

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

**Методические рекомендации для
самостоятельной работы обучающихся по дисциплине**

Б1.Б.16 ИНФОРМАТИКА

Направление подготовки Агроинженерия

Профиль образовательной программы Технический сервис в АПК

Форма обучения заочная

СОДЕРЖАНИЕ

1. Организация самостоятельной работы.....	3
2. Методические рекомендации по самостоятельному изучению вопросов.....	4
3. Методические рекомендации по подготовке к занятиям.....	23
3.1 Лабораторные работы по теме «Информация и информационные процессы. Представление информации. Информационно-логические основы построения персонального компьютера. Состав и структура ЭВМ и ПЭВМ. Программное обеспечение персонального компьютера»	23
3.2 Лабораторные работы по теме «Основы алгоритмизации и программирования. Базы данных. Модели решения задач. Компьютерные сети. ».....	24
4. Методические рекомендации по выполнению индивидуальных домашних задания.....	26

1. ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

1.1. Организационно-методические данные дисциплины

№ п.п.	Наименование тем	Общий объем часов по видам самостоятельной работы				
		подготовка курсового проекта (работы)	подготовка реферата/эссе	индивидуальные домашние задания (ИДЗ)	самостоятельное изучение вопросов (СИВ)	подготовка к занятиям (ПкЗ)
1	2	3	4	5	6	7
1	Тема 1 Информация и информационные процессы		x			
2	Тема 2 Представление информации		x		7	
3	Тема 3 Системы счисления		x			
4	Тема 4 Логические основы построения персонального компьютера		x		15	
5	Тема 5 Персональный компьютер		x		15	
6	Тема 6 Программное обеспечение		x		15	
7	Тема 7 Текстовые редакторы		x	2		2
8	Тема 8 Графические редакторы		x	3		2
9	Тема 9 Электронные таблицы		x	5		6
10	Тема 10 Базы данных		x			
11	Тема 9 Электронные таблицы		x			3
12	Тема 10 Базы данных		x			3
13	Тема 11		x		30	

	Алгоритмизация и программирование					
14	Тема 12 Локальные и глобальные сети		х		23	
15	Тема 13 Основы защиты информации		х		16	

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО САМОСТОЯТЕЛЬНОМУ ИЗУЧЕНИЮ ВОПРОСОВ

2.1 Адекватность информации.

адекватность – это степень соответствия образа, создаваемого с помощью полученной информации реальному объекту.

От степени адекватности информации зависит правильность принятия решений человеком или управляющей системой.

Адекватность может выражаться в трех формах.

Синтаксическая адекватность отображает формально-структурные характеристики информации и не принимает во внимание ее содержательную часть. На синтаксическом уровне учитывается тип носителя, способ представления, скорость передачи информации.

Семантическая (смысловая) адекватность определяет степень соответствия образа объекта и самого объекта. На этом уровне анализируются те сведения, которые отражает информация, рассматриваются смысловые связи. Эта форма служит для формирования понятий и представлений, выявления смысла, содержания информации и ее обобщения.

Прагматическая (потребительская) адекватность отражает соответствие информации цели управления, реализуемой на ее основе. Эта форма адекватности связана с практическим использованием информации, с ее соответствием целевой функции деятельности системы.

2.2 Информационные процессы

Информационный процесс - совокупность последовательных действий (операций), производимых над информацией (в виде данных, сведений, фактов, идей, гипотез, теорий и пр.), для получения какого-либо результата (достижения цели).

Информация проявляется именно в информационных процессах. Информационные процессы всегда протекают в каких-либо системах (социальных, социотехнических, биологических и пр.).

Наиболее обобщенными информационными процессами являются сбор, преобразование, использование информации.

К основным информационным процессам, изучаемым в курсе информатики, относятся: поиск, отбор, хранение, передача, кодирование, обработка, защита информации.

2.3 Кодирование при передаче и хранении информации.

Под кодированием информации понимается преобразование формы представления информации с целью обеспечения удобства её передачи по каналам связи или хранения.

В широком смысле под кодированием понимают представление символов одного алфавита символами другого.

