег $\langle \text{HTML} \rangle$ (*в самом начале документа*), а последним - $\langle / \text{HTML} \rangle$ (*в самом конце документа*).

$\langle \text{HTML} \rangle$

$\langle / \text{HTML} \rangle$

Заголовочная часть.

$\langle \text{HEAD} \rangle$ и $\langle / \text{HEAD} \rangle$. Между этими тегами располагается информация о документе (*название, ключевые слова для поиска, описание и т.д.*). Однако наиболее важным является название документа, которое мы видим в верхней строке окна браузера и в списках "Избранное (*BookMark*)". Специальные программы-спайдеры поисковых систем используют название документа для построения своих баз данных.

Для того чтобы дать название своему HTML-документу текст помещается между тегами <TITLE> и </TITLE>.

```
<HTML>
<HEAD>
<TITLE>Моя первая страница</TITLE>
</HEAD>
</HTML>
```

Тело документа.

Третьей главной частью документа является его тело. Оно следует сразу за заголовком и находится между тегами <BODY> и </BODY>. Первый из них должен стоять сразу после тега </HEAD>, а второй - перед тегом </HTML>. Тело HTML-документа - это место, куда можно помещать информацию, отформатированную средствами HTML.

```
<HTML>
<HEAD>
<TITLE> Моя первая страница</TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Теперь можно написать HTML-код странички:

```
<HTML>
<HEAD>
<TITLE>Моя первая страница</TITLE>
</HEAD>
<BODY>
Страница для отображения!
</BODY>
</HTML>
```



Контрольные вопросы и задания

1. Интернет (WWW)?
2. Язык HTML
3. Структура *HTML* документа
4. Объявление HTML
5. Заголовочная часть.
6. Тело документа.

ПРАКТИЧЕСКИЕ РАБОТЫ

39-§. СОЗДАНИЕ ВЕБ СТРАНИЦЫ В HTML.

Цель работы: *Использовать HTML-теги при разработке web-страниц.*

Краткая теоретическая информация:

HTML (*Hyper Text Markup Language*) - язык разметки гипертекста.

Для создания веб-страниц используются языки **HTML** и **CSS** (*Cascading Style Sheets*). **HTML** отвечает за *структуру* и *содержание* страницы, **CSS** – за *внешний вид*. **Браузер** объединяет HTML- и CSS-код и формирует внешний вид страницы.

CSS (*Cascading Style Sheets*) - язык таблиц стилей, который позволяет прикреплять стиль (например, шрифты и цвет) к структурированным документам (например, документам HTML и приложениям XML).

CSS описывает, как **HTML**-элементы должны отображаться на экране, бумаге или на других носителях. **CSS** экономит много работы. Он может контролировать расположение нескольких *веб-страниц* все сразу.

CSS можно добавлять к элементам **HTML** тремя способами:

- **Встроенный** - с помощью атрибута **Style** в элементах HTML;
- **Internal** (*внутренний*) - с помощью **<style>** элемента в **<head>** разделе;
- **Внешний** - с помощью внешнего **CSS**-файла.

Тег - это специальное слово, заключенное в *угловые скобки*, Тег является основной составляющей HTML, этим словом код начинается и им же заканчивается, а внутри тега заключается все содержимое, которое отображается на *интернет странице*. Почти все теги являются парными, и выглядят так: **<Открывающий тег>** содержимое **</Закрывающий тег>**.

Тот тег, где используется слеш («/» *косая черта*) всегда будет закрывающим тегом.

Небольшой список *парных тегов*:

<h1> </h1> - заголовок,

<p> </p> - абзац,

 - жирный шрифт.

Одиночные теги.

Одиночные теги - которые называют *метками*, их не закрывают, они выполняют конкретную функцию. Например,

 - *эта метка переносит текст на следующую строку*,

 - изображение,

<input> - элемент формы.

Иногда в HTML тегах устанавливают атрибуты со значениями:

<имя тега атрибут="значение"> </имя тега>

<имя тега атрибут="значение">

Виды (категории) HTML-тегов

Теги делятся на несколько видов. Их можно разделить на десять категорий, в других учебниках количество категорий может различаться:

1. Парные теги,
2. Одиночные теги,
3. Блочные теги, (*html, head u body*),
4. Строчные теги,
5. Новые теги HTML5, документа,
6. Устаревшие теги,
7. Нестандартные теги,
8. Теги верхнего уровня
9. Теги головы HTML-документа,
10. Семантические теги.

Один и тот же тег, может одновременно принадлежать нескольким категориям, например, тег <small> </small> относится к следующим видам: *парный, строчный, семантический*.

Блочные теги.

Блочные теги - это теги в начале и в конце которых автоматически ставится перенос строки, также они занимают всю ширину *родительского элемента*.

`<h1> </h1>` - заголовок первого уровня,

`<p> </p>` - абзац,

`<table> </table>` - таблица.

Строчные теги

Строчные теги - это теги в начале и в конце которых не ставится перенос строки.

` ` - жирный шрифт,

`<i> </i>` - наклонный шрифт,

`<input>` - элемент формы.

Новые теги HTML5

Новые теги HTML5 - это теги, которые появились в спецификации языка HTML5.

`<article> </article>` - основной контент страницы,

`<canvas> </canvas>` - замена флеш-технологии,

`<video>` - видеофайл.

Устаревшие теги.

Устаревшие теги - это теги использовать которые с точки зрения спецификации языка **HTML5**, не рекомендуется. Однако при создании сайтов, вы можете использовать устаревшие теги, браузеры всё равно их поймут.

`<big> </big>` - увеличивает размер шрифта на единицу,

` ` - форматирует текст,

`<isindex>` - поисковая строка.

Нестандартные теги.

Нестандартные теги - это теги, которые не поддерживаются ни одной спецификацией HTML, но некоторые браузеры их понимают.

`<bgsound>` - проигрывает музыку при открытии страницы,

<blink> </blink> - мигающий текст,
<marquee> <marquee> - бегущая строка.

Теги верхнего уровня

Теги верхнего уровня - это теги, отвечающие за создание HTML-документа, их всего три:

<html> </html> - начало и конец HTML-документа,
<head> </head> - голова HTML-документа,
<body> </body> - тело HTML-документа.

Теги головы HTML-документа

В голове HTML-документа, между тегами <head> </head>, обычно находятся теги отвечающие за описание страницы, за подключение внешних файлов и др., они не видны пользователю. Единственный тег головы, который видит пользователь, это тег `title`.

<link> - подключает внешние файлы `.css`, `.xml`, `.ico`,
<meta> </meta> - мета-теги,
<title> </title> - название страницы,
<style> </style> - внедряет **CSS**-код в страницу,
<script> </script> - загружает внешний **JavaScript**-файл, также между этими тегами может размещаться JavaScript-код.


Семантические теги.

Семантические теги - это теги которые предназначены для того, чтобы компьютерные программы (*поисковые системы, сборщики информации, речевые браузеры и т.д.*), понимали смысл заложенной в них информации.

<article> </article> - обрамляют основной контент страницы,
<blockquote> - </blockquote> - обрамляют цитаты, изъятые из внешних источников,
<small> </small> - обрамляет текст, написанный мелким шрифтом в юридических документах. Также следует отметить, что *нынешняя политика HTML* стремится сделать почти все теги семантическими.

Создание макета главной страницы.

Открыть текстовый редактор. В Windows запустить **Блокнот** или **Notepad++**.

1. Открыть меню «**Пуск**» , в строке поиска ввести *блокнот* или *notepad++*, а затем щелкнуть по «*Блокнот*» или «*Notepad++*» вверху меню «Пуск».

2. Ввести `<!DOCTYPE html>` и нажать **Enter**. Это будет сообщением веб-браузеру, что это документ **HTML**.

3. Ввести `<html>` и нажать **Enter**. Это открывающий тег HTML-кода.

4. Ввести `<head>` и нажать **Enter**. Этот тег содержит описание веб-страницы и *заголовочные элементы*. Контент этого тега на странице, как правило, не отображается. Контентом являются описание страницы, метаданные, таблицы стилей **CSS** и другие языки сценариев.

5. Ввести `<title>`. Этот тег содержит имя страницы.

6. Ввести имя страницы. Ввести текст, который будет отображаться на вкладке страницы (*например, Моя первая веб-страница*).

7. Ввести `</title>` и нажмите **Enter**. Этот тег закрывает тег имени страницы.

8. Ввести `</head>` и нажмите **Enter**. Этот тег закрывает тег описания страницы. На данном этапе HTML-код должен выглядеть примерно так:

```
<!DOCTYPE html>
<html>
<head>
<title> Моя первая веб-страница </title>
</head>
```

9. Ввести тег `<body>` под закрывающим тегом «`head`». Этот тег содержит *весь контент HTML-документа*. Контент этого тега отображается на веб-странице.

10. Ввести `<h1>`. Этот тег содержит заголовок страницы. Заголовок - это текст большого размера, который, как правило, расположен вверху страницы.

11. Ввести заголовок страницы. Это может быть название страницы или приветствие.

12. Ввести `</h1>` после текста заголовка и нажмите **Enter**. Этот тег закрывает тег заголовка.

Добавить *дополнительные заголовки по мере необходимости*. Можно создать до шести заголовков; это делается с помощью тегов `<h1>` `</h1>` - `<h6>` `</h6>`. Заголовки будут иметь разные размеры.

13. Ввести `<p>`. Этот тег содержит *абзац текста*. Такой тег отобразит текст нормального размера.

14. Ввести текст, например, *ТГПУ имени Низами. Кафедра «Информационные технологии»*.

15. Ввести `</p>` после текста и нажмите **Enter**. Этот тег закрывает тег абзаца текста. Ниже приведен пример абзаца в HTML-документе:

```
<p> Это первый абзац. </p>
```

- Чтобы создать несколько абзацев под одним заголовком, добавьте подряд несколько строк.

- Изменить цвет текста. В начале текста ввести тег ``, а в конце тег ``. Вместо слова "цвет" подставить нужный цвет (*на английском языке*), причем кавычки не удаляйте. Эти теги позволят изменить цвет любого текста (*например, заголовка*). К примеру,

```
<p> <font color = "Lime"> текст </font> </p>
```

- Также шрифт можно сделать полужирным, наклонным и другим. Ниже приведены примеры форматирования текста с помощью *HTML-тегов*:

```
<b> Полужирный текст </b>
```

```
<i> Наклонный текст </i>
```

```
<u> Подчеркнутый текст </u>
```

_{Подстрочный текст}

^{Надстрочный текст}

16. Ввести </body>, чтобы закрыть тег «body».

17. Ввести </html>, чтобы закрыть **HTML-код**. Введите этот тег под закрывающим тегом «body». Так вы сообщите веб-браузеру, что после этого тега HTML-кода нет.

18. Нажать **Файл**. Эта опция находится в строке меню вверху экрана.

19. Щелкнуть по *сохранить как*. Эта опция находится в меню «**Файл**». Также можно нажать **Ctrl+S**.

20. Ввести имя HTML-документа. Ввести его в строке «**Имя файла**».

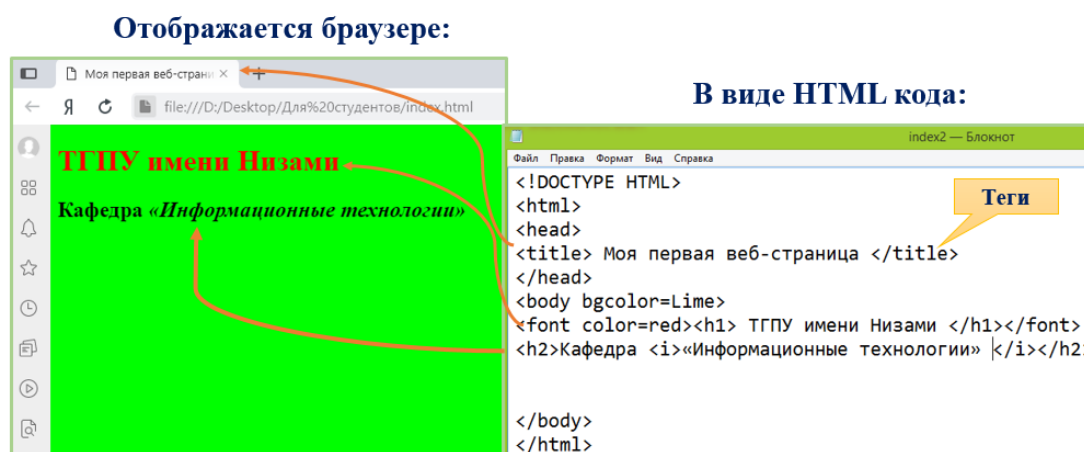
21. Изменить формат файла с **ТХТ** на **HTML**. Для этого: щелкнуть по меню «**Тип файла**», выбрать «**Все файлы**», а затем ввести **.html** в конце имени файла, например, *index.html*.

22. Щелкнуть по **Сохранить**. Эта опция находится внизу окна. Будет создан *HTML-файл*.

23. Обычно HTML-файлы открываются в *веб-браузере по умолчанию*.

Открыть HTML-файл в браузере, в котором можно просмотреть *созданную веб-страницу*.

Весь HTML-код должен быть *примерно таким*:



(рис.4.39.1). Результат работы

Дополнительные элементы.

Добавление картинки на страницу. Для этого:

Ввести `` после URL-адреса изображения, чтобы закрыть тег картинки. К примеру, если URL-адрес изображения `http://www.myimage.com/ocean.jpg`, ввести следующий код:

```

```

Добавление ссылки на другую веб-страницу. Для этого:

- Ввести `<a href=` Этот тег содержит ссылку на другую страницу.
- Скопировать и вставить URL-адрес после знака равенства (=) в кавычках.
- Ввести `>` после URL-адреса, чтобы закрыть адрес.
- Ввести текст ссылки после символа `<>`.
- Ввести `` после текста ссылки, чтобы закрыть тег ссылки. Ниже приведен пример ссылки на Яндекс.