Виды кодирования:

- 1) кодирование по образцу. Пример : Для перевода символов, вводимых с клавиатуры, в числовой код, хранящийся в памяти компьютера, используется кодовая таблица, в которой каждому символу алфавита, а также множеству специальных управляющих команд соответствует числовой код.
 - 2) Криптографическое кодирование, или шифрование, используется тогда, когда нужно защитить информацию от несанкционированного доступа.
 - 3) Оптимальное кодирование. Обычно используется в архиваторах.
 - 4) Помехозащищенное кодирование служит для передачи данных по каналам связи и учитывает возможность возникновения помех и связанного с этим искажения или утраты части данных.
- (существуют 2 типа помехозащитного кодирования: кодирование с фиксацией ошибок и с кодированием ошибок).

2.4 Основы классификации и структурирования информации

Классификация- система распределения объектов по классам в соответствии с определенным признаком.

Структуру информационной системы составляет совокупность отдельных ее частей, называемых подсистемами. Таким образом, структура любой информационной системы может быть представлена совокупностью обеспечивающих подсистем:

- 1) **Информационное обеспечение**-совокупность единой системы классификации и кодирования информации, унифицированных систем документации, схем информационных потоков циркулирующих в организации, а также методология построения баз данных. Назначение подсистемы информационного обеспечения состоит в своевременном формировании и выдаче достоверной информации для принятия управленческих решений.
- 2) **Техническое обеспечение**- это комплекс технических средств, предназначенных для работы информационной системы, а также соответствующая документация на эти средства и технологические процессы. Комплекс технических средств составляют: компьютеры любых моделей; устройства сбора, накопления, обработки, передачи и вывода информации.
- 3) **Математическое и программное обеспечение**- это совокупность математических методов, моделей, алгоритмов и программ, необходимых для достижения целей и решения задач информационной системы, а также для нормального функционирования комплекса технических средств. В состав программного обеспечения входят общесистемные и специальные программные продукты, а также техническая документация.
- 4) **Организационное обеспечение**- это совокупность методов и средств, регламентирующих взаимодействие работников с техническими средствами и между собой в процессе разработки и эксплуатации информационной системы.
- 5) **Правовое обеспечение** – это совокупность правовых норм, определяющих создание, юридический статус и функционирование информационных систем, регламентирующий порядок

получения, преобразования и использования информации. Главной целью правового обеспечения является укрепление законности.

2.5 Форма представления данных в компьютере

Данные в компьютере- сигналы, зафиксированные на физическом носителе в форме пригодной для постоянного хранения, передачи и обработки. Для хранения данных используется ячейка памяти.

Виды информации:
1. Числовая
2. Звуковая
3. Графическая
4. Видео.

Числовые данные. Числовые данные в компьютере представляются двумя подмножествами: целыми и действительными числами. Целые числа - любое число, записанное без десятичной точки. Действительные числа (вещественные) - числа, где нужно использовать десятичную точку, а не запятую.

Мультимедийные данные. Графические изображения представлены в компьютере в виде двоичных чисел. Существуют 2 типа графических форматов: векторный и растровый. Растровый формат - совокупность точек, каждая из которых имеет атрибуты, ресурсоемкий формат хранения. Например: рисунок размером 320 на 200 точек потребует для хранения $= 320 \times 200 \times 4 = 256000$ байт. Векторный формат - графический объект, составленный из точек, отрезков, линий, прямых, дуг, окружностей. Например: тоже графическое изображение в векторном формате занимает 41 Кбайт, а в растровом - 395 Кбайт. Разница в 10 раз.

2.6 Выполнение арифметических операций

Сложение

Если результат сложения двух цифр в P -ичной системе счисления больше $P - 1$ (то есть полученное число двузначное), то старшая цифра результата всегда равна 1. Таким образом, при сложении столбиком в следующий разряд может переходить только единица, а результат сложения в любом разряде будет меньше, чем P . Результат сложения двух положительных P -ичных чисел либо имеет столько же значащих цифр, что и максимальное из двух слагаемых, либо на одну цифру больше, но этой цифрой может быть только единица. Такая закономерность обуславливает возможность сложения в столбик не только десятичных, но и любых P -ичных чисел.

Поскольку сложение в десятичной системе для нас является привычным (как и другие арифметические действия), оно не вызывает затруднений. В других P -ичных системах для облегчения сложения (вычитания) применяются специальные таблицы, сходные с привычной нам таблицей умножения

Вычитание

Алгоритм:

Если необходимо вычесть из цифры a цифру b и $a < b$, то в столбце b таблицы сложения ищем значение числа a . Самая левая цифра в строке, в которой найдено значение числа a , и будет результатом вычитания. Если же $a < b$, нам нужно заимствовать единицу из левого разряда, поэтому в столбце b мы ищем число $1a$, и левая цифра в соответствующей строке будет результатом вычитания.

Умножение

Для того чтобы выполнять умножение чисел в Р-ичной системе, нам необходимы как таблица сложения, так и таблица умножения для этой системы

Умножение производится по правилам умножения столбиком с использованием соответствующих таблиц умножения и сложения.

Деление

Как и для умножения, для деления нужны обе таблицы, умножения и сложения в соответствующей Р-ичной системе. Само деление выполняется по привычным правилам деления уголком с последовательным вычитанием сомножителей.

Алгебра логики устанавливает истинность или ложность логического высказывания с помощью алгебраических методов. Истина(1), ложь(0). – так описываются сложные логические высказывания. Логическое высказывание - фразы "2 больше 1", "5,8 является целым числом". Логические операции. **Конъюнкция** (и) обозначается как и, похожей на восьмерку-умножение, **дизъюнкция**(или) двумя палками - сложение, **отрицание**(не) чертой над высказыванием.

2. 7 Основные понятия логики

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

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

2.8 Логические операции. Таблицы истинности

Логическая операция **КОНЪЮНКЦИЯ** (логическое умножение).

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

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

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

Логическая операция **ИМПЛИКАЦИЯ** (логическое следование).

Импликация (\Rightarrow) - это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся ложным тогда и только

тогда, когда условие (первое высказывание) истинно, а следствие (второе высказывание) ложно.

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

Таблицу, показывающую, какие значения принимает составное высказывание при всех сочетаниях (наборах) значений входящих в него простых высказываний, называют *таблицей истинности* составного высказывания. Составные высказывания в алгебре логики записываются с помощью логических выражений. Для любого логического выражения достаточно просто построить таблицу истинности

Алгоритм построения таблицы истинности:

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

2.9Связь между алгеброй логики и двоичным кодированием

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

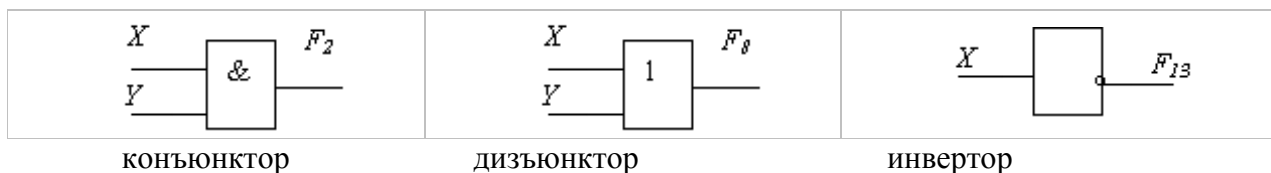
Из этого следует два вывода: 1) одни и те же устройства компьютера могут применяться для обработки и хранения как числовой информации, представленной в двоичной системе счисления, так и логических переменных; 2) на этапе конструирования аппаратных средств алгебра логики позволяет значительно упростить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

2.10 Данные и команды в памяти компьютера и в регистрах процессора

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

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

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



Р и с у н о к 0.1 -- Базовые логические элементы

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

2.11 Различные виды классификации ПО.

1. Классификация ПО по степени взаимодействия

Прикладное программное обеспечение не взаимодействует напрямую с аппаратным обеспечением компьютера. В качестве примера прикладного программного обеспечения можно привести текстовый процессор. **Системное программное обеспечение** составляют операционная система с набором инструментов (утилит) администрирования и настройки, а также базовая система ввода-вывода (БИОС). **Промежуточное программное обеспечение** — класс программного обеспечения, располагающийся между системным и пользовательским программным обеспечением. К промежуточному программному обеспечению относят серверы баз данных.