```
<a href="https://www.ya.ru">Яндекс</a>.
```

Добавление разрыва строки. Для этого ввести `
`. Будет вставлен перенос строки. Этим тегом можно пользоваться, чтобы отделить друг от друга различные разделы страницы.



Контрольные вопросы и задания

1. Что такое HTML?
2. Что такое тэг?
3. Какие виды тэгов бывают?
4. Для чего нужны параметры?
5. Каков синтаксис использования параметров?
6. Какова структура HTML-документа?
7. Какие тэги содержатся в шапке документа?
8. Какие тэги содержатся в теле документа?
9. Какие тэги используются для форматирования текста?

40-§. СОЗДАНИЕ WEB-СТРАНИЦ С ПОМОЩЬЮ DREAMWEAVER.

Цель работы: *Дать понятие о программе Macromedia Dreamweaver.*

Краткая теоретическая информация:

Macromedia Dreamweaver (в переводе – ткач мечты)- это HTML-редактор, созданный известной во всем мире компанией Macromedia.

Богатый инструментарий, удобный интерфейс, открытость приложения для всевозможных настроек и другие особенности делают Dreamweaver одним из самых *популярных HTML-редакторов в мире.*



(рис.4.40.1). Интерфейс программы Dreamweaver

Возможности **Macromedia Dreamweaver 8** значительно шире, чем возможности обычных редакторов для создания веб-страниц. Однако вместе с этим программа благодаря легко настраиваемому и простому пользовательскому интерфейсу подходит как для профессионалов, так и начинающих создателей *веб-проектов* любой сложности.

Основной упор в данном продукте сделан на визуализацию разработки веб-страницы. Работа проходит в режиме *WYSIWYG (What You See Is What You Get – что видишь, то и получаешь)*, то есть Dreamweaver объединяет в себе средство для визуального расположения элементов, управления их свойствами на уровне языка **HTML** и средства поддержки *Dynamic HTML*.

Macromedia Dreamweaver обеспечивает:

- генерацию свободного от *ошибок HTML-кода*;
- *динамическое отображение* создаваемой страницы в выбранном браузере;
- сохранение расположения для ранее созданных документов и документов, *созданных другими средствами*;
- средства наглядного дизайна *таблиц и фреймов*;
- генерацию *Dynamic HTML* для различных браузеров;
- расширяемую коллекцию скриптовых *программ JavaScript*;
- поддержку анимации, схожую с инструментами Macromedia Director.

Кроме этого:

- Macromedia Dreamweaver 8 - это и разработчик дизайна, и редактор кода в одном инструменте;
- Dreamweaver 8 позволяет с легкостью использовать новые технологии дизайна и кодирования;
- с помощью Dreamweaver 8 и Flash Video поместить видео на сайт не составит никакого труда;
- мощная CSS-панель предоставляет возможность произвести изменения, не набирая код вручную;
- Dreamweaver взаимодействует с XML.

Macromedia Dreamweaver позволяет создавать страницы на основе фреймов. Страница делится на фреймы с помощью команды **Modify - Frameset**, после чего можно задать цвет фона каждого фрейма или выполнить какие-либо другие действия. Расположив фреймы, можно заняться дизайном каждого фрейма в качестве отдельной страницы или продолжить работать с ними в виде фреймов. При этом при необходимости можно переключиться в редактор HTML-кода и вносить изменения вручную.

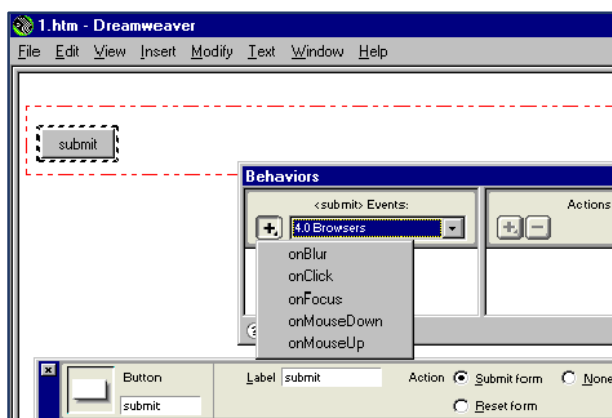
Чтобы вставить какие-либо элементы, нужно использовать меню **Insert** и различные диалоговые панели, появляющиеся при выборе отдельных команд. Помимо традиционных меню есть множество отдельных

инструментальных панелей: для вставки отдельных элементов, для установки атрибутов, для управления фреймами и т.п.

Спецификация каскадных таблиц стилей в программе появилась довольно давно. Однако этот механизм стал по-настоящему актуальным только с внедрением в современные браузеры поддержки технологии **Dynamic HTML**. Пакет Macromedia Dreamweaver обеспечивает удобную работу с каскадными таблицами стилей. В панели определения стилей есть возможность создать собственные стили или импортировать существующие.

Поддержка JavaScript

Macromedia Dreamweaver 8 обладает расширенными возможностями написания скриптовых программ и их связи с отдельными элементами создаваемого документа. Например, мы помещаем в документ форму, а в нее - обычную кнопку. В панели **Behaviors** мы можем сразу же получить список событий, поддерживаемых данным элементом, и даже выбрать ряд предопределенных скриптовых программ, например, *вывести сообщение в диалоговой панели* или *в статусной строке*.



(рис.4.40.2). Панель Behaviors

Macromedia Dreamweaver 8 поддерживает создание *скриптовых программ* для различных браузеров, что означает возможность выбора платформы из списка представленных браузеров.

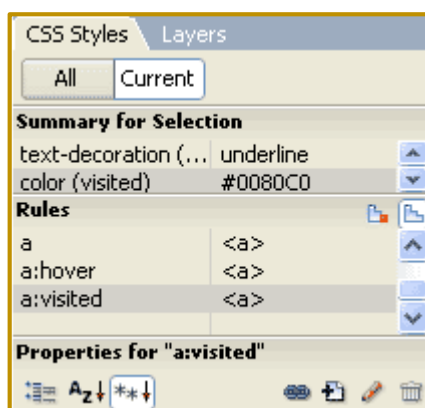
Анимация.

Механизм поддержки анимации в данном продукте называется **Timeline** (в переводе - линия времени). Его суть заключается в том, что после выбора

объекта задается сценарий его поведения. Для этого используется временная шкала (*timeline*), разделенная на кадры. К каждому кадру можно присоединить какое-либо событие.

Новые возможности.

Прежде всего, Macromedia Dreamweaver 8 дает возможность визуальной работы с данными XML. Кроме того, программа обладает улучшенной технологией **WebDAV**, возможностью фоновой загрузки файлов во время работы с документом, новыми шаблонами сайтов, а также включает новые инструменты для облегчения работы в создании страниц в режиме *дизайна* или *кода*.



(рис.4.40.3). Панель CSS

Модифицированная панель CSS.

- Модифицированная панель CSS

Новая (*модифицированная*) панель CSS содержит сгруппированные в одном месте инструменты для изучения, работы и применения стилей CSS к страницам в режиме визуального редактирования. С новым интерфейсом легче просматривать каскадные таблицы, примененные к отдельным элементам.

- Усовершенствованное представление CSS

Усовершенствованное представление CSS позволяет просматривать результат применения CSS в разных браузерах с учетом их особенностей. Dreamweaver полностью поддерживает продвинутую технику CSS, например, *псевдоэлементы*, *элементы формы* и т.д.

- Опция **Zoom**

Опция Zoom служит для более точного контроля над дизайном и позволяет приближать или удалять страницу создаваемого документа. Границы манипуляции - от 6 до 6400%.

- **Специальная вставка**

С новой опцией вставки в Dreamweaver 8 появилась возможность сохранить все форматирование, созданное в **Microsoft Word**, или просто вставить текст.

- **Визуализация слоев CSS** (*в режиме дизайна границ, фона слоев CSS*). Применение визуальной помощи в работе со слоями поможет разобраться при работе со сложными структурами документов.

- **Улучшенная синхронизация и сравнение файлов**

Эта функция позволит быстро сравнить файлы для определения, что было изменено в них в процессе работы. Есть возможность сравнить два локальных файла, файл на локальном и удаленном сайтах и два файла на удаленном компьютере.

- **Обновлена помощь от O'REILLY**

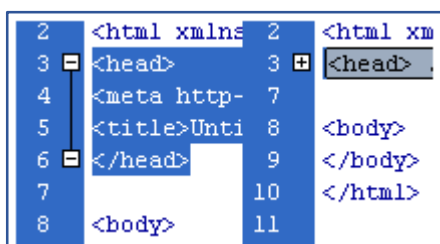
Добавлены новые материалы по XML, XSLT и XPath, обновлены материалы по ASP.

- **Панель Style Rendering**

С помощью этой панели при создании документа можно посмотреть, как пользователь будет видеть страницу с поддержкой стилевых таблиц CSS и без нее, а также в разных представлениях.

- **Поддержка новых технологий**

Поддержка ColdFusion MX 7, PHP 5, Flash Video.



(рис.4.40.4). Поддержка ColdFusion MX 7, PHP 5, Flash Video

- **Сворачивание кода.**

Есть возможность свернуть часть кода, что сильно упрощает работу и делает код более простым для восприятия.

- **Отступы**

В Dreamweaver 8 есть кнопка, не менее опасная в плане замусоривания страницы, чем инструменты свойств текста. Эта кнопка - увеличение отступа. Использовать такой способ управления отступом не рекомендуется, поскольку Dreamweaver применяет не предназначенные для этого теги. Если необходимо создать несколько абзацев с большим отступом от левого края, пользуйтесь для этого классами CSS.

- **Стили в Dreamweaver**

Встроенный редактор таблиц стилей позволяет создавать классы CSS (*Make Custom Style Class*) или назначать стилевое оформление отдельным элементам HTML (*Redefine HTML Tag*).



Контрольные вопросы и задания

1. Какие типы документов можно *открывать* с помощью Macromedia Dreamweaver 8?
2. Какие типы кода можно *создавать* в программе Macromedia Dreamweaver 8?
3. Что представляет собой *программное обеспечение* Macromedia Dreamweaver 8?
4. Чем отличается Macromedia Dreamweaver 8 от других средств *разработки веб-сайтов*?
5. Каковы *минимальные требования* к системе для работы Macromedia Dreamweaver 8?
6. Совместима ли программа Macromedia Dreamweaver 8 с операционной системой **Windows 8.1**?

ЛАБОРАТОРНЫЕ РАБОТЫ

41-§. ПОИСК И СОХРАНЕНИЕ ИНФОРМАЦИИ В СЕТИ ИНТЕРНЕТ

Цель работы: Создать аннотированный список Интернет-ресурсов (*библиотек*) по информационным технологиям. Аннотированный список должен содержать не менее *10 Интернет-ресурсов*.

Краткая теоретическая информация:

Поисковые системы

Сеть Интернет представляет собой базу данных, в которой хранится огромное количество самой разнообразной информации. Очень важно суметь найти во всем этом многообразии именно ту информацию, которая окажется действительно полезной и поможет решить поставленную задачу.

Искать информацию можно с помощью поисковой системы. В Интернете имеется достаточное количество подобных систем:

- yandex.ru;
- google.ru;
- nigma.ru;
- yahoo.ru;
- rambler.ru;
- aport.ru и др.

В каждой поисковой системе имеются свои механизмы поиска. Некоторые системы имеют дополнительные средства фильтрации уже отобранных данных, например, nigma.ru.

Поисковая система – *web-сайт, предоставляющий возможность поиска информации в Интернете.*

Большинство поисковых систем ищут информацию на сайтах Всемирной паутины, но существует также системы, способные искать файлы на ftp-серверах, а также информацию в группах новостей **Usenet**.

По принципу действия поисковые системы делятся на два типа:

- поисковые каталоги;
- поисковые индексы.

Поисковые каталоги

Поисковые каталоги служат для тематического поиска. Информация на этих серверах структурирована по темам и подтемам. Имея намерение осветить какую-то узкую тему, нетрудно найти список web-страниц, ей посвященных.

Каталог ресурсов в Интернете или каталог Интернет-ресурсов, или просто интернет-каталог – структурированный набор ссылок на сайты с кратким их описанием.

Каталог, в котором ссылки на сайты внутри категорий сортируются по популярности сайтой называется рейтинг.

Поисковые индексы

Поисковые индексы работают как алфавитные указатели. Пользователь задает слово или группу слов, характеризующих его область поиска, - и получает список ссылок на web-страницы, содержащие указанные термины.

Первой поисковой системой для Всемирной паутины был «Wandex», уже не существующий индекс, разработанный **Мэтью Грэйем** из Массачусетского технологического института в 1993.

Как работает поисковой индекс?

Поисковые индексы автоматически, при помощи специальных программ (*веб-пауков*), сканируют страницы Интернета и индексируют их, то есть заносят в свою огромную базу данных.

Поисковый робот (*веб-паук*) – программа, являющаяся составной частью поисковой системы и предназначенная для обхода страниц Интернета с целью занесения информации о них (*ключевые слова*) в базу поисковика.

По своей сути паук больше всего напоминает обычный браузер. Он сканирует содержимое страницы, забрасывает его на сервер поисковой системы, которой принадлежит и отправляется по ссылкам на следующие страницы.

В ответ на запрос, где найти нужную информацию, поисковый сервер возвращает список гиперссылок, ведущих к web-страницам, на которых нужная информация имеется или упоминается. Обширность списка может быть любой в зависимости от содержания запроса.

Работа с поисковыми серверами.

При входе на главную страницу поискового сервера достаточно набрать в поле ввода свой запрос в виде набора ключевых слов и нажать кнопку начала поиска. Запросы могут содержать любые слова, причем не обязательно заботиться о падежах и склонениях.

Современные поисковые сервера достаточно хорошо понимают естественный язык, однако на многих из них сохранены возможности расширенного или специального поиска, позволяющие искать слова по маске, объединять слова запроса логическими операциями “И”, “ИЛИ” и т.д.

После завершения поиска в базе данных сервер выводит на экран первую порцию из 10 или более документов, содержащих ключевые слова. Кроме ссылки, обычно приведено несколько строк текста описания документа или просто его начало. Открывая ссылки в новом или этом же окне браузера, можно переходить к выбранным документам, а строка ссылок внизу страницы позволяет перейти к следующей порции документов.

По отсутствию одной ссылки видно, что сейчас открыта вторая порция найденных документов.

Различные серверы сортируют найденные документы по-разному - по дате создания, по посещаемости документа, по наличию в документе всех или части слов запроса (*релевантности*), некоторые серверы позволяют сузить область поиска, выбрав на главной странице категорию искомого документа.

В Интернет немало страниц для метапоиска, позволяющих обратиться сразу к нескольким популярным поисковым серверам с одним и тем же запросом.

Выполнение работы

1. Познакомиться с содержанием материалов для самостоятельного изучения по теме «Поиск и сохранение информации в сети интернет»

2. В текстовом файле создать таблицу, отражающую результаты поиска:

№ п/п	Адрес Интернет-ресурса	Автор ресурса	Скриншот главной страницы	Аннотация
1.				
2.				
3.				

Примечание

1. Адрес Интернет-ресурса. Представляет собой *url-адрес* ресурса. Например, *http://computerhistory.narod.ru/*

2. Автор ресурса. Кто создал ресурс или кому этот ресурс принадлежит. Например, автор идеи и web-дизайн: Юсупов С., *e-mail: ComputerHistory@yandex.ru*

3. Скриншот главной страницы.

4. Аннотация. Отражается, чем полезен ресурс, его содержание, как можно использовать в учебной и профессиональной деятельности.

5. Сохранить созданный текстовый файл в формате *.doc* и выложить его в блог.

6. Результаты выполнения задания необходимо *показать преподавателю!*

Задание:

Запустите браузер и в строке адреса укажите адрес поисковой системы google.ru. С помощью каталога данной поисковой системы найдите следующую информацию (*уточните вариант у преподавателя*):

1. Примерная стоимость мультимедийного компьютера;
2. Долгосрочный прогноз погоды в нашем регионе (*не менее чем на 10 дней*);
3. Текст конституции Республики Узбекистан;
4. Гороскоп своего знака зодиака на текущий день.
5. Об информации, информационных технологиях и о защите информации.

По результатам поиска составьте письменный отчет в Word: представьте в документе найденный, скопированный и отформатированный материал. Вначале выполненного задания укажите текст задания. Сохраните отчет в вашу папку.



Контрольные вопросы и задания

1. Что такое поисковая система?
2. Какие типы поисковых систем вы знаете?
3. Что такое поисковый каталог? Как осуществляется поиск информации в таком каталоге?
4. Дайте определение поисковому роботу
5. Назовите способы улучшения результатов поиска
6. Какова технология поиска, по ключевым словам.
7. Какие правила формирования поисковых запросов в системе *Yandex* вы знаете?

42-§. HTML ТЕГИ ДЛЯ ОФОРМЛЕНИЯ ТЕКСТА. ТЕГИ ДЛЯ РАБОТЫ С ТАБЛИЦАМИ И С HTML ИЗОБРАЖЕНИЯМИ.

Цель работы: Изучить основы организации таблиц в html. Знакомство с возможностями размещения изображений в html

Краткая теоретическая информация:

Для разметки текста в html используется набор тегов, позволяющий разделять текст на логические части, а также оптимизировать содержание для более удобного понимания поисковыми работами, что приводит к улучшению ранжирования страницы текста в поисковой выдаче по ключевым запросам.

Теги форматирования шрифта

Название	Комментарии
<code>...</code>	Выделение текста жирным шрифтом
<code><i>...</i></code>	Выделение текста курсивом
<code><u>...</u></code>	Подчеркивание текста
<code><s>...</s></code>	Зачеркивание текста
<code><sup>...</sup></code>	Верхний индекс
<code><sub>...</sub></code>	Нижний индекс
<code>...</code>	Размер шрифта (<i>вместо * от 1 до 7</i>)
<code>...</code>	Гарнитура шрифта (<i>вместо "" – название шрифта</i>)
<code>...</code>	Цвет текста (<i>вместо "" название или код цвета</i>)
<code><body bgcolor=></code>	Цвет фона страницы

(табл.4.42.1). Теги форматирования шрифта

Создания таблицы начинается с тега `<table>` (*от англ. «таблица»*). Тег `<tr>` служит для создания строки. В строке располагаются ячейки – тег `<td>`. Завершается таблица закрывающим тегом `</table>`

Теги вставки таблиц

Название	Комментарии
<code><Table>...</Table></code>	Обрамляет всю таблицу
<code>Border=1, 2, 3, ...</code>	Задаёт внешнюю границу
<code>Cellspacing=</code>	Расстояние между второй линией, является внутренней границей
<code>Cellpadding=</code>	Задаёт расстояние от содержимого ячейки до рамки слева
<code>Bgcolor=""</code>	Цвет фона таблицы
<code>BorderColor=</code>	Цвет внешней границы
<code>Width=</code>	Ширина таблицы
<code>Align=</code>	Выравнивание (<i>left, right, center</i>)
<code>background</code>	Файл (<i>фон таблицы</i>)

(табл.4.42.2). Теги вставки таблиц

```

<table>
<tr> <td> </td> <td> </td> </tr>
<tr> <td> </td> <td> </td> </tr>
<tr> <td> </td> <td> </td> </tr>
</table>

```

(рис.4.42.1). Структура таблицы

Атрибуты тега <tr>-строки

align	left, center, right	выравнивание по горизонтали
valign	top, middle, bottom, baseline	выравнивание по вертикали
bgcolor	цвет	задний фон
bordercolor	цвет	цвет границы

(табл.4.42.3). Атрибуты тега <tr>-строки

Атрибуты тега <td> или <th>-ячейки

align	left, center, right	выравнивание по горизонтали
valign	top, middle, bottom, baseline	выравнивание по вертикали
width	число или процент	ширина ячейки
bgcolor	цвет	цвет фона
background	файл	файл фона
bordercolor	цвет	цвет границы
nowrap		заставляет текст внутри ячейки отображаться без переносов, одной сплошной строкой

(табл.4.42.4). Атрибуты тега <td> или <th>-ячейки

Тег <caption> заголовка таблицы может иметь атрибут, определяющий расположение заголовка - align - со следующими значениями:

top - заголовок над таблицей,

bottom - заголовок под таблицей,

left - заголовок сверху и выровнен влево,

right - заголовок сверху и выровнен вправо. К сожалению не все значения «работают» во всех браузерах.

Объединение ячеек в таблице

Все происходит при помощи двух атрибутов тега <td>: **colspan** - объединение ячеек по горизонтали, **rowspan** - объединение ячеек по вертикали.

Группировка ячеек: Colgroup

Элемент `colgroup` позволяет создавать группы колонок с одинаковыми свойствами.

Атрибуты тега `colgroup`

Атрибут	Выравнивание содержимого столбца
<code>align</code>	Left - по левому краю (<i>по умолчанию</i>) Center - по центру Right - по правому краю
<code>dir</code>	Определяет направление символов: Ltr - слева направо Rtl - справа налево
<code>span</code>	Количество столбцов, к которым будет применяться оформление (<i>по умолчанию 1</i>)
<code>style</code>	Задаёт встроенную таблицу стилей
<code>valign</code>	Вертикальное выравнивание в ячейке таблицы Bottom - по нижнему краю ячейки Middle - по центру ячейки (<i>по умолчанию</i>) Top - по верхнему краю ячейки
<code>width</code>	Ширина столбца

(табл.4.42.5). Атрибуты тега `colgroup`

Создание HTML-документа с применением таблиц

1. Запустите программу Блокнот и наберите текст следующей Web-страницы. Применяйте *приемы копирования при создании таблицы*, работая в программе Блокнот.

```
<html>
<head>
<title> Создание таблицы </title>
</head>
<TABLE border="10" width="100%" bordercolordark="red"
bordercolorlight="brown">
<CAPTION> Пример разноцветной таблицы </CAPTION>
<TR bgcolor="white">
<TD>ячейка1,1</TD>
<TD>ячейка1,2</TD>
</TR>
<TR>
<TD bgcolor="gray">ячейка2,1</TD>
```

```
<TD>ячейка2,2</TD>
</TR>
</TABLE>
</html>
```

2. Сохраните файл в папке под именем index.
3. Для просмотра созданной Web-страницы загрузите браузер.

Результат:

ячейка1,1	ячейка1,2
ячейка2,1	ячейка2,2

(рис.4.42.2). Результат работы

Ссылки

Ссылки являются основой гипертекстовых документов и позволяют переходить с одной веб-страницы на другую.

Адрес ссылки может быть как абсолютным, так и относительным.

Абсолютные адреса должны начинаться с указания протокола и содержать имя сайта.

Относительные ссылки ведут отсчет от корня сайта или текущего документа.

Для создания ссылки необходимо сообщить браузеру, что является ссылкой, а также указать адрес документа, на который следует сделать ссылку. Оба действия выполняются с помощью тега `<a>`. Общий синтаксис создания ссылок следующий.

```
<a href="URL"> текст ссылки</a>
```

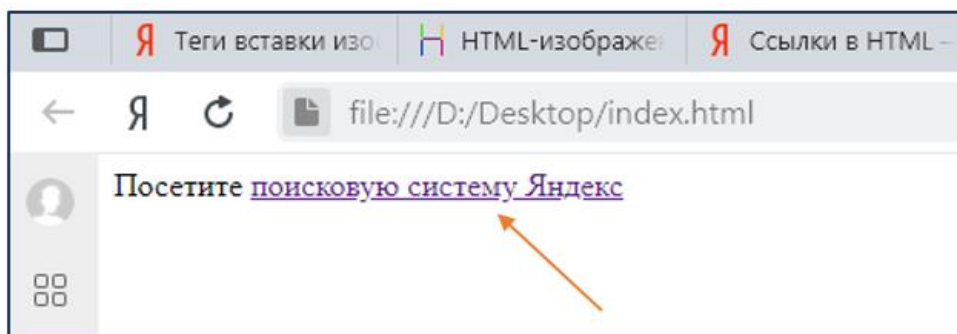
Атрибут `href` определяет URL (*универсальный указатель ресурса*), иными словами, адрес документа, на который следует перейти, а содержимое

контейнера `<a>` является ссылкой. Текст, расположенный между тегами `<a>` и ``, по умолчанию становится синего цвета и подчеркивается.

Пример:

```
<html>
<head>
<title> Ссылки в html </title>
</head>
<body>
Посетите
<a href="http://www.yandex.ru">
поисковую систему Яндекс
</a>
</body>
</html>
```

Результат:



(рис.4.42.3). Результат ссылки

Абсолютные ссылки обычно применяются для указания документа на другом *сетевом ресурсе*, впрочем, допустимо делать абсолютные ссылки и внутри текущего сайта.

Ссылки относительно текущего документа

При создании относительных ссылок надо понимать, какое значение для атрибута `href` следует указывать, поскольку оно зависит от исходного расположения файлов.