2. Классификация ПО по виду лицензирования

На заре компьютерной эры все программы были открытыми и бесплатными, к компьютерам они прилагались, составляя крайне незначительную часть стоимости. По мере развития и массового внедрения в жизнь компьютерной техники наступила эпоха коммерциализации программного обеспечения. Исходные коды программ стали повсеместно закрываться патентами, лицензиями, соглашениями о неразглашении, и практически все программное обеспечение переводилось на коммерческие рельсы. Символом успешности в области создания коммерческого программного обеспечения стал Билл Гейтс, владелец и в прошлом глава корпорации Майкрософт. Наиболее известным и используемым программным продуктом этой всемирной

корпорации является операционная система Майкрософт Виндоус. Монопольное положение на рынке сбыта программного обеспечения, удобное для его крупных производителей, во многих проявлениях не было воспринято обществом. Открытое программное обеспечение неразрывно связано с именем Ричарда Столлмана. Увидев в тотальной коммерциализации программного обеспечения не просто монополистическую тенденцию, но явную склонность к ущемлению базовых человеческих прав и свобод, Столлман предложил, используя имеющиеся правовые механизмы, производить программы, защищенные от коммерческой несвободы специальным видом лицензии.

3. Классификация по платформе назначения: Одноплатформенное программное обеспечение разрабатывается и выполняется только на одной операционной платформе (Майкрософт), Переносимое программное обеспечение — программное обеспечение, разработанное в технологиях программирования, подразумевающих возможность создания вариантов программы под разные платформы (C++). Межплатформенное программное обеспечение — программное обеспечение, одинаково функционирующее в любой операционной среде. Полностью межплатформенных приложений всего два вида: Джава- и веб-приложения.

4. Классификация по взаимодействию с пользователями: Программы с графическим оконным интерфейсом (все современные); Консольные программы; Программы системного лотка отображаются исключительно в области уведомлений панели задач, управляются из контекстного меню и не имеют оконного интерфейса. **3) По способу**

установки: требующие установки; не требующие установки; переносимые, то есть запускаемые с любых сменных носителей.

5. По отношению к сети.

6. По аппаратной платформе.

7. По целям разработки: массовые, заказное, целевое и индивидуальное.

8. По степени опасности: опасные и безопасные.

2. 12 Операционная система. Состав операционной системы

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

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

Состав ОС

1. Программные модули, управляющие файловой системой, т.к. процесс работы компьютера в определенном смысле сводится к обмену файлами между устройствами.

2. Командный процессор, — программа, которая запрашивает у пользователя команды и выполняет их.

3. Драйверы устройств - специальные программы, которые обеспечивают управление работой устройств и согласование информационного обмена с другими устройствами, а также позволяют производить настройку некоторых параметров устройств. Каждому устройству соответствует свой драйвер.

4. Программные модули, создающие графический пользовательский интерфейс. В операционных системах с графическим интерфейсом пользователь может вводить команды с помощью мыши, тогда как в режиме командной строки необходимо вводить команды с помощью клавиатуры.

5. Справочная система - позволяет оперативно получить необходимую информацию как о функционировании операционной системы в целом, так и о работе ее отдельных модулей.

2.13 Загрузка операционной системы

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

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

После проведения самотестирования специальная программа, содержащаяся в BIOS, начинает поиск загрузчика операционной системы. Происходит поочередное обращение к имеющимся в компьютере дискам (гибким, жестким, CD - ROM) и поиск на определенном месте (в первом, так называемом загрузочном секторе диска) наличия специальной программы Master Boot (программы-загрузчика операционной системы).

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

2. 14 Операционная система MSDOS. Основные команды MSDOS

Команды – способ общения пользователя с компьютером в операционной системе.

Командная строка – строка экрана, начинающаяся с приглашения операционной системы. **Формат команды** – правило формирования команды пользователем с клавиатуры.

При формировании команды в соответствии с установленным форматом запомните правила:

- формат команды состоит из имени команды латинскими буквами (без типа) и отделенных от нее одним пробелом параметров, уточняющих ее действие;
- в большинстве случаев параметры между собой пробелом не разделяются, а в качестве разделителя часто используется символ / или \;
- параметрами могут быть: имя дисководов, путь, имя файла, тип файла, латинские буквы, символы, цифры;

Рассмотрим основные команды DOS (См. Табл. 7.1)

Т а б л и ц а 0.1 – Основные команды DOS

Команды работы с каталогами	Команды работы с файлами
Команда смены текущего дисководов.	Создание текстовых файлов:
A: — переход на дисковод A:	<i>Cору con</i> имя-файла
C: — переход на дисковод C:	нажать клавишу <i>F6</i> и затем <i>Enter</i> .
D: — переход на дисковод D:	
Создание каталога:	Копирование файлов:
<i>Md [дисковод:] путь</i>	<i>Cору</i> имя-файла1 имя-файла2 или
	<i>сору</i> имя-файла1 [имя-каталога]
Изменение текущего каталога :	Переименование файлов:
<i>Cd [дисковод:] путь</i>	<i>Ren</i> имя-файла1 имя-файла2
Уничтожение каталога:	Соединение (конкатенация) файлов:
<i>Rd [дисковод:] путь</i>	<i>Cору</i> имя-файла [+ имя-файла]. Имя-файла
Команды работы с каталогами	Команды работы с файлами
Удаление каталога со всем содержимым: <i>Deltree</i> имя-файла-или-каталога [/Y]	Удаление файлов: <i>Del</i> имя-файла
Просмотр каталога. <i>dir</i> [дисковод:] [путь\] [имя-файла] [параметры]	Вывод файла на экран. <i>type</i> имя-файла

2. 15Файловые менеджеры

Применение файлового менеджера значительно упрощает управление компьютером по сравнению с MSDOS, так как процесс ввода команд и их параметров заменяется выбором из предлагаемого списка возможных значений (меню). Выбор, как правило, осуществляется с помощью клавиш редактирования со стрелками и завершается нажатием клавиши <Enter>.

Кроме того, при правильной работе с файловым менеджером практически не требуется указывать пути к файлам и каталогам, что при непосредственной работе в DOS часто сопровождается ошибками.

Перечень процедур управления компьютером с помощью файлового менеджера весьма широк. Основными из них являются выбор диска, каталога и файла; создание нового каталога и текстового файла; просмотр и корректировка текстовых файлов; копирование, перемещение, удаление файлов и каталогов; поиск файлов и каталогов на дисках; работа с архивными файлами; получение информации о компьютере и оперативной памяти; сравнение каталогов дисков; просмотр и корректировка атрибутов файлов; автоматизация загрузки обрабатывающих программ с помощью меню пользователя и файла расширений; настройка параметров пакета и их сохранение.

2.16 Операционная система AltLinux

Операционная система (ОС) ALT Linux — это комплекс взаимосвязанных программ, который действует как интерфейс между приложениями и пользователями с одной стороны и аппаратурой компьютера с другой стороны, и предназначенный для выполнения двух основных функций:

- предоставление пользователю или программисту вместо реальной аппаратуры расширенной виртуальной машины (операционной среды);
- повышение эффективности использования компьютера путём рационального управления его ресурсами в соответствии с определенными критериями.

Программное обеспечение ОС ALT Linux подразделяется на:

- ядро операционной системы (используется ядро Linux 2.6.18-std-smp-alt5);
- программные компоненты операционной системы.

ОС ALT Linux включает в себя компоненты, реализующие развитый интерфейс пользователя в части удаленного контроля и управления функционированием ОС ALT Linux и составляющих ее компонентов.

2.17 Общее представление о функциональности

В эпоху текстовых редакторов главной задачей разработчиков было обеспечить при наборе, верстке и распечатке документа его максимальную схожесть с машинописным текстом. Для современного текстового процессора такого рода требования являются не верхним пределом, а нижней границей функциональности, минимальными требованиями. Какими же возможностями должен обладать полнофункциональный текстовый процессор?