```
<a href="Ссылаемый документ.html"> Ссылка </a>
```

Внутренние ссылки в HTML (якорная ссылка)

В HTML ссылки, которые организуют переходы внутри одной страницы, называются *внутренними или якорными ссылками*.

Механизм создания внутренних ссылок в HTML. Он состоит из двух шагов:

1. Создание закладок или якорей (*на которые необходимо переходить по ссылкам*):

1 способ:

```
<a name="название_закладки"> </a>
```

```
<p> Текст для закладки </p>
```

В качестве якоря служит тег `a` с атрибутом `name` - название якоря (закладки)

2 способ:

```
<p id="название_закладки"> Текст для закладки </p>
```

Для обозначения якоря используется атрибут `id`, добавляемый к тегу (теги могут быть практически любые: `div`, `span`, `p`, `h`, ...)

2. Создание ссылок на якоря (*на закладки*):

```
<a href="#название_закладки"> Текст ссылки </a>
```

Знак шарп или решетка (`#`) ставится обязательно перед названием якоря.

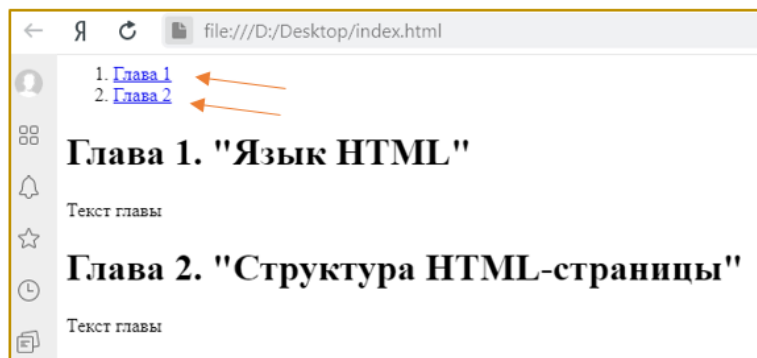
Пример:

```
<html>
<head>
<title> Ссылки в html </title>
</head>
<body>
<ol>
<!-- Создание ссылок -->
<li><a href="#glava1">Глава 1</a></li>
<li><a href="#glava2">Глава 2</a></li>
</ol>
<!-- Создание якоря -->
```

```

<h1 id="glava1">Глава 1. "Язык HTML"</h1>
<p> Текст главы </p>
<!-- Создание якоря -->
<h1 id="glava2">Глава 2. "Структура HTML-страницы"</h1>
<p> Текст главы </p>
</body>
</html>

```



(рис.4.42.4). Внутренние ссылки

HTML-изображения добавляются на веб-страницы с помощью тега ``. Использование графики делает веб-страницы визуально привлекательнее. Изображения помогают лучше передать суть и содержание веб-документа.

С помощью HTML-тегов `<map>` и `<area>` можно создавать карты-изображения с активными областями.

Теги вставки изображения

Название	Комментарии
<code></code>	Тег вставки изображения
<code>Src</code>	Задаёт адрес изображения
<code>Width</code>	Задаёт ширину изображения
<code>Height</code>	Задаёт высоту изображения
<code>Align=</code>	Выравнивание изображения <i>1. По вертикали</i> Bottom – нижняя граница изображения выравнивается по нижнему краю строки Middle – центр изображения выравнивается по нижнему краю строки Top – верхняя граница изображения выравнивается по верхнему краю строки <i>2. По горизонтали</i> Left – по левому краю страницы Right – по правому краю страницы

Border	Задает ширину рамки в пикселях
Title	Подсказка при наведении мыши

(табл.4.42.6). Теги вставки изображения

Пример:

```
<html>
<head>
<title> Размещение изображения в html </title>
</head>
<body>
  <p>
  <p>
  <p>
</body>
</html>
```



Контрольные вопросы и задания

1. Какие тэги используются для создания таблиц? Какие параметры содержит тэг?
2. Как объединить ячейки по вертикали?
3. Как объединить ячейки по горизонтали?
4. Какой тэг используется для создания гиперссылок, какие параметры он содержит?
5. Какие 4 типа гиперссылок бывают? Приведите примеры их использования.
6. Для чего нужен параметр href?
7. Какой тэг используется для вставки изображений, какие параметры содержит?
8. Для чего нужен параметр src?

43-§. СОЗДАНИЕ WEB-СТРАНИЦ С ПОМОЩЬЮ DREAMWEAVER

Цель работы: Научиться разрабатывать web-страницы в среде Macromedia Dreamweaver, *форматировать текст, вставлять графику и ссылки.*

Краткая теоретическая информация:

Редактор **Macromedia Dreamweaver** в своем последнем воплощении включает больше профессиональных средств и функций. При этом обладает широчайшими возможностями конфигурирования. Стили HTML позволяют определять для страниц *CSS-подобные форматы*, работающие в любом браузере, а тесная интеграция с *Fireworks* весьма упрощает создание графических элементов.

Настройка среды Dreamweaver

1. Открыт программу Dreamweaver
2. Чтобы создать новый документ, нужно войти в меню **ФАЙЛ (File)**.
3. Выбрать вкладку **Новый (New)**
4. В открывшемся окне предлагается набор категорий и виды документов, принадлежащих этим категориям.

5. Добавить к документу линейки, для этого зайти в меню **ВИД-Линейки -Сантиметры (Centimeters)**

Отобразить модульную сетку: **ВИД - Сетка – Установки сетки ...** и выполните для нее следующие настройки:

- цвета линии – *голубой*;
- установить флажок Show grid;
- установить промежуток между линиями сетки, *равной 38*;
- установить режим отображения сетки в положение **Dots (точки)**.

1. Задать заголовок страницы: *Пример*:
2. Установите цвет фона - *светло-зеленый*.
3. Создайте новую веб-страницу

4. Задайте заголовок страницы: *Пример:*

Работа с текстом

1. Запустить Dreamweaver в режиме *Designer*. Задать заголовок веб-страницы.

Пример: Введите текст и отформатируйте ее. Сохраните страницу под именем *1.html* (*дайте команду File – Save As. Создайте отдельную папку и сохраните в ней данный файл*).

Создание гиперссылки

1. Открыть сохраненный ранее файл *1.html*.

2. Создать гиперссылку на веб-страничку *Macromedia*.

3. Для того чтобы научиться работать с "якорями", необходимо создать новую веб-страничку. Скопировать из любого текстового файла несколько абзацев текста. После этого написать в начале страницы небольшое оглавление. И сохранить ее в файле *2.html*.

4. Создать 4 гиперссылки («якоря») на 4 абзаца текста. Проверить работу всех созданных гиперссылок.

5. Сохранить файлы.

6. Создать новую пустую web-страницу. Сохранить ее в файле *3.html*.

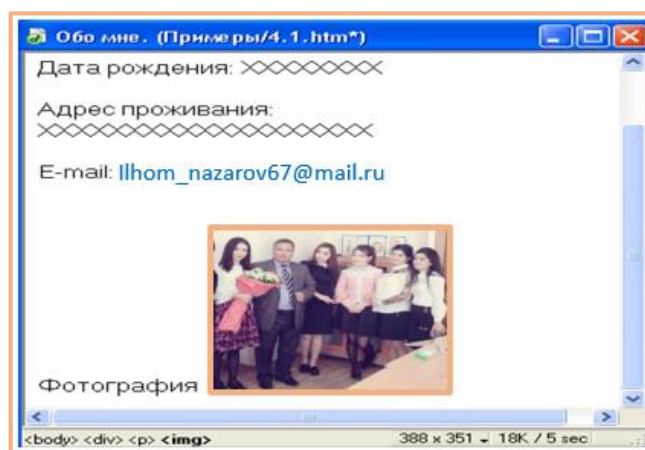
7. Открыть первую страницу (*файл 1.html*). Создать первую "внутри сайтовую" гиперссылку, т.е. гиперссылку, ведущую на другую страницу сайта. Для этого сначала выделить фамилию, имя и отчество: «*Назаров Ильхом Усманиевич*».

8. В редакторе свойств справа от поля ввода адреса находятся две кнопки. Одна из них имеет вид мишени. Вторая имеет вид папки и при нажатии открывает диалоговое окно **Select File**. В верхней части этого окна находится список файлов текущей папки и раскрывающийся список выбора папок, как в стандартном диалоговом окне открытия файлов **Windows**. Выбрать нужный файл Web-страницы (*3.html*), чтобы создать на нее гиперссылку. Нажать кнопку **ОК**.

9. Сохранить главную страницу.

Вставка графического изображения

1. Открыть файл *3.html*.
2. Создать с помощью программы Paint *портрет Назаров И.У* и сохранить в файле *Nazarov.gif*.
3. Перейти на страницу сведений об авторе. Вставить портрет сразу же после слова "Фотография".

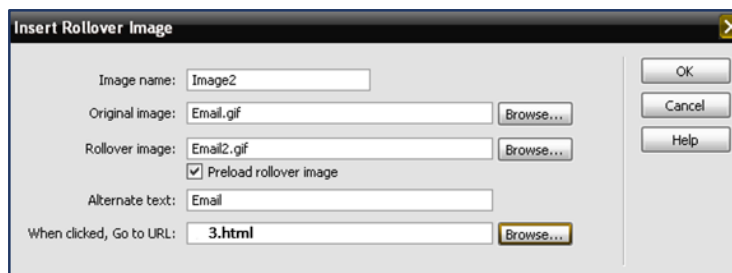


(рис.4.43.1). Результат работы

4. Уменьшить размеры вставленного изображения.

Создание активных изображений

1. Открыть страницу сведений об авторе *3.html*, куда уже вставили изображение-гиперссылку. Заменить ее активным изображением.
2. Создать в графическом редакторе два изображения со значком "книжный зал": одно - обычное, а второе - инвертированное, которое будет перекрывать первое. Первое изображение сохранить в файле *Email 1.gif*, а второе - в файле *Email 2.gif*.
3. Стереть в файле *3.html* всю строку с адресом электронной почты так, чтобы на ее месте остался пустой параграф. Текстовый курсор оставить на получившемся пустом параграфе.
4. Вставить активное изображение с помощью кнопки *Rollover Image*.
5. В соответствующие поля ввода подставлены все нужные значения. Ввести их и нажать кнопку **ОК**, после чего активное изображение будет создано. Изменить его размер.



(рис. 1.43.2). Настройка параметров при создании активного изображения

6. Проверить созданное активное изображение в действии. Для этого вывести его в Web-обозревателе для предварительного просмотра и поместить курсор мыши над активным изображением.

7. Результаты работы показать преподавателю.



Контрольные вопросы и задания

1. Какие типы графических файлов существуют?
2. Как выполнить вставку графического изображения?
3. Как создать изображение-гиперссылку?
4. Что такое карта-изображение?
5. Как создать карту-изображение?
6. Как изменить фон страницы?
7. Как разместить в качестве фона страницы графическое изображение из файла?

44-§. РАБОТА С РАЗЛИЧНЫМИ СРЕДСТВАМИ ЗАЩИТЫ ИНФОРМАЦИИ

Цель работы: *Формирование профессиональных компетенций, необходимых по средствам защиты информации*

Краткая теоретическая информация:

Информация и ее классификация

Информация - это сведения (сообщения, данные) независимо от формы их представления».

Информацию можно классифицировать по нескольким видам и в зависимости от категории доступа к ней подразделяется на общедоступную информацию, а также на информацию, доступ к которой ограничен - конфиденциальные данные и государственная тайна.

Информация в зависимости от порядка ее предоставления или распространения подразделяется на информацию:

1. свободно распространяемую;
2. предоставляемую по соглашению лиц, участвующих в соответствующих отношениях;
3. которая в соответствии с федеральными законами подлежит предоставлению или распространению;
4. распространение которой в Республики Узбекистан ограничивается или запрещается.

Информация по назначению бывает следующих видов:

1. массовая;
2. специальная;
3. секретная;
4. личная.

Средства защиты информации необходимо применять непосредственно к информации доступ к которой ограничен - это государственная тайна и конфиденциальные данные.

«Перечень сведений составляющих государственную тайну» относится:

1. сведения в военной области;
2. сведения в области экономики, науки и техники;
3. сведения в области внешней политики и экономики;

4. сведения в области *разведывательной, контрразведывательной и оперативно-розыскной деятельности*, а также в области *противодействия терроризму* и в области *обеспечения безопасности лиц*, в отношении которых принято решение о применении мер государственной защиты.

Конфиденциальные данные - это информация, доступ к которой ограничен в соответствии с законами государства и нормами, которые компании устанавливаются самостоятельно. Можно выделить следующие виды конфиденциальных данных:

- личные конфиденциальные данные;
- служебные конфиденциальные данные;
- судебные конфиденциальные данные;
- коммерческие конфиденциальные данные;
- профессиональные конфиденциальные данные.

Основные носители информации:

- печатные и электронные средства массовой информации, социальные сети, другие ресурсы в интернете;
- сотрудники организации, у которых есть доступ к информации на основании своих связей;
- средства связи, которые передают или сохраняют информацию (*телефоны, АТС, другое телекоммуникационное оборудование*);
- документы всех типов (*личные, служебные, государственные*);
- программное обеспечение как самостоятельный информационный объект, особенно если его версия дорабатывалась специально для конкретной компании;
- электронные носители информации, которые обрабатывают данные в автоматическом порядке.

В соответствии законом Республики Узбекистан от 02.07.2019 г. N ЗРУ-547 "О персональных данных" (*Принят Законодательной палатой 16.04.2019 г., одобрен Сенатом 21.06.2019 г.*)

Государство гарантирует защиту персональных данных.

Собственник и *(или)* оператор, а также третье лицо принимают правовые, организационные и технические меры по защите персональных данных, обеспечивающие:

- реализацию права субъекта на защиту от вмешательства в его частную жизнь;
- целостность и сохранность персональных данных;
- соблюдение конфиденциальности персональных данных;
- предотвращение незаконной обработки персональных данных.

Конфиденциальность персональных данных

Конфиденциальностью персональных данных является обязательное для соблюдения собственником и *(или)* оператором или иным получившим доступ к персональным данным лицом требование о недопустимости их раскрытия и распространения без согласия субъекта или наличия иного законного основания.

Собственник и *(или)* оператор и иные лица, получившие доступ к персональным данным, обязаны не раскрывать и не распространять персональные данные без согласия **субъекта**.

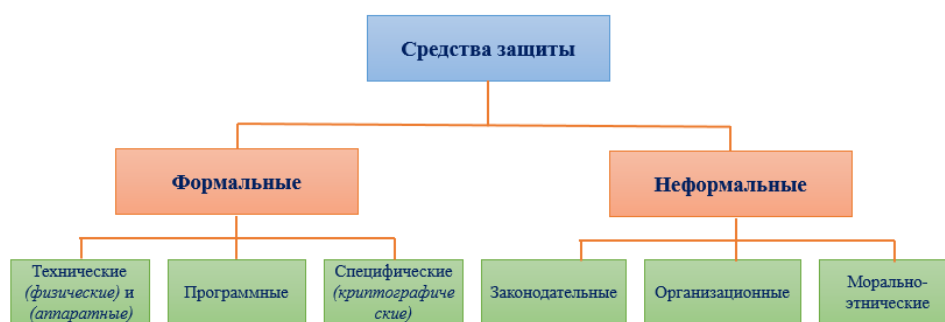
Классификация средств защиты информации

Защита информации представляет собой принятие правовых, организационных и технических мер, направленных на:

- обеспечение защиты информации от *неправомерного доступа, уничтожения, модифицирования, блокирования, копирования, предоставления, распространения*, а также от иных неправомерных действий в отношении такой информации;
- соблюдение конфиденциальности информации ограниченного доступа;
- реализацию права на доступ к информации.

Средства защиты информации

Средства защиты информации принято делить на нормативные *(неформальные)* и технические *(формальные)*.



(рис.4.44.1). Средства защиты

Неформальные средства защиты информации

Неформальными средствами защиты информации - являются нормативные (*законодательные*), административные (*организационные*) и морально-этические средства, к которым можно отнести: документы, правила, мероприятия. Правовую основу (*законодательные средства*) информационной безопасности обеспечивает государство.

Формальные средства защиты - это специальные технические средства и программное обеспечение, которые можно разделить на *физические, аппаратные, программные и криптографические*.

Физические средства защиты информации - это любые *механические, электрические и электронные механизмы*, которые функционируют независимо от информационных систем и создают препятствия для доступа к ним.

Аппаратный средства защиты информации - это любые *электрические, электронные, оптические, лазерные и другие устройства*, которые встраиваются в информационные и телекоммуникационные системы: специальные компьютеры, системы контроля сотрудников, защиты серверов и корпоративных сетей. Они препятствуют доступу к информации, в том числе с помощью её маскировки.

Программные средства защиты информации - это простые и комплексные программы, предназначенные для решения задач, связанных с обеспечением информационной безопасности.

Аутентификация и идентификация

Чтобы исключить неправомерный доступ к информации применяют такие способы, как *идентификация* и *аутентификация*.

Идентификация - это механизм присвоения собственного уникального имени или образа пользователю, который взаимодействует с информацией.

Аутентификация - это система способов проверки совпадения пользователя с тем образом, которому разрешен допуск.

Эти средства направлены на то, чтобы предоставить или, наоборот, запретить допуск к данным. Подлинность, как правила, определяется тремя способами: *программой, аппаратом, человеком*. При этом объектом аутентификации может быть не только человек, но и техническое средство (*компьютер, монитор, носители*) или данные. Простейший способ защиты - пароль.



Контрольные вопросы и задания

1. Дайте определение понятиям: *«информация»*, *«информационная безопасность»*, *«защита информации»*, *«информационная угроза»*.
2. Дайте характеристику основным составляющим информационной безопасности.
3. Перечислите основные объекты защиты.
4. Дайте характеристику понятиям *«государственная тайна»*, *«конфиденциальная информация»* и *«персональные данные»*.
5. Дайте характеристику средствам защиты информации.
6. Перечислите способы тайной передачи информации на расстоянии.

ГЛАВА 5. ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ИХ ИСПОЛЬЗОВАНИЕ

45-§. ОСНОВНЫЕ ПОНЯТИЯ ИНФОРМАЦИОННЫХ СИСТЕМ.

Ключевые слова: *информационная система, мэйнфрейм, процесс, система, подсистема, модель, экспертные информационные системы*

Под **системой** понимают любой объект, который одновременно рассматривается и как единое целое, и как объединенная в интересах достижения поставленных целей совокупность разнородных элементов. Системы отличаются между собой как по составу, так и по главным целям.

Информационной системой - *называется взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели.*

Информационные системы обеспечивают *сбор, хранение, обработку, поиск, выдачу информации*, необходимой в процессе принятия решений задач из любой области. Они помогают анализировать проблемы и создавать новые продукты. В качестве основного технического средства переработки информации используют *персональный компьютер (ПК)*. В крупных организациях наряду с персональным компьютером в состав технической базы информационной системы может входить *мэйнфрейм* или *суперЭВМ*.

Особую роль в информационных системах отводится человеку, т.к. техническое воплощение информационной системы само по себе ничего не будет значить, если не учтена роль человека, для которого предназначена производимая информация и без которого невозможно ее получение и представление.

Необходимо понимать разницу между *компьютерами* и *информационными системами*. Компьютеры, оснащенные специализированными программными средствами, являются технической базой и инструментом для информационных систем.

Информационная система немислима без персонала, взаимодействующего с компьютерами и телекоммуникациями.

Изменение подхода к использованию ИС - информационных систем.

Период времени	Концепция использования информации	Вид информационной системы - ИС	Цель использования ИС
1950-1960 гг.	Бумажный поток расчетных документов	ИТ обработки расчетных документов на электромеханических бухгалтерских машинах	Повышение скорости обработки документов. Упрощение процедуры обработки счетов и расчетов.
1960-1970 гг.	Основная помощь в подготовке отчетов	Управленческие ИТ для производственной информации	Ускорение процесса подготовки отчетности
1970-1980 гг.	Управленческий контроль реализации	Системы поддержки принятия решений. Системы для высшего звена управления.	Выработка наиболее рационального решения
1980-2000 гг.	Информация - стратегический ресурс, обеспечивающий конкурентное преимущество	Стратегические ИТ. Автоматизированные подразделения	Повышение конкурентоспособности

(табл.5.45.1). Этапы развития информационных систем

1. В 50-е гг. была осознана роль информации как важнейшего ресурса предприятия, организации, региона, общества в целом; начали разрабатывать автоматизированные ИС разного рода. Первые ИС были предназначены исключительно для обработки *счетов и расчета зарплаты*, а реализовывались на электромеханических бухгалтерских счетных машинах. Это приводило к некоторому сокращению затрат и времени на подготовку бумажных документов. Вначале, когда появилась возможность обработки информации с помощью вычислительной техники, был распространен термин "системы обработки данных" (СОД), этот термин широко использовался при разработке систем радиуправления ракетами и другими космическими объектами, при создании систем сбора и обработки статистической информации о состоянии атмосферы, *учетно-отчетной информации предприятий* и т.п. По мере

увеличения памяти ЭВМ основное внимание стали уделять проблемам организации БД (*БД*). Это направление сохраняет определенную самостоятельность и в настоящее время и занимается в основном разработкой и освоением средств *технической* и *программной* реализации обработки данных с помощью вычислительных машин разного рода. Для сохранения этого направления по мере его развития появились термины "*базы знаний*", "*базы целей*", позволяющие расширить толкование проблемы собственно создания и обработки БД до задач, которые ставятся в дальнейшем при разработке ИС.

2. 60-е гг. знаменуются изменением отношения к ИС. Информация, полученная из них, стала применяться для периодической отчетности по многим параметрам. Для этого организациям требовалось компьютерное оборудование широкого назначения, способное обслуживать множество функций, а не только обрабатывать счета и считать зарплату, как было ранее. Основные черты этого поколения ИС - информационных систем:

- техническое обеспечение систем составляли маломощные *ЭВМ 2-3 поколения*;
- информационное обеспечение (*ИО*) представляло собой массивы (*файлы*) данных, структура которых определялась той программой, в которой они использовались;
- программное обеспечение специализированные прикладные программы;
- архитектура ИС - централизованная. Как правило, применялась пакетная обработка задач. Конечный пользователь не имел непосредственного контакта с ИС, вся предварительная обработка информации и ввод производились *персоналом ИС*.

Недостатки ИС - информационных систем - этого поколения:

- сильная взаимосвязь между программами и данными, то есть изменения в предметной области приводили к изменению структуры данных, а это заставляло переделывать программы.

- трудоемкость разработки и модификации систем.
- сложность согласования частей системы, разработанных разными людьми в разное время.

3. В 70-х - начале 80-х гг. ИС предприятий начинают использоваться в качестве средства управления производством, поддерживающего и ускоряющего процесс подготовки и принятия решений. В своем большинстве ИС этого периода предназначались для решения установившихся задач, которые четко определялись на этапе создания системы и затем практически не изменялись. Появление персональных ЭВМ приводит к корректировке идеи АСУ от ВЦ и централизации управления к распределенному вычислительному ресурсу и децентрализации управления. Такой подход нашел свое применение в системах поддержки принятия решения (*СППР*), которые характеризуют новый этап компьютерной ИТ организационного управления. При этом уменьшается нагрузка на централизованные вычислительные ресурсы и верхние уровни управления, что позволяет сосредоточить в них решение крупных долгосрочных стратегических задач. Жизнеспособность любой ИТ в немалой степени зависит от *оперативного доступа пользователей* к централизованным ресурсам и уровня информационных связей как по "*горизонтали*", так и по "*вертикали*" в пределах организационной структуры. В то же время для обеспечения эффективного управления крупными предприятиями была развита и остается актуальной идея создания интегрированных АСУ.

4. К концу 80-х гг. концепция использования ИС вновь изменяется. Они становятся *стратегическим источником информации* и используются на всех уровнях предприятия любого профиля. Стремление преодолеть недостатки предыдущего поколения ИС породило технологию *создания и управления базами данных*. База данных создается для группы взаимосвязанных задач, для многих пользователей и это позволяет частично решить проблемы ранее созданных ИС. Вначале СУБД разрабатывались для больших ЭВМ, и их количество не превышало десятка. Благодаря появлению ПЭВМ технология

БД стала массовой, создано большое количество инструментальных средств и СУБД для разработки ИС, что в свою очередь вызвало появление большого количества прикладных ИС в прикладных областях.

Основные черты ИС этого поколения:

- основу ИО составляет база данных,
- программное обеспечение состоит из прикладных программ и СУБД.
- технические средства: ЭВМ 3-4 поколения и ПЭВМ.
- средства разработки ИС: процедурные языки программирования 3-4 поколения, расширенные языком работы с БД (*SQL, QBE*).
- архитектура ИС: наиболее популярны две разновидности: персональная локальная ИС, централизованная БД с сетевым доступом.

5. Большим шагом вперед явилось развитие принципа *"дружественного интерфейса"* по отношению к пользователю. Например, повсеместно применяется графический интерфейс, развитые системы помощи и подсказки пользователю, разнообразные инструменты для упрощения разработки ИС: системы быстрой разработки приложений (*RAD-системы*), средства автоматизированного проектирования ИС (*CASE-средства*). К концу 80-х годов выявились и недостатки систем этого поколения.

6. Недостатки информационных систем (*ИС*) этого поколения:

- большие капиталовложения в компьютеризацию предприятий не дали ожидаемого эффекта, соответствующего затратам (*увеличились накладные расходы, но не произошло резкого повышения производительности*);
- внедрение ИС столкнулось с инертностью людей, нежеланием конечных пользователей менять привычный стиль работы, осваивать новые технологии;
- к квалификации пользователей стали предъявляться более высокие требования (*знание ПК, конкретных прикладных программ и СУБД, способность постоянно повышать свою квалификацию*).

Современные информационные системы

В связи с указанными выше недостатками постепенно стало формироваться современное поколение ИС. Техническая платформа - мощные ЭВМ 4-5 поколения, использование разных платформ в одной ИС (*большие ЭВМ, мощные стационарные ПК, мобильные ПК*). Наиболее характерно широкое применение вычислительных сетей - от локальных до глобальных. Информационное обеспечение: ведутся интенсивные разработки с целью повышения интеллектуальности банка данных в следующих направлениях:

- новые модели знаний, учитывающие не только структуру информации, но и активный характер знаний,
- средства оперативного анализа информации (*OLAP*) и средства поддержки принятия решений (*DSS*),
- новые формы представления информации, более естественные для человека (*мультимедиа, полнотекстовые БД, гипертекстовые БД, средства восприятия и синтеза речи*).

В последнее время появился широкий спектр специализированных ИС - экономические ИТ (*ЭИС*), бухгалтерские ИТ (*БУИС*), банковские ИТ (*БИС*), ИТ рынка ценных бумаг, маркетинговые ИС (*МИС*) и т.п.

Современная ИС определяется следующими свойствами:

- может быть подвергнута анализу, построена и управляема на основе общих принципов построения систем;
- является динамичной и развивающейся;
- при построении необходимо использовать системный подход;