Ввод и редактирование текста

Поиск и замена

Создание, открытие и сохранение документов

Возможность сохранения документов в различных форматах — одна из важнейших функций современного текстового процессора. Поскольку мир современного документооборота и распространения информации очень многообразен, тот же документ может быть востребован в разных своих представлениях:

- *в виде текста* — сохранение только текстового содержимого;
- *в виде документа* — сохранение в системе электронного документооборота (СЭД);
- *в виде документа другого формата* — для открытия в текстовом процессоре другого типа;
- *в формате PDF* — для последующей печати или чтения в другой операционной системе;
- *в формате HTML* — для публикации на веб-странице;
- *в виде записи блога* — для публикации в виртуальном дневнике.

Интеграция в документ дополнительных сведений, реквизитов и атрибутов

Поскольку сегодня практически весь крупный и средний бизнес, государственные учреждения и структуры, научные учреждения и учебные заведения оснащены тем или иным видом системы электронного документооборота, то наличие в документе дополнительных сведений, реквизитов и атрибутов в одном случае является необходимым, в другом облегчает процесс создания карточки документа и регистрации документа в информационной системе.

Форматирование

Для качественного отображения документа очень важно, чтобы текстовый процессор умел отображать фрагменты текста различными типами шрифта. Стандартным набором параметров шрифта, которые должен уметь изменять текстовый процессор, являются: *семейство начертание*, в том числе обычное, полу жирное, курсивное и *размер* шрифта

Кроме того, текстовый процессор должен обеспечивать средства *подчеркивания* и *зачеркивания* шрифта, ввод *надстрочных* и *подстрочных* символов, задания *цвета* для шрифта и фона (рис. 17.6).

К числу дополнительных параметров, имеющих не во всех текстовых процессорах, обычно относят средства для придания шрифту разнообразных эффектов, таких как *рельеф*, *тень* или *анимация*.

Если стандартный набор параметров в основном используется в официальных документах и деловой переписке, то дополнительный может быть широко задействован при создании разного рода публикаций (буклетов, брошюр, афиш, веб-страниц).

Возможности некоторых текстовых процессоров, касающиеся изменения межбуквенного интервала и ширины букв, свойственны скорее настольным системам верстки.

Форматирование абзацев

Форматирование абзацев непосредственно связано с аккуратной версткой текста и заключается в задании величины выравнивания, отступов (слева, справа и первой строки) и междустрочного расстояния. При форматировании абзацев применяется четыре типа выравнивания: по правому краю, по левому краю, по центру и по ширине

Создание списков

Создание списков, как нумерованных, так и маркированных, — одна из важнейших функций современного текстового процессора. При этом обычно поддерживается возможность создания вложенных списков

Стилевое оформление документа

К числу инструментов текстового процессора, автоматизирующих подготовку документа, являются средства для автоматического создания оглавлений и указателей, списков иллюстраций и использованной литературы, а также для автоматического форматирования вводимого текста. Однако чтобы воспользоваться этими средствами, вводимый текст нужно оформлять предназначенным для этого стилем, например, заголовки — стилями заголовков с соблюдением уровней вложенности, текст — стилем обычного текста, и т. п.

Разбиение документа на разделы, страницы, нумерация страниц и оформление колонтитулов

При создании многостраничного документа весьма важно автоматизировать процедуры нумерации страниц, создания колонтитулов, разбиения документа на разделы, когда внутри каждого из разделов может существовать собственная нумерация страниц. Эти вопросы выходят на первый план, когда приходится оформлять в текстовом процессоре курсовую или дипломную работу, методические рекомендации или многостраничный документ, предназначенный для печати в малой типографии.

Использование текстовых фреймов для верстки документа

Текстовые фреймы обычно используются при создании шаблонов бланков официальных документов, а также разного рода брошюр, журналов, афиш и проспектов. С помощью фреймов любой объект, включая текст, иллюстрации или таблицы, можно разместить произвольным образом внутри печатного листа

Таблицы

Возможность табличного представления информации также является одним из важнейших свойств современного текстового процессора. В зависимости от возможностей текстового процессора таблица может допускать более или менее разнообразное стилевое оформление. Текстовый процессор может поддерживать преобразование текста в таблицу и таблицы в текст, заполнение таблицы из внешнего источника данных, некоторую автоматизацию подсчетов в таблице, например, суммирование по строкам и столбцам.