- выходной продукцией является информация, на основе которой принимаются решения;
- следует воспринимать как автоматизированную, т.е. человеко-компьютерную систему обработки информации.

Процессы, протекающие в информационной системе.

Процессы, обеспечивающие работу информационной системы:

- ввод информации из внешних или внутренних источников;
- обработка входной информации и представление ее в удобном виде;
- вывод информации для представления потребителям или передачи в другую систему;

- обратная связь - это информация, переработанная людьми данной организации для коррекции входной информации.

Внедрение информационных систем может способствовать:

- получению более рациональных вариантов решения управленческих задач за счет внедрения математических методов и интеллектуальных систем и т.д.;

- освобождению работников от рутинной работы за счет ее автоматизации;

- обеспечению достоверности информации;

- замене бумажных носителей данных на магнитные диски или ленты;

- совершенствованию структуры потоков информации и системы документооборота в фирме;

- уменьшению затрат на производство продуктов и услуг;

- предоставлению потребителям уникальных услуг;

- отысканию новых рыночных ниш;

- привязке к фирме покупателей и поставщиков за счет предоставления им разных скидок и услуг.

Структура информационной системы

Структуру информационной системы составляет совокупность отдельных ее частей, называемых подсистемами.

Подсистема - это часть системы, выделенная по какому-либо признаку.

Общую структуру информационной системы можно рассматривать как совокупность подсистем независимо от сферы применения, а подсистемы называют *обеспечивающими*. Структура любой информационной системы может быть представлена совокупностью обеспечивающих подсистем:

информационное обеспечение, программное обеспечение, техническое обеспечение, математическое обеспечение, правовое обеспечение, организационное обеспечение.

Классификация ИС по степени автоматизации

В зависимости от степени автоматизации информационных процессов в системе управления фирмой информационные системы определяются как *ручные, автоматические, автоматизированные*.

Ручные ИС характеризуются отсутствием современных технических средств переработки информации и выполнением всех операций человеком. **Автоматические ИС** выполняют все операции по переработке информации *без участия человека*. Автоматизированные ИС предполагают участие в процессе обработки информации и человека, и технических средств, причем главная роль отводится *компьютеру*.

Классификация информационных систем по признаку структурированности задач.

Различают три типа задач, для которых создаются информационные системы: структурированные (*формализуемые*), неструктурированные (*неформализуемые*) и частично структурированные.

Структурированная (формализуемая) задача - задача, где известны все ее элементы и взаимосвязи между ними.

Неструктурированная (неформализуемая) задача - задача, в которой невозможно выделить элементы и установить между ними связи.

В структурированной задаче удастся выразить ее содержание в форме *математической модели*, имеющей точный алгоритм решения подобные задачи, обычно приходится решать многократно, и они носят рутинный характер.

Целью использования ИС для решения структурированных задач является *полная автоматизация* их решения, т.е. сведение роли человека к нулю.

Решение неструктурированных задач из-за невозможности создания математического описания и разработки алгоритма связано с большими трудностями. Возможности использования здесь информационной системы невелики решение в таких случаях принимается человеком из эвристических соображений на основе своего опыта и, косвенной информации из разных источников.

В практике работы любой организации существует сравнительно немного полностью *структурированных* или совершенно *неструктурированных* задач. О большинстве задач можно сказать, что известна лишь часть их элементов и связей между ними. Такие задачи называются *частично структурированными*. В этих условиях можно создать ИС, получаемая в ней информация анализируется человеком, который будет играть определяющую роль. Такие информационные системы являются автоматизированными, так как в их функционировании принимает участие человек.

ИС, используемые для решения частично структурированных задач, подразделяются на два вида: создающие управленческие отчеты и ориентированные главным образом на обработку данных (*поиск, сортировку, агрегирование, фильтрацию*). Используя сведения, содержащиеся в этих отчетах, управляющий принимает решение. Разрабатывающие возможные альтернативы решения. Принятие решения при этом сводится к выбору одной из предложенных альтернатив.

Информационные системы, создающие управленческие отчеты, обеспечивают информационную поддержку пользователя, т.е. предоставляют доступ к информации в базе данных и ее частичную обработку.

Информационные системы, разрабатывающие альтернативы решений, могут быть модельными или экспертными.

Модельные информационные системы предоставляют пользователю математические, статистические, финансовые и другие модели, использование которых облегчает выработку и оценку альтернатив решения. Пользователь

может получить недостающую ему для принятия решения информацию путем установления диалога с моделью в процессе ее исследования. Основными функциями модельной информационной системы являются:

- возможность работы в среде типовых математических моделей, включая решение основных задач моделирования типа "как сделать, чтобы?", "что будет, если?", анализ чувствительности и др.;
- достаточно быстрая и адекватная интерпретация результатов моделирования;
- оперативная подготовка и корректировка входных параметров и ограничений модели;
- возможность графического отображения динамики модели;
- возможность объяснения пользователю необходимых шагов формирования и работы модели.

Экспертные информационные системы обеспечивают выработку и оценку возможных альтернатив пользователем за счет создания экспертных систем, связанных с обработкой знаний.



Контрольные вопросы и задания

1. Понятие системы
2. Информационная система
3. Этапы развития информационных систем
4. Процессы, протекающие в информационной системе
5. Внедрение информационных систем
6. Структура информационной системы
7. Классификация ИС

46-§. ПОНЯТИЕ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ.

Ключевые слова: *эффективность, угроза, информационная безопасность, защита информации, ресурсы, риск*

Проблема обеспечения *информационной безопасности* в рамках любого государства в последнее время все чаще является предметом обсуждения не только в научных кругах, но и на политическом уровне. Данная проблема также становится объектом внимания международных организаций, в том числе и ООН.



Информационная безопасность – это защищенность информационной среды общества посредством различных средств и методов.

Под информационной средой понимается совокупность информационных ресурсов и наличие соответствующей инфраструктуры их создания и использования.

Целью информационной безопасности является предотвращение влияния неблагоприятных событий (*угроз*) или обеспечение минимального ущерба от них информационной среде.

Современный этап развития общества характеризуется возрастающей ролью *электронных ресурсов*, представляющих собой совокупность информации, информационной инфраструктуры, субъектов, осуществляющих *сбор, формирование, распространение и использование информации*, а также системы регулирования возникающих при этом отношений.

Стремительный рост компьютерных технологий в различных сферах человеческой деятельности, с одной стороны, позволил обеспечить высокие достижения в этих сферах, а с другой стороны, стал источником самых непредсказуемых и вредных для человеческого общества последствий. В результате, можно говорить о появлении принципиально нового сегмента международного противоборства, затрагивающего как вопросы безопасности отдельных государств, так и общую систему международной безопасности на всех уровнях.

Становится очевидной необходимость теоретической разработки международно-правовых основ регулирования взаимоотношений субъектов международного права в сфере качественно изменяющихся под воздействием информационной революции условий обеспечения международной и национальной безопасности, в сфере обеспечения информационной безопасности государства.

Новые технологии порождают и новые преступления. Согласно унификации Комитета министров Европейского Совета, определены криминальные направления компьютерной деятельности. К ним относятся:

- компьютерное мошенничество;
- подделка компьютерной информации;
- повреждение данных или программ;
- компьютерный саботаж;
- несанкционированный доступ к информации;
- нарушение авторских прав.

Актуальность проблемы информационной безопасности заключается:

- в особом характере общественной опасности возможных преступлений;
- в наличии тенденции к росту числа преступлений в информационной сфере;
- в не разработанности ряда теоретических положений, связанных с информационной безопасностью.

Проблема обеспечения безопасности носит комплексный характер, для ее решения необходимо сочетание *законодательных, организационных и программно-технических* мер и средств. В курсе Информационной безопасности нас будут интересовать программно-технические средства, реализующиеся программным и аппаратным обеспечением, решающие разные задачи по защите. Они могут быть встроены в операционные системы,

либо могут быть реализованы в виде отдельных продуктов. Во многих случаях центр тяжести смещается в сторону защищенности операционных систем.

23 декабря 1999 года Генеральная Ассамблея ООН приняла резолюцию, в которой выражается озабоченность тем, что распространение и использование современных информационных технологий и средств потенциально может быть использовано в целях, несовместимых с задачами обеспечения международной стабильности и безопасности.

Специалистам в области информационной безопасности сегодня почти невозможно обойтись без знаний соответствующих стандартов и спецификаций. На то имеется несколько причин.

Формальная состоит в том, что необходимость следования некоторым стандартам закреплена законодательно. Однако наиболее убедительны содержательные причины.

Во-первых, стандарты и спецификации - одна из форм накопления знаний, прежде всего о процедурном и программно-техническом уровнях информационной безопасности. В них зафиксированы апробированные, высококачественные решения и методологии, разработанные наиболее квалифицированными специалистами.

Во-вторых, и те, и другие являются основным средством обеспечения взаимной совместимости аппаратно - программных систем и их компонентов.

Среди множества различных стандартов и спецификаций, можно выделить две группы документов, которые будут рассмотрены в данном курсе:

- *оценочные стандарты*, предназначенные для оценки и классификации информационных систем и средств защиты по требованиям безопасности;
- *спецификации*, регламентирующие различные аспекты реализации и использования средств и методов защиты.

Эти группы, разумеется, дополняют друг друга. Оценочные стандарты описывают важнейшие, с точки зрения информационной безопасности, понятия и аспекты информационных систем (ИС), играя роль

организационных и архитектурных спецификаций. Спецификации определяют, как именно строить ИС предписанной архитектуры и выполнять организационные требования.

Из числа оценочных необходимо выделить стандарт Министерства обороны США «Критерии оценки доверенных компьютерных систем» (*Trusted Computer System Evaluation Criteria – TCSEC*) и его интерпретацию для сетевых конфигураций (*Trusted Network Interpretation*), «Гармонизированные критерии Европейских стран» (*Information Technology Security Evaluation Criteria (ITSEC)*). Harmonised Criteria of France - Germany- the Netherlands - the United Kingdom, международный стандарт «Критерии оценки безопасности информационных технологий» (*Common Criteria for Information Technology Security Evaluation (CCITSE)*). К этой же группе относятся: Федеральный стандарт США (*Federal Information Processing Standardization, FIPS*) «Требования безопасности для криптографических модулей» (*FIPS 140-2*), Британский стандарт BS 7799 часть 2 «Управление информационной безопасностью. Практические правила» (*Information Security Management Systems - Specification with guidance for use*), а также международный стандарт ISO/IEC 17799 «Информационная технология. Практический кодекс по менеджменту информационной безопасности» (*Information Technology – Code of practice for information security management*), являющийся изложением BS 7799 часть 1.

Технические спецификации, применимые к современным распределенным ИС, создаются, главным образом, «Тематической группой по технологии Интернет» (*Internet Engineering Task Force, IETF*) и ее подразделением – рабочей группой по безопасности. Ядром рассматриваемых технических спецификаций служат документы по безопасности на IP-уровне (*IPsec*). Кроме этого, анализируется защита на транспортном уровне (*Transport Layer Security, TLS*), а также на уровне приложений (*спецификации GSS-API, Kerberos*). Акцентируется внимание на административном и процедурном уровнях безопасности («Руководство по информационной безопасности

предприятия», «Как выбирать поставщика интернет-услуг», «Как реагировать на нарушения информационной безопасности»).

В вопросах сетевой безопасности невозможно разобраться без освоения спецификаций:

X.800 *«Архитектура безопасности для взаимодействия открытых систем»;*

X.500 *«Служба каталогов: обзор концепций, моделей и сервисов»;*

X.509 *«Служба каталогов: каркасы сертификатов открытых ключей и атрибутов».*

Это *«стандартный минимум»*, которым должны активно владеть все действующие специалисты в области информационной безопасности.

По существу, проектирование системы безопасности подразумевает ответы на следующие вопросы:

- какую информацию защищать;
- какого рода атаки на безопасность системы могут быть предприняты;
- какие средства использовать для защиты информации каждого вида.

Поиск ответов на данные вопросы называется формированием политики безопасности, которая помимо чисто технических аспектов включает также и решение организационных проблем. На практике реализация политики безопасности состоит в присвоении субъектам и объектам идентификаторов, фиксации набора правил, позволяющих определить, имеет ли данный субъект авторизацию, достаточную для предоставления к данному объекту указанного типа доступа.

Формируя политику безопасности, необходимо учитывать несколько базовых принципов. Так, **Зальтцер** и **Шредер** на основе своего опыта работы сформулировали следующие рекомендации для проектирования системы безопасности операционных систем:

- Проектирование системы должно быть открытым. Нарушитель и так все знает (*криптографические алгоритмы открыты*).

- Не должно быть доступа по умолчанию. Ошибки с отклонением легитимного доступа будут обнаружены скорее, чем ошибки там, где разрешен неавторизованный доступ.

- Нужно тщательно проверять текущее авторство. Так, многие системы проверяют привилегии доступа при открытии файла и не делают этого после. В результате пользователь может открыть файл и держать его открытым в течение недели и иметь к нему доступ, хотя владелец уже сменил защиту.

- Давать каждому процессу минимум возможных привилегий.

- Защитные механизмы должны быть просты, постоянны и встроены в нижний слой системы, это не аддитивные добавки (*известно много неудачных попыток «улучшения» защиты слабо приспособленной для этого ОС MS-DOS*).

- Важна физиологическая приемлемость. Если пользователь видит, что защита требует слишком больших усилий, он от нее откажется.

- Ущерб от атаки и затраты на ее предотвращение должны быть сбалансированы.

Приведенные соображения показывают необходимость продумывания и встраивания защитных механизмов на самых ранних стадиях проектирования системы.



Контрольные вопросы и задания

1. Дайте определение информационной безопасности.
2. Чем определяется информационная безопасность?
3. Какие исторические события можно связать с понятием «нарушение информационной безопасности»?
4. Что относится к субъектам информационных отношений?
5. Перечислите виды компьютерных преступлений.

6. Какие руководящие документы существуют в области информационной безопасности?

7. Какие стандарты и спецификации информационной безопасности вы знаете?

8. Приведите список угроз безопасности информационной системы.

9. Приведите список мер противодействия угрозам информационной безопасности.

47-§. ЭЛЕКТРОННО-ЦИФРОВАЯ ПОДПИСЬ.

Ключевые слова: *электронно-цифровая подпись, ключ проверки, подпись, счет-фактура, учет, криптостойкость, защита.*

Электронно-цифровая подпись (ЭЦП) - это реквизит электронного документа, предназначенный для защиты данного электронного документа от подделки, полученный в результате криптографического преобразования информации с использованием закрытого ключа электронной цифровой подписи и позволяющий идентифицировать владельца сертификата ключа подписи, а также установить отсутствие искажения информации в электронном документе.

Электронная-цифровая подпись - *комбинация знаков или паролей, которая служит эквивалентом обычной подписи на бумаге.*

Электронно-цифровая подпись - это программно-криптографическое средство, которое обеспечивает:

- проверку целостности документов;
- конфиденциальность документов;
- установление лица, отправившего документ.

Преимущества использования электронно-цифровой подписи

Использование электронно-цифровой подписи позволяет:

- значительно сократить время, затрачиваемое на оформление сделки и обмен документацией;
- усовершенствовать и удешевить процедуру подготовки, доставки, учета и хранения документов;
- гарантировать достоверность документации;
- минимизировать риск финансовых потерь за счет повышения конфиденциальности информационного обмена;
- построить корпоративную систему обмена документами.

Виды электронно-цифровой подписи.

Существует **три вида** электронной цифровой подписи:

- простая электронно-цифровая подпись;
- усиленная неквалифицированная электронно-цифровая подпись;
- усиленная квалифицированная электронно-цифровая подпись.



(рис.5.47.1). Электронно-цифровая подпись.

Простая электронно-цифровая подпись.

Посредством использования кодов, паролей или иных средств, простая электронно-цифровая подпись подтверждает факт формирования электронной подписи определенным лицом.

Простая электронно-цифровая подпись имеет низкую степень защиты. Она позволяет лишь определить автора документа. Простая электронно-цифровая подпись *не защищает документ от подделки.*

Усиленная неквалифицированная электронно-цифровая подпись.

1. получена в результате криптографического преобразования информации с использованием ключа электронной подписи;

2. позволяет определить лицо, подписавшее электронный документ;
3. позволяет обнаружить факт внесения изменений в электронный документ после момента его подписания;
4. создается с использованием средств электронной подписи.

Усиленная неквалифицированная электронно-цифровая подпись имеет *среднюю степень защиты*. Чтобы использовать неквалифицированную электронную подпись, необходим сертификат ключа ее проверки.

Усиленная квалифицированная электронно-цифровая подпись

Для квалифицированной электронной подписи характерны признаки неквалифицированной электронной подписи. Усиленная квалифицированная электронно-цифровая подпись соответствует следующим дополнительным признакам подписи:

1. ключ проверки электронной подписи указан в квалифицированном сертификате;
2. для создания и проверки электронной подписи используются средства электронной подписи, получившие подтверждение соответствия требованиям законодательства.

Усиленная квалифицированная электронно-цифровая подпись является наиболее универсальной и стандартизированной подписью с высокой степенью защиты. Документ, визированный такой подписью, аналогичен *бумажному варианту с собственноручной подписью*.

Использовать такую подпись можно и без каких-либо дополнительных соглашений и регламентов между участниками электронного документооборота. Если под документом стоит квалифицированная подпись, можно точно определить, какой именно сотрудник организации ее поставил. А также установить, изменялся ли документ уже после того, как был подписан.

Когда применяются разные виды подписи

Простая электронно-цифровая подпись

Обращение заявителей - юридических лиц за получением государственных и муниципальных услуг осуществляется путем подписания

обращения уполномоченным лицом с использованием простой электронной подписи.

Использование простой электронной подписи для получения государственной или муниципальной услуги допускается, если федеральными законами или иными нормативными актами не установлен запрет на обращение за получением государственной или муниципальной услуги в электронной форме, а также не установлено использование в этих целях иного вида электронной подписи.

Усиленная квалифицированная электронно-цифровая подпись

Для некоторых видов отчетности использование квалифицированной подписи прямо определено нормативными документами.

Электронный счет-фактуру следует подписывать только усиленной квалифицированной электронной подписью руководителя либо иных лиц, уполномоченных на это приказом (*иным распорядительным документом*) или доверенностью от имени организации, индивидуального предпринимателя.

Заявление о постановке на учет (*снятии с учета*) заверяется только усиленной квалифицированной подписью.

Криптостойкость средств электронно-цифровой подписи.

Симметрии	Несимметрии	Время (<i>скорость</i>)
56	384	1–2 сек.
128	2304	30 сек.
1024	10000	10 мин.
2048	30000	сутки
4096	100000	не вскрывается

Продолжительность реконструкции (*подбора или взлома*) ключа определяется производительностью компьютера и длиной ключа.

Защита информации.

Целями защиты информации являются:

✓ предотвращение утечки, хищения, утраты, искажения, подделки информации;

- ✓ предотвращение угроз безопасности личности, общества, государства;

- ✓ предотвращение несанкционированных действий по уничтожению, модификации, искажению, копированию, блокированию информации; предотвращение других форм незаконного вмешательства в информационные ресурсы и информационные системы, обеспечение правового режима документированной информации как объекта собственности;

- ✓ защита конституционных прав граждан на сохранение личной тайны и конфиденциальности персональных данных, имеющих в информационных системах; – сохранение государственной тайны, конфиденциальности документированной информации в соответствии с законодательством;