Вставка в документ нетекстовых объектов

Именно средства вставки в текстовый документ графических и мультимедийных объектов, гиперссылок и других сложных объектов отличают сложный текстовый процессор от обычного текстового редактора. Эта функциональность позволяет оформлять текстовые документы, которые затем могут быть легко преобразованы как в мультимедийную презентацию, так и в полноценную веб-страницу.

Рецензирование и отображение изменений

Эта функция обогащает текстовый процессор возможностью взаимодействия и коллективной работы нескольких человек над одним документом. Автор документа всегда может увидеть, что именно исправил рецензент, прочесть написанные примечания, принять или отвергнуть предложенные изменения.

Программирование и автоматизация рутинных операций

Часто некоторую последовательность операций приходится повторять. В офисных приложениях для того, чтобы рутинные операции такого сорта не выполнять многократно, есть возможность записи или программирования макрокоманд и связывания этих макрокоманд с определенным сочетанием клавиш или элементом управления. Многие офисные приложения имеют развитый макроязык (язык написания макрокоманд) и даже встроенные системы программирования.

Подсчет статистики документа

Хотя средства подсчета статистики документа обычно не относят к самым важным, они могут быть чрезвычайно полезны для систем электронного документооборота, позволяя выяснить, сколько документов, абзацев, предложений, слов и букв было обработано.

Печать конвертов для рассылки со слиянием данных

Автоматизированную печать конвертов (или автоматизированную рассылку документа по электронным адресам) в режиме, когда данные адресатов выбираются из источника данных, можно считать важной функцией, которая экономит время.

2.18 Сравнительная характеристика текстовых процессов

Текстовые процессоры разделяются на три группы:

1. *редакторы общего назначения* (сюда относятся, например, редакторы «Лексикон», MS Word, WordPerfect и др.);
2. *редакторы научных текстов* (ChiWriter, TeX и др.);
3. *редакторы исходных текстов программ* (например, Multi-Edit или встроенные редакторы систем программирования для языков Бейсик, Фортран, Си и др.).

Многооконный редактор «Лексикон» — один из простейших текстовых редакторов отечественной разработки для несложных документов на русском или английском языке

Текстовый редактор MS DOS Editor может применяться только для редактирования текстов файлов канонического формата, состоящих лишь из букв и цифр и не имеющих изображений. Редактор может вызываться из MS DOS и Windows и по своим возможностям близок к «Лексикону», но имеет более удобный интерфейс.

Текстовый редактор Windows 95 NotePad («Блокнот») — простейший редактор, входящий в Windows 95. Имеет ограниченные возможности, но очень простой для изучения и использования.

Многооконный текстовый процессор MS Word 2002 (последняя версия редактора MS Word) — один из самых совершенных, входит в пакет MS Office 2002 и имеет сотни операций над текстовой и графической информацией. Как и ОС Windows, построен по оконной технологии с использованием ниспадающих меню и пиктограмм. Обладает универсальностью, позволяет достаточно быстро и с высоким качеством готовить практически любые документы: от деловой записки или письма до макета книги.

2.19 Сравнительная характеристика табличных процессов

Табличные процессоры позволяют производить обработку, анализ и хранение информации текстовой, числовой и графической, наглядно представленной в виде таблиц. При этом часть ячеек таблицы содержит исходную информацию, а часть производную. Производная информация является результатом обработки исходной информации средствами табличного процессора.

Табличные процессоры характеризуются:

- типом процессора и операционной системы необходимой для работы (i-386, Pentium, ..., MS-DOS 6.22, Windows 95, OS/2,...);
- необходимым объемом свободной памяти - оперативной и внешней (на жестком диске);
- типами используемых мониторов, принтеров и другой периферийной техники;
- возможностью работать в сети и типом сетевой архитектуры;
- совместимостью по формату файлов созданных в других версиях электронных таблиц и базах данных.