- ✓ обеспечение прав субъектов в информационных процессах и при разработке, производстве и применении информационных систем, технологий и средств их обеспечения.

Принцип работы ЭЦП

Механизм работы электронной подписи: при подписании документа добавляется блок данных, который и называется ЭЦП. Процесс его получения происходит в два этапа. С помощью ПО и математической функции происходит вычисление отпечатка сообщения, обладающего некоторыми особенностями:

- ✓ По отпечатку сообщения невозможно восстановить тело документа;

- ✓ Отпечаток уникален для каждого сообщения и имеет фиксированную длину.

- ✓ На втором этапе ПО и ключ шифрует отпечаток, расшифровать который можно будет только открытым личным ключом электронной подписи.

Использование электронной цифровой подписи упрощает ведение документооборота и сотрудничество с государственными и муниципальными

органами, дает возможность осуществления закупок и участия в официальных торгах. Документ, заверенный ЭЦП, приравнивается по юридической силе подписанному собственноручно.



Контрольные вопросы и задания

1. Электронно-цифровая подпись?
2. Преимущества использования электронно-цифровой подписи
3. Виды электронно-цифровой подписи
4. Простая электронно-цифровая подпись
5. Усиленная неквалифицированная электронно-цифровая подпись
6. Усиленная квалифицированная электронно-цифровая подпись

ПРАКТИЧЕСКИЕ РАБОТЫ

48-§. АВТОМАТИЗИРОВАННЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

Цель работы: *получить представление об автоматических и автоматизированных системах управления в технической сфере деятельности.*

Краткая теоретическая информация:

Автоматизированная информационная система - это совокупность программных и аппаратных средств, предназначенных для хранения и (или) управления данными и информацией, а также для производства вычислений.

АИС объединяет следующие составляющие:

- языковые средства и правила, используемые для отбора, представления и хранения информации, для отображения картины реального мира в модель данных, для представления пользователю необходимой информации;
- информационный фонд системы;
- способы и методы организации процессов обработки информации;
- комплекс программных средств, реализующих алгоритмы преобразования информации;
- комплекс технических средств, функционирующих в системе;
- персонал, обслуживающий систему.

Любая АИС функционирует в окружении внешней среды, являющейся для АИС источником входной и потребителем выходной информации. В пределах АИС, начиная со входа в систему и кончая выходом из нее, информационный поток проходит несколько этапов обработки.

С помощью АИС обеспечивается многовариантность расчетов, принимаются рациональные управленческие решения, в том числе в режиме

реального времени, организуется комплексный учет и экономический анализ, достигаются достоверность и оперативность получаемой и используемой в управлении информации и т.д.

Основная цель АИС - хранение, обеспечение эффективного поиска и передачи информации по соответствующим запросам для наиболее полного удовлетворения информационных запросов большого числа пользователей.

К основным принципам автоматизации информационных процессов относят: *окупаемость, надежность, гибкость, безопасность, дружелюбность, соответствие стандартам.*

Выделяются четыре типа АИС:

1. Охватывающий один процесс (*операцию*) в одной организации.
2. Объединяющий несколько процессов в одной организации.
3. Обеспечивающий функционирование одного процесса в масштабе нескольких взаимодействующих организаций.
4. Реализующий работу нескольких процессов или систем в масштабе нескольких организаций.

В АИС обычно применяются *автоматизированные рабочие места (АРМ)* на базе ПК, распределённые базы данных, программные средства, ориентированные на конечного пользователя.

Основное назначение автоматизированных информационных систем не просто *собрать* и *сохранить* электронные информационные ресурсы, но и обеспечить к ним доступ пользователей. Одной из важнейших особенностей АИС является организация поиска данных в их информационных массивах (базах данных).

Поэтому АИС практически являются *автоматизированными информационно-поисковыми системами (АИПС)* - программный продукт, предназначенный для реализации процессов *ввода, обработки, хранения, поиска, представления данных* т.п. АИПС бывают фактографическими и документальными.

1. *Фактографические АИПС* обычно используют табличные реляционные БД с фиксированной структурой данных (*записей*).

2. *Документальные АИПС* отличаются неопределённостью или переменной структурой данных (*документов*). Для их разработки обычно применяются оболочки АИС.

Способами обеспечения автоматизированных информационных систем и их технологий являются *программное, техническое, лингвистическое, организационное и правовое обеспечение*, используемые или создаваемые при проектировании информационных систем и обеспечивающие их эксплуатацию.

В работе предложены разнообразные по форме задания: вопросы и задания на закрепление теоретических знаний, задания в тестовой форме, практические работы, направленные на формирование практических умений и навыков.

Приступая к работе, следует ознакомиться с источниками, указанными в библиографическом списке. Работая с нормативными актами, рекомендуется использовать различные информационно-справочные системы. Для выполнения практических заданий необходимо подключение к сети *Internet*. Рабочая тетрадь подготовлена в формате **MS Word 2016** (.docx).

Рекомендуется последовательное выполнение заданий, поскольку они связаны логикой освоения учебного материала.

Задания для работы по теме: Информационные технологии

Задание № 1. Дайте определения следующим понятиям:

Понятие	Определение
<i>Информационные технологии</i>	
<i>Информационная система</i>	
<i>Программное обеспечение (ПО) информационных систем</i>	

Задание № 2. Установите соответствие программного обеспечения (*ПО*) и его вида: ПО общего назначения, операционные системы, сервисные программы, методо-ориентированное ПО, проблемно-ориентированное ПО,

ПО для глобальных сетей, программы технического обслуживания, инструментальное ПО, ПО для организации вычислительного процесса.

Базовое (системное) ПО	Прикладное ПО

Задание № 3. Установите соответствие программ. Оставшиеся пустые строки заполните самостоятельно. (*Windows 10, Adobe Photoshop, CorelDRAW, Notepad++, Camtasia Studio*).

Операционные системы	Прикладные ПО общего назначения	Проблемно-ориентированное прикладное ПО

Задание № 4. Перечислите принципы правового регулирования отношений в сфере информации, информационных технологий и защиты информации.

Задание № 5. Что предусматривает Государственное регулирование в сфере применения информационных технологий?

Задание № 6. Перечислите свойства ИТ

Свойства информационной технологии

- 1.
- 2.
- 3.
- 4.
- 5.

Задание № 7. Охарактеризуйте свойства ИТ

Задание № 8. Выделите основные классификационные признаки информационных технологий.

Классификация ИТ

- 1.
 - 2.
 - 3.
 - 4.
 - 5.
- и т.д.

Задания для работы по теме: Автоматизированные информационные системы

Задание № 9. Для каких целей создаются государственные информационные системы?

--

Задание № 10. Дайте определения следующим понятиям:

Понятие	Определение
<i>Информация</i>	
<i>Данные</i>	

Знания	
Информационная среда-	

Задание № 11. Какие требования предъявляют к информации:

Задание № 12. Назовите и охарактеризуйте информационные системы в зависимости от степени автоматизации

Степень автоматизации	Характеристика
1. ручные ИС	
2. автоматизированные ИС	
3. автоматические ИС	

Задание № 13. Выделите основные классификационные признаки информационных систем.



Задание № 14. На что необходимо обращать внимание при анализе качества информационных процессов в ИС?

--

Задание № 15. В чем заключается назначение компьютерных справочно-правовых систем? Перечислите достоинства и ограничения справочно-правовых систем.

--

Задание 16. Сделайте вывод по проделанной работе.



Контрольные вопросы и задания

1. Как различаются системы по характеру использования информации?
2. Как классифицируются системы в зависимости от степени автоматизации?
3. Что такое интегрированные информационные системы?
4. Назовите компоненты обеспечения автоматизированной информационной системы.
5. Что относится к правовому обеспечению АИС?
6. Что относится к организационному и информационному обеспечению АИС?
7. Перечислите основные функции систем.

49-§. РАБОТА С РАЗЛИЧНЫМИ СРЕДСТВАМИ ЗАЩИТЫ ИНФОРМАЦИИ

Цель работы: рассмотреть понятия: компьютерный вирус, антивирусная программа, архив, программа архивации; виды вирусов и антивирусных программ; меры профилактики ПК от компьютерных вирусов. Научиться работать с архивами, антивирусными программами.

Краткая теоретическая информация:

Информация (лат. *Informatio* - разъяснение, изложение), первоначально - сведения, передаваемые людьми устным, письменным или другим способом с помощью условных сигналов, технических средств и т.д.

На практике используют несколько групп методов защиты, в том числе:

- **препятствие на пути предполагаемого похитителя**, которое создают физическими и программными средствами;

- **управление**, или оказание воздействия на элементы защищаемой системы;
- **маскировка**, или преобразование данных, обычно *криптографическими* способами;
- **регламентация**, или разработка нормативно-правовых актов и набора мер, направленных на то, чтобы побудить пользователей, взаимодействующих с базами данных, к должному поведению;
- **принуждение**, или создание таких условий, при которых пользователь будет вынужден соблюдать правила обращения с данными;
- **побуждение**, или создание условий, которые мотивируют пользователей к должному поведению.

Каждый из методов защиты информации реализуется при помощи различных категорий средств. Основные средства - *организационные* и *технические*.

Организационные средства защиты информации

Разработка комплекса организационных средств защиты информации должна входить в компетенцию службы безопасности. Чаще всего специалисты по безопасности:

- **разрабатывают внутреннюю документацию**, которая устанавливает правила работы с компьютерной техникой и конфиденциальной информацией;
- **проводят инструктаж** и периодические проверки персонала; инициируют подписание дополнительных соглашений к трудовым договорам, где указана ответственность за разглашение или неправомерное использование сведений, ставших известными по работе;
- **разграничивают зоны ответственности**, чтобы исключить ситуации, когда массивы наиболее важных данных находятся в распоряжении одного из сотрудников; организуют работу в общих программах документооборота и следят, чтобы критически важные файлы не хранились вне сетевых дисков;

- **внедряют программные продукты**, которые защищают данные от копирования или уничтожения любым пользователем, в том числе топ-менеджментом организации;

- **составляют планы восстановления системы** на случай выхода из строя по любым причинам.

Технические средства защиты информации

Группа технических средств защиты информации совмещает *аппаратные* и *программные* средства. Основные:

- резервное копирование и удаленное хранение наиболее важных массивов данных в компьютерной системе - на регулярной основе;

- дублирование и резервирование всех подсистем сетей, которые имеют значение для сохранности данных;

- создание возможности перераспределять ресурсы сети в случаях нарушения работоспособности отдельных элементов;

- обеспечение возможности использовать резервные системы электропитания;

- обеспечение безопасности от пожара или повреждения оборудования водой;

- установка программного обеспечения, которое обеспечивает защиту баз данных и другой информации от несанкционированного доступа.

В комплекс технических мер входят и меры по обеспечению физической недоступности объектов компьютерных сетей, например, такие практические способы, как *оборудование помещения, камерами и сигнализацией*.

ЗАЩИТА ИНФОРМАЦИИ, АНТИВИРУСНАЯ ЗАЩИТА

Компьютерный вирус - программа способная самопроизвольно внедряться и внедрять свои копии в другие программы, файлы, системные области компьютера и в вычислительные сети, с целью создания всевозможных помех работе на компьютере.

Признаки заражения:

- ✓ прекращение работы или неправильная работа ранее функционировавших программ
- ✓ медленная работа компьютера
- ✓ невозможность загрузки ОС
- ✓ исчезновение файлов и каталогов или искажение их содержимого
- ✓ изменение размеров файлов и их времени модификации
- ✓ уменьшение размера оперативной памяти
- ✓ непредусмотренные сообщения, изображения и звуковые сигналы
- ✓ частые сбои и зависания компьютера и др.

Классификация компьютерных вирусов

По среде обитания:

- ✓ Сетевые – распространяются по различным компьютерным сетям
- ✓ Файловые – внедряются в исполняемые модули (COM, EXE)
- ✓ Загрузочные – внедряются в загрузочные сектора диска или сектора, содержащие программу загрузки диска
- ✓ Файлово-загрузочные - внедряются и в загрузочные сектора, и в исполняемые модули

По способу заражения:

- ✓ Резидентные – при заражении оставляет в оперативной памяти компьютера свою резидентную часть, которая потом перехватывает обращения ОС к объектам заражения
- ✓ Нерезидентные – не заражают оперативную память и активны ограниченное время

По воздействию:

- ✓ Неопасные – не мешают работе компьютера, но уменьшают объем свободной оперативной памяти и памяти на дисках
- ✓ Опасные – приводят к различным нарушениям в работе компьютера
- ✓ Очень опасные – могут приводить к потере программ, данных, стиранию информации в системных областях дисков

По особенностям алгоритма:

- ✓ Паразиты – изменяют содержимое файлов и секторов, легко обнаруживаются
- ✓ Черви – вычисляют адреса сетевых компьютеров и отправляют по ним свои копии
- ✓ Стелсы – перехватывают обращение ОС к пораженным файлам и секторам и подставляют вместо них чистые области
- ✓ Мутанты – содержат алгоритм шифровки-дешифровки, ни одна из копий не похожа на другую
- ✓ Трояны – не способны к самораспространению, но маскируясь под полезную, разрушают загрузочный сектор и файловую систему

Основные меры по защите от вирусов

- ✓ оснастите свой компьютер одной из современных антивирусных программ: Doctor Web, Norton Antivirus, AVP
- ✓ постоянно обновляйте антивирусные базы
- ✓ делайте архивные копии ценной для Вас информации (*гибкие диски, CD*)

Классификация антивирусного программного обеспечения

- ✓ **Сканеры (детекторы).** Принцип работы антивирусных сканеров основан на проверке файлов, секторов и системной памяти, и поиске в них известных и новых (*неизвестных сканеру*) вирусов.
- ✓ **Мониторы.** Это целый класс антивирусов, которые постоянно находятся в оперативной памяти компьютера и отслеживают все подозрительные действия, выполняемые другими программами. С помощью монитора можно остановить распространение вируса на самой ранней стадии.
- ✓ **Ревизоры.** Программы-ревизоры первоначально запоминают в специальных файлах образы главной загрузочной записи, загрузочных секторов логических дисков, информацию о структуре каталогов, иногда - объем установленной оперативной памяти. Программы-ревизоры первоначально запоминают в специальных файлах образы главной загрузочной записи, загрузочных секторов логических дисков, информацию о

структуре каталогов, иногда - объем установленной оперативной памяти. Для определения наличия вируса в системе программы-ревизоры проверяют созданные ими образы и производят сравнение с текущим состоянием.

Задание №1. Ответьте на вопросы:

Вопрос	Ответ
<i>Что такое компьютерный вирус?</i>	
<i>В чем состоит принцип работы вируса?</i>	
<i>Перечислите вредные действия вирусов.</i>	

Задание №2. Запишите признаки заражения ПК вирусом.

№	Признак
1	
2	
3	

Задание №3. Проанализируйте и запишите, какие типы файлов подвержены заражению?

Типы файлов, подверженные заражению	Типы файлов, не подверженные заражению

Задание №4. Проанализируйте и запишите основные способы заражения ПК.

№	Способ заражения ПК
1	
2	
3	

4	
---	--

Задание №5. Запишите меры профилактики заражения ПК вирусом:

№	Способ профилактики
1	
2	
3	
4	
5	
6	

Задание №6. Запишите классификацию вирусов в виде таблицы

№	Вид (название) вируса	Особенность вируса
1		
2		
3		

Задание №7 Сравните виды антивирусных программ, дайте им краткую характеристику.

№	Вид	Характеристика	Достоинства	Недостатки
1	<i>Антивирусы-сканеры</i>			
2	<i>Антивирусы-мониторы</i>			

Задание №8. Перечислите функции, выполняемые антивирусом Касперского.

№	Функция
1	

2	
3	
4	
5	
6	

Задание 9. Заполнить таблицу.

Описать 5 антивирусных программ. Наименование антивирусной программы	Характеристики	Условия использования (платно/бесплатно)
...

Задание 10. Сделайте вывод по проделанной работе.



Контрольные вопросы и задания

1. Вследствие каких причин возникает проблема угрозы информации?
2. Перечислите меры, которые могут быть направлены на защиту информации.
3. Назовите три направления, по которым может быть построена защита информации. Дайте им краткую характеристику.
4. В чем, по вашему мнению, заключается защита от аппаратных и программных сбоев?
5. Перечислите признаки надежных паролей и парольных фраз.
6. Каким образом можно организовать защиту информации от утечки?
7. Дайте определению компьютерному вирусу.
8. Какие симптомы заражения компьютера вирусами вы знаете?
9. Приведите классификацию вирусных программ.
10. Какова цель вредоносных программ?
11. Что означает понятие «фишинг»?

12. Какова цель троянских программ?

50-§. ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ КОМПЬЮТЕРНОЙ СЕТИ

Цель работы: *познакомиться с существующими угрозами безопасности в компьютерной сети и их преодолением.*

Краткая теоретическая информация:

Сетью называют группу компьютеров, соединенных между собой при помощи специального оборудования, которое, при установленных специальных программных средствах, обеспечивает *обмен информацией* в реальном режиме времени.

Компьютерная сеть (*вычислительная сеть*) - система, обеспечивающая обмен данными между вычислительными устройствами (*компьютеры, серверы, маршрутизаторы и другое оборудование*).

Для передачи информации могут быть использованы различные среды.

По территориальной распространённости

- **LAN** (*ЛВС, Local Area Network*) - локальные сети, имеющие замкнутую инфраструктуру до выхода на поставщиков услуг. Термин «LAN» может описывать и маленькую *офисную сеть*, и *сеть уровня большого учреждения*, занимающего несколько сотен гектаров (*10 км*) в радиусе. Локальные сети являются сетями закрытого типа, доступ к ним разрешён только ограниченному кругу пользователей, для которых работа в такой сети непосредственно связана с их профессиональной деятельностью.

- **CAN** (*Campus Area Network*) - кампусная сеть, объединяет локальные сети близко расположенных зданий.

- **MAN** (*Metropolitan Area Network*) - городские сети между учреждениями в пределах одного или нескольких городов, связывающие много локальных вычислительных сетей.

- **WAN** (*Wide Area Network*) - глобальная сеть, покрывающая большие географические регионы, включающие в себя как локальные сети, так и прочие телекоммуникационные сети, и устройства. Глобальные сети являются открытыми и ориентированы на обслуживание любых пользователей.

По архитектуре

- Клиент-сервер
- Одноранговая сеть (*децентрализованная или пиринговая сеть*)

По типу сетевой топологии

- Шина
- Решётка
- Кольцо
- Дерево
- Двойное кольцо
- Смешанная топология
- Звезда
- Fat Tree
- Ячеистая

По типу среды передачи

- Проводные (*телефонный провод, коаксиальный кабель, витая пара, волоконно-оптический кабель*)
- Беспроводные (*передача информации по радиоволнам в определённом частотном диапазоне*)

По скорости передачи

- низкоскоростные (*до 10 Мбит/с*),
- среднескоростные (*до 100 Мбит/с*),
- высокоскоростные (*свыше 100 Мбит/с*);

Особенности безопасности компьютерных сетей

Основной особенностью любой сетевой системы является то, что ее компоненты распределены в пространстве и связь между ними физически осуществляется при помощи сетевых соединений и программно при помощи механизма сообщений.

Сетевые системы характерны тем, что, наряду с обычными (*локальными*) атаками, осуществляемыми в пределах одной компьютерной системы, к ним применим специфический вид атак, обусловленный распределенностью

ресурсов и информации в пространстве. Это так называемые сетевые (или удаленные) атаки.

Средства обеспечения безопасности сети

- Компьютерная и сетевая безопасность.
- Брандмауэр
- Механизм виртуальных частных сетей
- Безопасная информационная система
- Конфиденциальность, доступность, целостность данных
- Сервисы сетевой безопасности (шифрование информации, аутентификацию, идентификация, авторизацию, аудит, технология защищенного канала)



(рис.5.50.1). Компьютерная сеть

В настоящее время одним из всемирно известных разработчиков и производителей сетевого и телекоммуникационного оборудования является компания **D-Link**. Она предлагает широкий набор решений для домашних пользователей, корпоративного сегмента, провайдеров интернет-услуг. Также компания предоставляет решения для учебных заведений.

Комплект состоит из одного коммутатора третьего уровня D-Link DES-3810-28, десяти управляемых коммутаторов второго уровня D-Link DES-3200-10. Для подключения к коммутаторам понадобятся консольные кабели и кабели **Ethernet**.

Задачи:

1. Для чего можно использовать команду ping?

2. Для чего используется сообщение *Redirect* протокола ICMP?
3. Какие сообщения протокола ICMP злоумышленник может использовать для своих целей?
4. Каким образом организуются распределенные DOS-атаки?
5. Как можно воспользоваться фрагментацией для проникновения пакета, нарушающего правила безопасности?
6. Как можно осуществить мониторинг сетевого трафика?
7. Опишите протокол ICMP.
8. Что из себя представляет IP-датаграммы?
9. Что из себя представляет политика безопасности?
10. Что из себя представляет политика сетевого подключения?
11. Что из себя представляет политика межсетевого экрана?

1. Протокол ICMP обеспечивает возможность управления протоком IP. Еще одна полезная функция, которая реализуется посредством ICMP, - возможность проверять доступность сети при помощи команды *ping*, использующей для этого сообщения *EchoRequest* и *EchoReply*.

2. *Redirect*. Этот тип сообщения создавался, чтобы маршрутизаторы информировали друг друга о существовании оптимального маршрута к месту назначения

3.

a. При помощи сообщения *Redirect* дается перенаправить сетевой трафик нашей сети через один или несколько узлов, контролируемых нарушителем, облегчая тем самым выполнение различных злонамеренных действий.

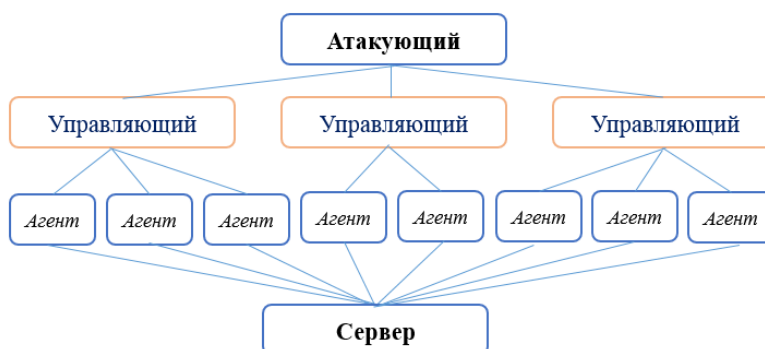
b. Посредством сообщения *DestinationUnreachable* маршрутизатор получает сведения о недостижимости назначения *IP-пакета*. Подделывая пакеты *DestinationUnreachable*, нарушитель может создать впечатление о недостижимости каких-либо важных узлов.

c. Утилита *ping* позволяет не только диагностировать сеть, но и получать информацию о ней. Каждый добытый злоумышленником бит

информации приближает его к цели - найти способ проникнуть в вашу сеть или нарушить ее работу.

4. **Распределенные DOS-атаки.** Организовать такую атаку достаточно сложно. Атакующий должен вначале проникнуть на другие - слабо защищенные - системы, чтобы создать ряд систем, которые называются управляющими и агентами. Сама атака на целевую систему выполняется агентами. Она контролируется управляющими системами. На вершине всей этой структуры находится компьютер атакующего.

В программе *Trin00* взаимодействие обеспечивается протоколом *TCP/IP*, а на компьютер жертвы направляется поток пакетов *UDP*. Новые инструменты устроены сложнее, при взаимодействии применяют шифрование и позволяют атакующему выбирать тип атаки: *UDP*, *SYN* или *ICMP flood* либо их комбинацию. Распределенная атака предоставляет агентам возможность перегружать канал связи целого узла.



(рис. 5.50.2). Один компьютер координирует атаки нескольких систем, сосредоточенных на одной цели

5. **Фрагментация пакетов.** При фрагментации сообщения заголовок TCP помещается в первый фрагмент. Номер первого фрагмента равен 0. Если такие пакеты проходят межсетевой экран, фильтр в состоянии проверить информацию в заголовке TCP (например, номер порта) только для первого фрагмента. Известный трюк, позволяющий пакету проникнуть через фильтр, состоит в том, что номер фрагмента просто устанавливается равным 1. Поскольку маршрутизатор считает, что это не первый фрагмент в

последовательности, заголовок ТСП не проверяется и пакет не блокируется, даже если он нарушает правила безопасности.



(рис. 5.50.3). Заголовок ТСП включается только в первый фрагмент

6. **Мониторинг сетевого трафика.** Для перехвата сетевого трафика в локальной сети бывает достаточно подключить компьютер к сети и воспользоваться сетевым адаптером в *promiscuousmode* (режим перехвата всех кадров). В Internet пакет может быть перехвачен в любом сегменте, по которому он проходит. Единственный способ гарантировать надежную защиту Internet-соединения – либо применить технологию шифрования, либо передавать пакеты обычных сетевых протоколов через закрытую частную сеть (VPN).

7. **Протокол межсетевых управляющих сообщений ICMP** (*Internet Control Message Protocol*) является обязательным стандартом **ТСП/IP**, описанным в документе RFC 792, «*Internet Control Message Protocol (ICMP)*». Используя ICMP, узлы и маршрутизаторы, связывающиеся по протоколу IP, могут сообщать об ошибках и обмениваться ограниченной управляющей информацией и сведениями о состоянии.

8. ICMP-сообщения инкапсулируются и передаются в IP-датаграммах, как показано на следующем рисунке.



9. **Политика безопасности** - это список того, что разрешено и запрещено делать на подключенном к сети компьютеру. В действительности хорошая политика охватывает множество различных тем и во всех деталях описывает разрешенные пользователям действия и штрафы за нарушение требований.

10. **Политикой сетевого подключения** должны быть определены типы устройств, разрешенные для подключения к сети. Например, позволяя подключение серверов и рабочих станций, можно запретить подключение к сети модемных серверов удаленного доступа. Аналогично в политике к сети должны быть детально определены настройки систем, которые допускается подключать к сети. Политика подключения к сети должна также описывать функции, для выполнения которых предназначены определенные компьютеры. Если в отделе может быть установлен *Web* - или *FTP-сервер*, полезно определить, в какой части сети его следует подключать.

11. **Межсетевой экран** использует набор правил, определяющих, какие пакеты (*в случае фильтра*) или сервисы (*в случае прокси-сервера*) будут работать через него. При этом выбирается одна из следующих двух стратегий:

- разрешить любой доступ, не запрещенный правилами;
- запретить любой доступ, не запрещенный правилами.

После выбора стратегии следует определить, каким сервисам будет разрешено работать через межсетевой экран и в каком направлении. Можно отобрать их на основе общей политики безопасности. Например, если нужно запретить загрузку программ из *Internet*, стоит заблокировать в межсетевом экране установку входящих соединений *FTP*. Аналогично можно ограничить работу с другими службами, такими как *Telnet* или *HTTP*.

Например, ваш список может состоять из таких правил:

- электронная почта может пересылаться в обоих направлениях, но должна проходить через защищенный *SMTP-сервер*;

- запрещена работа любых служб, требующих установки прямого соединения между внутренним клиентом и внешним сервером. Все разрешенные службы обязаны пользоваться прокси-сервером;

- запрещен доступ по протоколу FTP в обоих направлениях.



Контрольные вопросы и задания

1. Что такое компьютерная сеть?
2. На какие группы делятся компьютерные сети по территориальной распределённости?
3. Какую сеть называют глобальной?
4. На какие группы делятся компьютерные сети по ведомственной принадлежности?
5. На какие группы делятся компьютерные сети по скорости передачи информации?
6. Какова основная характеристика каналов передачи информации, и в каких единицах она измеряется?
7. Скорость передачи данных через ADSL-соединение равна 1024000 бит/с. Сколько минут займет передача файла размером 600000 Кбит через данное соединение?
8. Пользователь скачивает из сети файл размером 60 Мбайт. Скорость передачи первой половины данных составляет 256 Кбит в секунду, а второй – в два раза меньше. Сколько минут будет скачиваться файл?
9. На какие группы делятся компьютерные сети по типу среды передачи данных?
10. В ОС Windows: набор данных, определяющих, к каким папкам и файлам пользователь имеет доступ, какие изменения пользователем могут вноситься в работу системы, а также персональные настройки пользователя (*фон рабочего стола, экранная заставка и др.*) получили название...

СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ

ASCII	<i>American Standard Code for Informational Interchange</i> Американский стандартный код для информационного обмена
АИС	Автоматизированная информационная система
BIOS	<i>Basic Input-Output System</i> Базовая система ввода-вывода
USB	<i>Universal Serial Bus</i> Универсальная последовательная шина
АЛУ	Арифметико-логическое устройство
АСУ	Автоматизированная система управления
БД	База данных
ИКТ	Информационно-коммуникационные технологии
ИПС	Информационно-поисковая система
ИТ	Информационные технологии
ИС	Информационная система
МП	Машинный перевод
ОЗУ	Оперативное запоминающее устройство
ОС	Операционная система
ПЗУ	Постоянное запоминающее устройство
ПК	Персональный компьютер
СП	Системные программы
СУБД	Система управления базами данных
УУ	Устройство управления

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Постановление Президента Республики Узбекистан от 14 мая 2018 года № ПП-3724. «О мерах по ускоренному развитию электронной коммерции»
2. Закон Республики Узбекистан от 11.12.2003 г. N 560-П. Об информатизации.
3. Закон Республики Узбекистан от 11.12.2003 г. N 562-П. Об электронной цифровой подписи
4. Ш.М. Мирзиёев. «Мы все вместе построим свободное, демократическое и процветающее государство Узбекистан». *Выступление на торжественной церемонии вступления в должность Президента Республики Узбекистан на совместном заседании палат Олий Мажлиса*. Мирзиёев Ш.М. - Ташкент: Ўзбекистон, 2016. - 56 с.
5. Ш.М. Мирзиёев. "Критический анализ, жесткая дисциплина и персональная ответственность должны стать повседневной нормой в деятельности каждого руководителя". *В книге приведен доклад Президента Республики Узбекистан Шавката Мирзиёева на состоявшемся 14 января текущего года расширенном заседании Кабинета Министров, посвященном итогам социально-экономического развития страны в 2016 году и важнейшим приоритетным направлениям экономической программы на 2017 год*. Ташкент: Ўзбекистон, 2017. - 104 с.
6. Ш.М. Мирзиёев. Обеспечение верховенства закона и интересов человека – гарантия развития страны и благополучия народа. *Доклад Президента Республики Узбекистан Шавката Мирзиёева на торжественном собрании, посвященном 24-й годовщине принятия Конституции Республики Узбекистан*. Ташкент: Ўзбекистон, 2017. - 48 с.
7. Ш.М. Мирзиёев. Свободное, демократическое и процветающее государство Узбекистан мы построим вместе с нашим мужественным и благородным народом. *Выступление Шавката Мирзиёева на торжественной церемонии вступления в должность Президента Республики Узбекистан на*

совместном заседании палат Олий Мажлиса. Ташкент: Ўзбекистон, 2017. - 488 с.

8. Указ Президента Республики Узбекистан. О стратегии действий по дальнейшему развитию Республики Узбекистан. Президент Республики Узбекистан Ш. МИРЗИЁЕВ. г. Ташкент, 7 февраля 2017 г., № УП-4947.

9. Eric H. Glendinning, John McEwan. Oxford English for Information Technology (*English for IT*) - Student's Book. OXFORD University Press. www.oup.com/elt.

10. Eric H. Glendinning, John McEwan. Oxford English for Information Technology (*English for IT*) - Teacher's Book.

11. А.Н. Степанов Информатика: Учебник для вузов. 4-е изд. - СПб.: Питер, 2006. - 684 с.

12. Новые возможности системы Microsoft Office 2007/Мюррей К.; пер. с англ. - М.: Издательство «ЭКОМ Бином», 2007. – 48 с.

13. А.П. Курносов. Информатика: Учебное пособие - М.: «Колос С», 2005. – 272 с.

14. А.В. Улезько и другие. Информатика: Учебное пособие - Воронеж: ВГАУ, 2007. – 108 с.

15. О. Мединов Excel. Мультимедийный курс (+DVD) - Питер Пресс, 2009

16. В.А. Острейковский Информатика. Учебник. - М.: Высшая школа, 2000. – 508 с.

17. А.П. Курносов. Практикум по информатике. - Воронеж: ВГАУ, 2007. – 253 с.

18. А.П. Курносов. Практикум по информатике: Учебное пособие -М.: «Колос С», 2008. – 415 с.

19. Ю. Стоцкий и другие. Office 2007. Самоучитель Изучаем самостоятельно, - Питер Пресс, 2008

20. А. Трусов Excel 2007 для менеджеров и экономистов: логистические, производственные и оптимизационные расчеты (+CD) - Питер Пресс, 2009

21. Д. Ватолин, и другие. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. - М.: Диалог-мифы, 2002. – 384 с.
22. Н.Ю. Терехова Фрактальная графика: методические указания к лекциям и лабораторным работам по курсу «Основы графического дизайна» М: - МГТУ им. Н.Э. Баумана, 2006 г.- 24 с.
23. Ю.К. Пирогова, П.Б. Паршин Рекламный текст, семиотика и лингвистика. - М.: изд. Гребенникова 2000. 250 с.
24. В.С. Пташинский Самоучитель Office 2013/-М.: Эксмо, 2013. – 288 с.
25. Ю.И. Баратов, М.М. Антонов. Word 2010 с нуля! - М.Лучшие книги, 2011. – 224 с.
26. С.М. Окулов. Основы программирования. - 3-е изд. - М: БИНОМ. Лаборатория знаний, 2006.
27. Федеральный закон «Об электронной цифровой подписи» от 10 января 2002 года № 1-ФЗ
28. Электронная подпись и шифрование //МО ПНИЭИ
29. Э. Браудэ. Технология разработки программного обеспечения. СПб.: Питер, 2004.
30. Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул Технология разработки программного обеспечения. М.: ФОРУМ-Инфра-М, 2008.
31. Дж. Кериевски. Рефакторинг с использованием шаблонов. М.: ООО «И. Д. Вильямс», 2006.
32. Л. Константайн, Л. Локвуд. Разработка программного обеспечения. СПб.: Питер, 2004.
33. М.М. Котухов, А.Н. Кубанков, А.О. Калашников. Информационная безопасность. Москва Академия ИБС – МФТИ 2009
34. Вадим Гребенников. Управление информационной безопасностью. Стандарты СУИБ. «Издательские решения»
35. А.В. Артемов. Информационная безопасность: *(курс лекций)* *(Электронный ресурс)*: МАБИВ; Орел-2014.

36. Ю.Б. Бекаревич, Н.В. Пушкина. Самоучитель Microsoft Access 2013 СПб.:БХВ-Петербург, 2014. - 464 с
37. Н.Д. Берман, Т.А. Бочарова, Н.И. Шадрин. MS PowerPoint 2010: учебное пособие. Изд-во Тихоокеан. гос. ун-та, 2013. - 88 с.
38. Е.А. Докторов. Мультимедиа технологии: Конспект лекций. Часть 1/сост: - Ульяновск: УлГТУ, 2009. - 39 с.
39. Г.П. Катунин. Основы мультимедиа. Звук и видео. Монография. - Новосибирск, СибГУТИ, 2006. - 389 с.
40. Чепмен, Найджел. Цифровые технологии мультимедиа 2-е изд.– М.: Диалектика, 2005. - 624 с.
41. Андерсен, Бент Б. Мультимедиа в образовании – М.: Дрофа, 2007. - 224 с.
42. Е.М. Карчевский, И.Е. Филиппов. Word 2007 в примерах Учебно-методическое пособие Казанский федеральный университет 2010.
43. М.М. Владин. Книга с Видеокурсом!. Word 2010 с нуля!. Обложка, серия и оформление ООО «Лучшие книги», 2011-221 с.
44. В.В. Воробьев. Microsoft Word 2000: Пособие для начинающих. - К.: 2000. -35 с.
45. В.В. Воробьев. Microsoft Excel 2000: Пособие для начинающих. - К.: 2000. -36 с.
46. М.С. Кукушкина, О.А. Волкова. Работа в MS Office 2007. Текстовый процессор Word 2007. - Ульяновск : УлГТУ, 2010. – 76 с.
47. А.А. Гладкий, А.А. Чиртик. Excel-2007. Трюки и эффекты. -СПБ: Питер, 2007. – 368 с.
48. С.В. Плотникова, В.Г. Тихомиров. Электронные методы обработки информации. Часть II. Тамбов. Издательство ТГТУ. 2004.-27 с.
49. А. Лебедев. Понятный самоучитель Excel 2013. Питер, 2014. - 128 с.
50. Л.И. Галкин. Программирование для Excel. Создание макросов и процедур. Москва 2015. – 75 с.

51. И.Г. Семакин, А.П. Шестаков. Основы алгоритмизации и программирования. М.:Издательский центр “Академия”-2013. – 144 с.
52. Д.М. Златопольский. Сборник задач по программированию. - 3-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2011. - 304 с.
53. Д. Душистов. Решение 50 типовых задач по программированию на языке Pascal. БХВ-Петербург, 2011. - 304 с.
54. О.М. Макарова. Основы работы в CorelDRAW. Е - 2010. – 49 с.
55. И.Г. Захарова. Информатика. Введение в интернет: Учебное пособие. Тюмень. Изд-во ТГУ, 2005. 360 с.
56. Эндрю Блам. Сеть. Как устроен и как работает Интернет. Издание на русском языке AST Publishers,2014. – 136 с.
57. Е.В. Мальчук. HTML и CSS. Самоучитель. - М.: Издательский дом “Вильямс”, 2008. - 416 с.
58. И. Квинт. HTML, XHTML и CSS на 100%. СПб: Питер -2010.–384 с.
59. Л.М. Набиулина, и другие. Лабораторные работы по предмету «Информатика и информационные технологии». Ташкент – 2014. – 76 с.
60. С.В. Симонович. Информатика. Базовый курс/. СПб: Питер, 2000.
61. Ю.А. Шафрин Азбука компьютерных технологий. М.: Изд-во института психотерапии, 2000.
62. В. Леонтьев, Д. Турецкий. Новейшая энциклопедия программ. М, ОЛМА- ПРЕСС, 2002.
63. В.П. Леонтьев. Новейшая энциклопедия персонального компьютера. М: ОЛМА- ПРЕСС, 2003.

ИНТЕРНЕТ-РЕСУРСЫ

1. www.pdfactory.com
2. <http://www.indezone.com/products/powerpoint/learn/index.html>
3. <http://www.e-teaching.ru/po/Pages/reviewpowpoint2007.aspx>
4. <http://news.office-watch.com/t/n.aspx?a=903>

5. <http://www.youtube.com/watch?v=TR3gcCrTSzw>
6. <http://get.adobe.com/flashplayer/>.
7. http://www.ispringsolutions.com/products/ispring_free.html
8. www.pdfactory.com
9. [http://yandex.ru/\(http://ya.ru/\)](http://yandex.ru/(http://ya.ru/))
10. <http://www.yahoo.com/>
11. <http://www.rambler.ru/>
12. <http://www.punto.ru/>
13. <http://www.google.ru/> (<http://www.google.com/>)
14. <http://search.tut.by/>
15. <http://www.akavita.by/>

ПРИЛОЖЕНИЯ

Приложение 1

ASCII

ASCII – AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE.

ASCII была разработана (1963 год) для кодирования символов, коды которых помещались в 7 бит (128 символов). Со временем кодировка была расширена до 8-ми бит (256 символов), коды первых 128-и символов не изменились.

Печатные символы ASCII (код символа 32-127)

Буквы, цифры, знаки препинания и другие символы, расположенные на клавиатуре (англ.).

DEC	OCT	HEX	BIN	Symbol	HTML Number	HTML Name	Description
32	040	0x20	00100000		& #32;		Space
33	041	0x21	00100001	!	& #33;		Exclamation mark
34	042	0x22	00100010	"	& #34;	& quot;	Double quotes (or speech marks)
35	043	0x23	00100011	#	& #35;		Number
36	044	0x24	00100100	\$	& #36;		Dollar
37	045	0x25	00100101	%	& #37;		Procenttecken
38	046	0x26	00100110	&	& #38;	& amp;	Ampersand
39	047	0x27	00100111	'	& #39;		Single quote
40	050	0x28	00101000	(& #40;		Open parenthesis (or open bracket)
41	051	0x29	00101001)	& #41;		Close parenthesis (or close bracket)
42	052	0x2A	00101010	*	& #42;		Asterisk
43	053	0x2B	00101011	+	& #43;		Plus
44	054	0x2C	00101100	,	& #44;		Comma
45	055	0x2D	00101101	-	& #45;		Hyphen
46	056	0x2E	00101110	.	& #46;		Period, dot or full stop

47	057	0x2F	00101111	/	& #47;		Slash or divide
48	060	0x30	00110000	0	& #48;		Zero
49	061	0x31	00110001	1	& #49;		One
50	062	0x32	00110010	2	& #50;		Two
51	063	0x33	00110011	3	& #51;		Three
52	064	0x34	00110100	4	& #52;		Four
53	065	0x35	00110101	5	& #53;		Five
54	066	0x36	00110110	6	& #54;		Six
55	067	0x37	00110111	7	& #55;		Seven
56	070	0x38	00111000	8	& #56;		Eight
57	071	0x39	00111001	9	& #57;		Nine
58	072	0x3A	00111010	:	& #58;		Colon
59	073	0x3B	00111011	;	& #59;		Semicolon
60	074	0x3C	00111100	<	& #60;	& lt;	Less than (<i>or open angled bracket</i>)
61	075	0x3D	00111101	=	& #61;		Equals
62	076	0x3E	00111110	>	& #62;	& gt;	Greater than (<i>or close angled bracket</i>)
63	077	0x3F	00111111	?	& #63;		Question mark
64	100	0x40	01000000	@	& #64;		At symbol
65	101	0x41	01000001	A	& #65;		A
66	102	0x42	01000010	B	& #66;		B
67	103	0x43	01000011	C	& #67;		C
68	104	0x44	01000100	D	& #68;		D
69	105	0x45	01000101	E	& #69;		E
70	106	0x46	01000110	F	& #70;		F
71	107	0x47	01000111	G	& #71;		G
72	110	0x48	01001000	H	& #72;		H
73	111	0x49	01001001	I	& #73;		I
74	112	0x4A	01001010	J	& #74;		J
75	113	0x4B	01001011	K	& #75;		K
76	114	0x4C	01001100	L	& #76;		L
77	115	0x4D	01001101	M	& #77;		M

78	116	0x4E	01001110	N	& #78;		N
79	117	0x4F	01001111	O	& #79;		O
80	120	0x50	01010000	P	& #80;		P
81	121	0x51	01010001	Q	& #81;		Q
82	122	0x52	01010010	R	& #82;		R
83	123	0x53	01010011	S	& #83;		S
84	124	0x54	01010100	T	& #84;		T
85	125	0x55	01010101	U	& #85;		U
86	126	0x56	01010110	V	& #86;		V
87	127	0x57	01010111	W	& #87;		W
88	130	0x58	01011000	X	& #88;		X
89	131	0x59	01011001	Y	& #89;		Y
90	132	0x5A	01011010	Z	& #90;		Z
91	133	0x5B	01011011	[& #91;		Opening bracket
92	134	0x5C	01011100	\	& #92;		Backslash
93	135	0x5D	01011101]	& #93;		Closing bracket
94	136	0x5E	01011110	^	& #94;		Caret - circumflex
95	137	0x5F	01011111	_	& #95;		Underscore
96	140	0x60	01100000	`	& #96;		Grave accent
97	141	0x61	01100001	a	& #97;		a
98	142	0x62	01100010	b	& #98;		b
99	143	0x63	01100011	c	& #99;		c
100	144	0x64	01100100	d	& #100;		d
101	145	0x65	01100101	e	& #101;		e
102	146	0x66	01100110	f	& #102;		f
103	147	0x67	01100111	g	& #103;		g
104	150	0x68	01101000	h	& #104;		h
105	151	0x69	01101001	i	& #105;		i
106	152	0x6A	01101010	j	& #106;		j
107	153	0x6B	01101011	k	& #107;		k
108	154	0x6C	01101100	l	& #108;		l
109	155	0x6D	01101101	m	& #109;		m

110	156	0x6E	01101110	n	& #110;		n
111	157	0x6F	01101111	o	& #111;		o
112	160	0x70	01110000	p	& #112;		p
113	161	0x71	01110001	q	& #113;		q
114	162	0x72	01110010	r	& #114;		r
115	163	0x73	01110011	s	& #115;		s
116	164	0x74	01110100	t	& #116;		t
117	165	0x75	01110101	u	& #117;		u
118	166	0x76	01110110	v	& #118;		v
119	167	0x77	01110111	w	& #119;		w
120	170	0x78	01111000	x	& #120;		x
121	171	0x79	01111001	y	& #121;		y
122	172	0x7A	01111010	z	& #122;		z
123	173	0x7B	01111011	{	& #123;		Opening brace
124	174	0x7C	01111100		& #124;		Vertical bar
125	175	0x7D	01111101	}	& #125;		Closing brace
126	176	0x7E	01111110	~	& #126;		Equivalency sign - tilde
127	177	0x7F	01111111		& #127;		Delete
DEC	OCT	HEX	BIN	Symbol	HTML Number	HTML Name	Description

Расширенные символы ASCII Win-1251 кириллица (код символа 128-255)

DEC	OCT	HEX	BIN	Symbol
192	300	0xC0	11000000	А
193	301	0xC1	11000001	Б
194	302	0xC2	11000010	В
195	303	0xC3	11000011	Г
196	304	0xC4	11000100	Д
197	305	0xC5	11000101	Е
198	306	0xC6	11000110	Ж
199	307	0xC7	11000111	З
200	310	0xC8	11001000	И

201	311	0xC9	11001001	Й
202	312	0xCA	11001010	К
203	313	0xCB	11001011	Л
204	314	0xCC	11001100	М
205	315	0xCD	11001101	Н
206	316	0xCE	11001110	О
207	317	0xCF	11001111	П
208	320	0xD0	11010000	Р
209	321	0xD1	11010001	С
210	322	0xD2	11010010	Т

211	323	0xD3	11010011	У
212	324	0xD4	11010100	Ф
213	325	0xD5	11010101	Х
214	326	0xD6	11010110	Ц
215	327	0xD7	11010111	Ч
216	330	0xD8	11011000	Ш
217	331	0xD9	11011001	Щ
218	332	0xDA	11011010	Ъ
219	333	0xDB	11011011	Ы
220	334	0xDC	11011100	Ь
221	335	0xDD	11011101	Э
222	336	0xDE	11011110	Ю
223	337	0xDF	11011111	Я
224	340	0xE0	11100000	а
225	341	0xE1	11100001	б
226	342	0xE2	11100010	в
227	343	0xE3	11100011	г
228	344	0xE4	11100100	д
229	345	0xE5	11100101	е
230	346	0xE6	11100110	ж
231	347	0xE7	11100111	з
232	350	0xE8	11101000	и
233	351	0xE9	11101001	й

234	352	0xEA	11101010	к
235	353	0xEB	11101011	л
236	354	0xEC	11101100	м
237	355	0xED	11101101	н
238	356	0xEE	11101110	о
239	357	0xEF	11101111	п
240	360	0xF0	11110000	р
241	361	0xF1	11110001	с
242	362	0xF2	11110010	т
243	363	0xF3	11110011	у
244	364	0xF4	11110100	ф
245	365	0xF5	11110101	х
246	366	0xF6	11110110	ц
247	367	0xF7	11110111	ч
248	370	0xF8	11111000	ш
249	371	0xF9	11111001	щ
250	372	0xFA	11111010	ъ
251	373	0xFB	11111011	ы
252	374	0xFC	11111100	ь
253	375	0xFD	11111101	э
254	376	0xFE	11111110	ю
255	377	0xFF	11111111	я
DEC	OCT	HEX	BIN	Symbo

Приложение 2

Запись чисел в римской системе счисления

1	2	3	4	5
I	II	III	IV	V
6	7	8	9	10
VI	VII	VIII	IX	X
11	13	18	19	22
XI	XIII	XVIII	XIX	XXII

34	39	40	60	99
XXXIV	XXXIX	XL	LX	XCIX
200	438	649	999	1207
CC	CDXXXVIII	DCXLIX	CMXCIX	MCCVII
2045	3555	3678	3900	3999
MMXLV	MMM DLV	MMM DCLXXVIII	MMM CCM	MMM CMCXCIX

Приложение 3

Соответствие чисел, записанных в различных системах счисления

Десятичная	Двоичная	Восьмеричная	Шестнадцатеричная
1	001	1	1
2	010	2	2
3	011	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

Правила перевода чисел из одной системы счисления в другую

Для перевода двоичного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 2, и вычислить по правилам десятичной арифметики:

$$X_2 = A_n 2^{n-1} + A_{n-1} 2^{n-2} + A_{n-2} 2^{n-3} + \dots + A_2 2^1 + A_1 2^0$$

Приложение 4

Таблица с основными операторами языка программирования Pascal

Имя программы	Program
Начало и Конец	Begin и End.
Переменные	VAR
Целое число	Integer;
Вещественное число	Real;
Вывод на экран	Write('x = ', x);
Вывод на экран с новой строки	Writeln('x = ', x);
Модуль числа	Abs(x);
Возведение в квадрат	Sqr(x);
Квадратный корень числа	Sqrt(x);
Синус числа	Sin(x);
Косинус числа	Cos(x);
Арктангенс числа	Arctan(x);
Логарифм числа	Ln(x);
Возведение числа E в степень X	Exp(x);
Вычисление числа Пи	Pi;
Отбрасывает дробную часть числа	Trunc(x);
Округление дробного числа до ближайшего целого	Round(x);
Вызов библиотечного модуля (<i>после имени программы</i>)	USES CRT;
Ввод с клавиатуры	Read(x);
Ввод с клавиатуры с новой строки	ReadLn(x);
Раздел описания констант (<i>перед Var</i>)	Const
Инициализация датчика случайных чисел	Randomize;
Датчик случайных чисел, X max возможное число	Random(X);
Очистка экрана	ClrScr
Установка цвета экрана	TextBackGround
Переменные для хранения символов. Символьный порядковый тип	Char;
Логический тип переменных	Boolean;
Истина	True

Ложь	False
Условный оператор «Если»	IF
Условный оператор «То»	THEN
Условный оператор «Иначе»	ELSE

ГЛОССАРИЙ

Название термина на русском языке	The terminology of the English language	Объяснение термина
Алгоритм	<i>Algorithm</i>	Совокупность четко сформулированных правил, определяющих последовательность решения тех или иных задач за конечное число шагов.
Аппаратные средства	<i>Hardware</i>	Электронное и механическое оборудование в компьютерной технике (<i>в отличие от компьютерных программ</i>).
АРМ	<i>Automated workplace</i>	Автоматизированное рабочее место, оснащенное комплексом устройств, позволяющих автоматизировать часть выполняемых работником производственных операций. Основу любого АРМ составляет компьютер.
Алгоритм шифрования	<i>Encryption algorithm</i>	Набор математических правил, определяющих содержание и последовательность операций, зависящих от ключевой переменной (ключ шифрования), по преобразованию исходной формы представления информации (открытый текст) к виду, обладающему секретом обратного преобразования (<i>зашифрованный текст</i>).
Анализ риска	<i>Risk analysis</i>	Процесс изучения характеристик и слабых сторон системы, проводимый с использованием вероятностных расчетов, с целью определения ожидаемого ущерба в случае возникновения неблагоприятных событий. Задача анализа риска состоит в определении степени приемлемости того или иного риска в работе системы.
Антивирус	<i>Antivirus</i>	Программа, обнаруживающая или обнаруживающая и удаляющая вирусы. Если вирус удалить не удается, то зараженная программа уничтожается.
Аппаратная защита	<i>Hardware security</i>	Использование аппаратных средств, например, регистров границ или замков и ключей для защиты данных в ЭВМ
Аппаратное средство защиты информации	<i>Hardware means of information security</i>	Специальное защитное устройство или приспособление, входящее в комплект технического средства обработки информации.
Асимметричный шифр	<i>Asymmetric cipher</i>	Шифр, в котором ключ шифрования не совпадает с ключом дешифрирования.
Атака	<i>Attack</i>	Нарушение безопасности информационной системы, позволяющее захватчику управлять операционной средой.
Аутентификация	<i>Authenticate</i>	Проверка идентификации пользователя (проверка подлинности), устройства или другого компонента в системе, обычно для принятия решения о

		разрешении доступа к ресурсам системы; проверка целостности хранящихся или передающихся данных для обнаружения их несанкционированной модификации.
Автоматизация	<i>Automation</i>	Процесс использования автоматических устройств для управления какими-либо процессами или выполнения каких-либо действий
Автоматическая система	<i>Automatic system</i>	Совокупность управляемого объекта и автоматических управляющих устройств, функционирующая самостоятельно, без участия человека.
Автоматизированная система	<i>Automated system</i>	Комплекс технических, программных и других средств и персонала, предназначенный для автоматизации различных процессов (<i>не может функционировать без участия человека</i>).
Адаптер	<i>Adapter</i>	Устройство для согласования параметров входных и выходных сигналов других устройств с целью их сопряжения.
Алгоритмизация	<i>Algorithmization</i>	Составление алгоритмов для решения поставленных задач.
Алгоритмический язык	<i>Algorithmic language</i>	Язык описания алгоритмов.
Антивирусная программа	<i>Antivirus program</i>	Программа для защиты компьютера от компьютерных вирусов.
Аппаратура	<i>Equipment</i>	технические средства, физическое оборудование
Архив	<i>Archive</i>	Хранилище данных во внешней памяти.
Архивирование	<i>Archiving</i>	Процесс сжатия файлов с целью хранения их в более компактном виде.
Архитектура ЭВМ	<i>Computer architecture</i>	Совокупность общих принципов организации аппаратно - программных средств и их характеристик, определяющая функциональные возможности ЭВМ при решении соответствующих классов задач.
Атрибуты файла	<i>File attribute</i>	Дополнительные параметры, определяющие свойства файла.
База данных	<i>Data base</i>	Совокупность больших объемов (<i>массивов</i>) информации, хранение и обработка которых осуществляются с помощью компьютерной техники.
База знаний	<i>Knowledge base</i>	Совокупность знаний по данной предметной области, почерпнутых из публикаций, а также введенных в процессе взаимодействий эксперта (<i>экспертов</i>) с экспертной системой.
Бит	<i>Bit</i>	Минимальная единица информации, принимающая значение 0 или 1.
Байт	<i>Byte</i>	Наименьший элемент компьютерной памяти объемом 8 бит.
Блок-схема	<i>Block diagram</i>	Графическое описание алгоритма, выполняемое с помощью геометрических фигур.
Браузер	<i>Web browser</i>	Компьютерная программа, предназначенная для работы с ресурсами сети Интернет (<i>WWW-всемирной расширяемой сетью</i>). Средство навигации и просмотра Интернет-ресурсов.

Буфер	<i>Buffer</i>	Рабочая область оперативной памяти компьютера, используемая для кратковременного хранения информации.
Быстродействие компьютера (производительность)	<i>Computer performance (performance)</i>	Количество элементарных операций типа сложения, вычитания, пересылки и т.п., выполняемых за 1 с.
Видеоконтроллер	<i>Video controller</i>	Устройство, формирующее изображения в видеопамяти.
Винчестер (HDD)	<i>Winchester</i>	Несъемный жёсткий магнитный диск (<i>пакет дисков</i>).
Виртуальная реальность	<i>Virtual reality</i>	Искусственный мир, созданный программистами.
Вирус	<i>Virus</i>	Программа или фрагмент программ, причиняющий вред компьютеру и данным. «Прицепившись» к другим программам или файлам, они искажают информацию, удаляют файлы и т.п.
Внешняя память (ВЗУ)	<i>External memory</i>	Предназначена для размещения больших объемов информации и обмена ею с оперативной памятью.
Внутренняя память	<i>Internal memory</i>	Часть памяти компьютера, неотъемлемая от машины.
Входные данные	<i>Input</i>	Данные, вводимые в ЭВМ.
Выходные данные	<i>Output</i>	Данные, выводимые из ЭВМ.
Вычислительная машина	<i>Computer</i>	Устройство для организации автоматических вычислений и обработки информации.
Вычислительная сеть	<i>Area network</i>	Совокупность ЭВМ, объединенных сетью связи, позволяющей компьютерам проводить обмен информацией.
Вычислительная система	<i>Computing system</i>	Объединение вычислительных машин для организации хранения, обработки и поиска информации.
Вычислительная техника	<i>Computer facilities</i>	Совокупность устройств, предназначенных для автоматической или автоматизированной обработки данных.
Гиперссылка	<i>Hyperlink</i>	Текст или графическое изображение на сайте или в письме электронной почты, указывающий на другой файл, который может быть расположен в Интернете, и позволяющий перейти к этому файлу.
Гипертекст	<i>Hypertext</i>	Так называется система просмотра текстовой, графической и другой информации на экране монитора при помощи гиперссылок, которые связывают друг с другом страницы гипертекстового документа.
Графический редактор	<i>Graphic editor</i>	Программа, предназначенная для автоматизации процессов построения на экране дисплея графических изображений. Предоставляет возможности рисования линий, кривых, раскраски областей экрана, создания надписей различными шрифтами и т.д.

Документ	<i>Document</i>	Материальный объект с зафиксированной на нем информацией в виде текста, звукозаписи или изображения, предназначенный для передачи во времени и пространстве в целях хранения и общественного использования.
Двоичное кодирование	<i>Binary encoding</i>	Система представления данных последовательностью двух знаков: 1 и 0.
Диаграмма	<i>Chart</i>	Графическое представление данных. Они используются для анализа и сравнения данных, представления их в наглядном виде.
Дисплей	<i>Display</i>	Устройство отображения информации на электронном экране.
Дистанционное обучение	<i>Distance learning</i>	обучение на расстоянии с использованием учебников, персональных компьютеров и сетей ЭВМ.
Драйвер	<i>Driver</i>	программа, управляющая каким—либо модулем компьютера или периферийным устройством (<i>мышью, принтером, памятью и т.д.</i>).
Информация	<i>Information</i>	Сведения или данные, объективно отражающие различные стороны и элементы окружающего мира и деятельности человека на определенном этапе развития общества, представляющие для него какой-либо интерес и материализованные в форме, удобной для использования, передачи, хранения или обработки (<i>преобразования</i>) человеком или автоматизированными средствами.
Информатизация	<i>Informatization</i>	Организационный социально-экономический и научно-технический процесс создания оптимальных условий для удовлетворения информационных потребностей и реализации прав граждан, органов государственной власти, органов самоуправления, организаций, общественных объединений на основе формирования и использования информационных ресурсов.
Идентификация	<i>Identification</i>	Присвоение субъектам или объектам доступа идентификатора и (<i>или</i>) сравнение предъявляемого идентификатора с перечнем присвоенных идентификаторов.
Интерфейс	<i>Interface</i>	Граница раздела двух систем, устройств или программ; совокупность средств и правил, обеспечивающих взаимодействие устройств, программ.
Интеллект	<i>Intelligence</i>	Мыслительные способности человека.
Интеллектуальная собственность	<i>Intellectual property</i>	Исключительные права на результаты интеллектуальной деятельности
Информационная сфера	<i>Information sphere</i>	Совокупность информации, информационной инфраструктуры, субъектов, осуществляющих сбор, формирование, распространение и использование информации, а также системы регулирования возникающих при этом общественных отношений

Интегрированная система	<i>Integrated system</i>	Автоматизированная система, обеспечивающая различные потребности (в том числе информационные, вычислительные и др.) пользователей и поддерживающая единый порядок взаимодействия с пользователями, включая и способы представления данных.
Искусственный интеллект	<i>Artificial intelligence</i>	Свойство автоматических и автоматизированных систем выполнять отдельные функции человеческого интеллекта, т. е. выбирать и принимать оптимальные решения на основе ранее полученного опыта и рационального анализа внешних условий.
Информационная база	<i>Information base</i>	Совокупность информационных массивов, организованных соответствующим образом и размещенных на машинных носителях.
Информационная технология	<i>Information technology</i>	Комплекс методов, способов и средств, обеспечивающих сбор, накопление, хранение, поиск, обработку, анализ, выдачу данных, информации
Информационное обеспечение	<i>Information support</i>	Совокупность процессов по подготовке и предоставлению специально подготовленной информации для решения управленческих, научных, технических, производственных, коммерческих и других задач в соответствии с этапами их решения.
Информационные ресурсы	<i>Information resources</i>	Отдельные документы и отдельные массивы документов, документы и массивы документов в информационных системах (библиотеках, архивах, фондах, банках данных и др.).
Информационная потребность	<i>Information demand</i>	Состояние отдельного лица, коллектива или системы, характеризующееся необходимостью получения информации для успешного достижения каких-либо целей или выполнения работ.
Информационная система	<i>Information system</i>	Организационно упорядоченная совокупность документов (массивов документов) и информационных технологий, в том числе с использованием средств вычислительной техники и связи, реализующих информационные процессы.
Информационные услуги	<i>Information service</i>	Действия субъектов (собственников и владельцев) по обеспечению пользователей информационными продуктами.
Интернет	<i>Internet</i>	Провайдер - фирма, которая обеспечивает доступ в Интернет. Провайдер обычно не предоставляет клиентам никаких собственных материалов, а имеет лишь свою страницу во всемирной паутине.
Инtranet	<i>Intranet</i>	Сеть - корпоративная локальная сеть, создаваемая фирмой для того, чтобы обеспечить связь между ее сотрудниками на рабочих местах и различными филиалами этой фирмы.
Инструментальное ПО	<i>Tool SOFTWARE</i>	Предназначено для использования в ходе проектирования, разработки и сопровождения программ.
Интерпретатор	<i>Interpreter</i>	Система интерпретации (выполнения) программ на ЭВМ.

Каталог (<i>папка</i>)	<i>Directory (folder)</i>	Определенное место на диске (<i>в области данных диска</i>), где содержится информация о файлах и подкаталогах, привязанных к данному каталогу.
Клавиатура	<i>Keyboard</i>	Набор клавиш для ручного ввода данных.
Клавиша	<i>Key</i>	Элемент ручного управления, срабатывающий от нажатия.
Код	<i>Code</i>	Набор знаков в совокупности со схемой кодирования для представления информации в виде данных.
Кодирование данных	<i>Data encryption</i>	Выражение данных одного типа через данные другого типа. Количеством информации - числовая характеристика сигнала, отражающую ту степень неопределенности (<i>неполноту знаний</i>), которая исчезает после получения сообщения в виде данного сигнала. Эту меру неопределенности в теории информации называют энтропией.
Компилятор	<i>Compiler</i>	Система трансляции программ с одного языка на другой.
Консоль	<i>Console</i>	Средство для контроля за работой вычислительной системы и для управления ею вручную.
Контроллер	<i>Controller</i>	Устройство управления внешними устройствами.
Концентратор	<i>Concentrator</i>	Устройство повторяют всю информацию, которую они получают, то есть все устройства, подключенные к портам концентратора, получают одну и ту же информацию.
Курсор	<i>Cursor</i>	Значок для указания места на экране ЭВМ.
Легенда	<i>Legend</i>	Область, позволяющая сопоставить каждому из рядов данных на диаграмме цвет и стиль оформления.
Логические функции	<i>Logic function</i>	Функции, принимающие логические значения «истина» или «ложь».

СОДЕРЖАНИЕ

ГЛАВА 1. СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИХ ПРИМЕНЕНИЕ.		
МОДУЛЬ 1		
	Введение	4
1-§.	Информационная технология <i>предмет, цель и задачи</i> . Понятие информации, ее виды и свойства.	8
2-§.	Архитектура современного персонального компьютера (<i>ПК</i>).	18
3-§.	Операционные системы (<i>ОС</i>). Виды операционных систем.	29
4-§.	<i>Практическое занятие №1</i> . Сервисное программное обеспечение, утилиты.	37
5-§.	<i>Практическое занятие №2</i> . Подключения устройств к системному блоку.	41
6-§.	<i>Лабораторное занятие №1</i> . Тестирование компьютера (<i>процессор, видеокарта, HDD, RAM</i>).	51
7-§.	<i>Лабораторное занятие №2</i> . Настройка компьютера с помощью <i>BIOS Setup</i> .	59
8-§.	<i>Лабораторное занятие №3</i> . Программы-оболочки и архиваторы.	63
ГЛАВА 2. СОВРЕМЕННЫЕ ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ		
МОДУЛЬ 2		
9-§.	Алгоритм и его свойства.	72
10-§.	Язык программирования Pascal. Операторы Pascal.	79
11-§.	<i>Практическое занятие №3</i> . Среда программирования Pascal ABC.	92
12-§.	<i>Практическое занятие №4</i> . Операторы языка Pascal.	98
13-§.	<i>Практическое занятие №5</i> . Графические возможности языка PascalABC.	108
14-§.	<i>Лабораторное занятие №4</i> . Ввод и отладка линейной программы.	113
15-§.	<i>Лабораторное занятие №5</i> . Ввод и отладка программ с ветвлением.	118
16-§.	<i>Лабораторное занятие №6</i> . Составление программ с использованием циклов.	123
ГЛАВА 3. ИСПОЛЬЗОВАНИЕ ПРИКЛАДНЫХ ПРОГРАММ.		
МОДУЛЬ 3		

17-§.	Редакторы электронных таблиц. Их возможности.	128
18-§.	Основные понятия мультимедиа. Основы работы с аудио и видеоинформацией.	149
19-§.	Компьютерная графика. Виды компьютерной графики.	163
20-§.	Графический редактор <i>CorelDraw</i> и его возможности.	175
21-§.	Создание и управление базой данных (БД).	184
22-§.	Программы переводчики.	193
23-§.	<i>Практическое занятие №6.</i> Автоматизация редактирования и форматирования текста.	204
24-§.	<i>Практическое занятие №7.</i> Основы работы в <i>Microsoft Excel</i> .	210
25-§.	<i>Практическое занятие №8.</i> Создание презентаций в <i>Microsoft PowerPoint</i> .	217
26-§.	<i>Практическое занятие №9.</i> Основные понятия и возможности компьютерной графики.	223
27-§.	<i>Практическое занятие №10.</i> Порядок следования, объединение и группировка объектов в программе <i>CorelDraw</i> .	229
28-§.	<i>Практическое занятие №11.</i> Создание и управление базой данных.	233
29-§.	<i>Практическое занятие №12.</i> Создание формы базы данных в среде <i>MS. Access</i>	242
30-§.	<i>Практическое занятие №13.</i> Создание запросов и отчетов в СУБД <i>MS. Access</i> .	248
31-§.	<i>Лабораторное занятие №7.</i> Редактирование и форматирование документов в текстовом редакторе <i>MS Word</i> .	255
32-§.	<i>Лабораторное занятие №8.</i> Технология обработки данных в электронных таблицах.	261
33-§.	<i>Лабораторное занятие №9.</i> Создание обучающих презентаций.	268
34-§.	<i>Лабораторное занятие №10.</i> Настройка интерфейса <i>CorelDraw</i> и создание изображение.	274
35-§.	<i>Лабораторное занятие №11.</i> Создание базы данных в <i>MS. Access</i> .	281
ГЛАВА 4. КОМПЬЮТЕРНЫЕ СЕТИ.		
МОДУЛЬ 4		
36-§.	Компьютерные сети: понятие и возможности.	287
37-§.	Организация сети интернет.	295
38-§.	Технология создания <i>Web-страниц</i> .	304
39-§.	<i>Практическое занятие №14.</i> Создание веб страницы в <i>HTML</i> .	311
40-§.	<i>Практическое занятие №15.</i> Создание <i>Web-страниц</i> с помощью <i>Dreamweaver</i> .	319

41-§.	<i>Лабораторное занятие №12.</i> Поиск и сохранение информации в сети интернет.	325
42-§.	<i>Лабораторное занятие №13.</i> HTML теги для оформления текста. Теги для работы с таблицами и с HTML изображениями.	329
43-§.	<i>Лабораторное занятие №14.</i> Создание Web-страниц с помощью <i>Dreamweaver</i> .	338
44-§.	<i>Лабораторное занятие №15.</i> Работа с различными средствами защиты информации	341
ГЛАВА 5. ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ИХ ИСПОЛЬЗОВАНИЕ		
МОДУЛЬ 5		
45-§.	Основные понятия информационных систем.	347
46-§.	Понятие информационной безопасности.	356
47-§.	Электронно-цифровая подпись.	363
48-§.	<i>Практическое занятие №16.</i> Автоматизированные информационные системы	368
49-§.	<i>Практическое занятие №17.</i> Работа с различными средствами защиты информации	374
50-§.	<i>Практическое занятие №18.</i> Обеспечение безопасности компьютерной сети	382
	Список использованных сокращений	391
	Список использованной литературы	392
	Приложения	398
	Глоссарий	406

C O N T E N T

CHAPTER 1. MODERN INFORMATION TECHNOLOGIES AND THEIR APPLICATION.		
MODULE 1		
	Introduction	4
1-§.	Information technology <i>subject, purpose and objectives</i> . The concept of information, its types and properties.	8
2-§.	Architecture of a modern personal computer (<i>PC</i>).	18
3-§.	Operating systems (<i>OS</i>). Types of operating systems.	29
4-§.	<i>Practical lesson №1</i> . Service software, utilities.	37
5-§.	<i>Practical lesson №2</i> . Connecting devices to the system unit.	41
6-§.	<i>Laboratory lesson №1</i> . Testing your computer (<i>CPU, video card, HDD, RAM</i>).	51
7-§.	<i>Laboratory lesson №2</i> . Configuring your computer using BIOS Setup.	59
8-§.	<i>Laboratory lesson №3</i> . Wrappers and archivers.	63
CHAPTER 2. MODERN PROGRAMMING TECHNOLOGIES		
MODULE 2		
9-§.	Algorithm and its properties.	72
10-§.	The Pascal programming language. The Operators Of Pascal.	79
11-§.	<i>Practical lesson №3</i> . <i>ABC Pascal programming environment</i> .	92
12-§.	<i>Practical lesson №4</i> . <i>Operators in Pascal</i> .	98
13-§.	<i>Practical lesson №5</i> . <i>Graphical features of the PascalABC language</i> .	108
14-§.	<i>Laboratory lesson №4</i> . <i>Entering and debugging a linear program</i> .	113
15-§.	<i>Laboratory lesson №5</i> . <i>Entering and debugging programs with branching</i> .	118
16-§.	<i>Laboratory lesson №6</i> . <i>Creating programs using loops</i> .	123
CHAPTER 3. THE USE OF THE APPLICATION.		
MODULE 3		
17-§.	Spreadsheet editors. Their ability.	128
18-§.	Basic concepts of multimedia. Basics of working with audio and video information.	149
19-§.	Computer graphics. Types of computer graphics.	163
20-§.	CorelDraw graphics editor and its features.	175
21-§.	Creating and managing a database (<i>DB</i>).	184
22-§.	Translation programs.	193

23-§.	<i>Practical lesson №6. Automating text editing and formatting.</i>	204
24-§.	<i>Practical lesson №7. The basics of working in Microsoft Excel.</i>	210
25-§.	<i>Practical lesson №8. Creating presentations in Microsoft PowerPoint.</i>	217
26-§.	<i>Practical lesson №9. Basic concepts and features of computer graphics.</i>	223
27-§.	<i>Practical lesson №10. Order, merge, and group objects in CorelDraw.</i>	229
28-§.	<i>Practical lesson №11. Creating and managing a database.</i>	233
29-§.	<i>Practical lesson №12. Creating a database form in the MS environment. Access</i>	242
30-§.	<i>Practical lesson №13. Creating queries and reports in the MS DBMS. Access.</i>	248
31-§.	<i>Laboratory lesson №7.Editing and formatting documents in the MS Word text editor.</i>	255
32-§.	<i>Laboratory lesson №8.Technology for processing data in spreadsheets.</i>	261
33-§.	<i>Laboratory lesson №9.Creating training presentations.</i>	268
34-§.	<i>Laboratory lesson №10.Configure the CorelDraw interface and create an image.</i>	274
35-§.	<i>Laboratory lesson №11.Creating a database in MS. Access.</i>	281
CHAPTER 4. COMPUTER NETWORK.		
MODULE 4		
36-§.	Computer networks: concept and capabilities.	287
37-§.	Organization of the Internet.	295
38-§.	Technology for creating <i>Web pages</i> .	304
39-§.	<i>Practical lesson №14. Creating a web page in HTML.</i>	311
40-§.	<i>Practical lesson №15. Creating Web pages using Dreamweaver.</i>	319
41-§.	<i>Laboratory lesson №12.Search and save information on the Internet.</i>	325
42-§.	<i>Laboratory lesson №13.HTML tags for text formatting. Tags for working with tables and HTML images.</i>	329
43-§.	<i>Laboratory lesson №14.Creating Web pages using Dreamweaver.</i>	338
44-§.	<i>Laboratory lesson №15.Working with various information security tools</i>	341
CHAPTER 5. INFORMATION SYSTEMS AND THEIR USE		
MODULE 5		
45-§.	Basic concepts of information systems.	347
46-§.	The concept of information security.	356
47-§.	Digital signature.	363
48-§.	<i>Practical lesson №16. Automated information system</i>	368

49-§.	<i>Practical lesson №17. Working with various information security tools</i>	374
50-§.	<i>Practical lesson №18. Computer network security</i>	382
	List of abbreviations used	391
	References	392
	Application	398
	Glossary	406