2.20 Понятие алгоритма. Исполнитель алгоритма

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

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

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

2.21 Основные свойства алгоритмов

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

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

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

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

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

2.22 Формы записи алгоритмов

На практике наиболее распространены следующие формы представления алгоритмов:

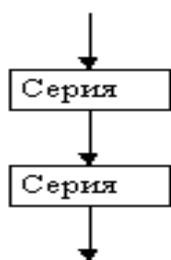
- словесная (запись на естественном языке);
- графическая (изображения из графических символов);
- псевдокоды (полуформализованные описания алгоритмов на условном алгоритмическом языке, включающие в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические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

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

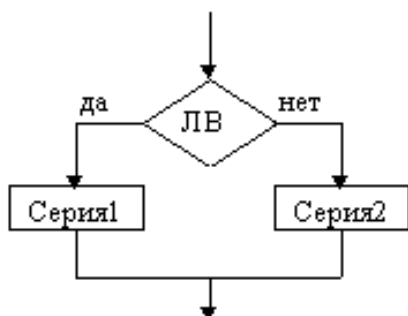
2.23 Графические обозначения (обозначения на блок-схемах)

При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из

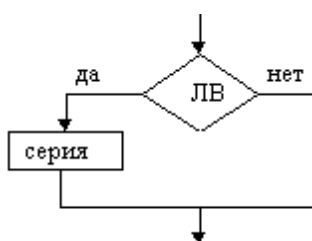
которых соответствует выполнению одного или нескольких действий ((См. Рис. 13.1 – 13.6)



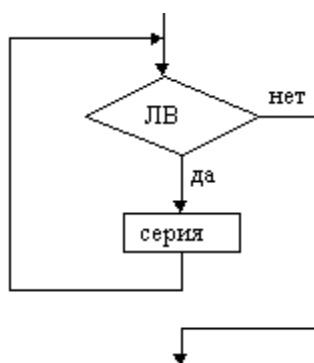
Р и с у н о к 0.1. -- Структура “следование”



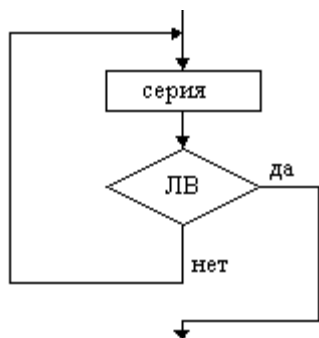
Р и с у н о к 0.2 -- Полная развилка



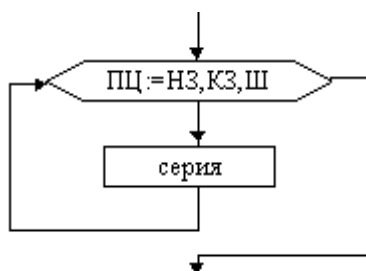
Р и с у н о к 0.3--Неполная развилка



Р и с у н о к 0.4. -- Цикл с предусловие (цикл ПОКА)



Р и с у н о к 0.5 -- Цикл с постусловием (цикл ДО)



Р и с у н о к 0.6 -- Цикл с параметром

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

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

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

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

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

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

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

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

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

Для организации цикла необходимо выполнить следующие действия:

- перед началом цикла задать начальное значение параметра;
- внутри цикла изменять параметр цикла с помощью оператора

присваивания;

- проверять условие повторения или окончания цикла;
- управлять циклом, т.е. переходить к его началу, если он не закончен, или выходить из цикла в противном случае.

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

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

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

2.24 Формализация понятия алгоритма. Современная теория алгоритма.

Приведенное определение алгоритма нельзя считать представленным в привычном математическом смысле. Математические определения фигур, чисел, уравнений, неравенств и многих других объектов очень четки. Каждый математически определенный объект можно сравнить с другим объектом, соответствующим тому же определению. Например, прямоугольник можно сравнить с другим прямоугольником по площади или по длине периметра. Возможность сравнения математически определенных объектов – важный момент математического изучения этих объектов. Данное определение алгоритма не позволяет сравнивать какие-либо две таким образом определенные инструкции. Можно, например, сравнить два алгоритма решения системы уравнений и выбрать более подходящий в данном случае, но невозможно сравнить алгоритм перехода через улицу с алгоритмом извлечения квадратного корня. С этой целью нужно формализовать понятие алгоритма, т.е. отвлечься от существа решаемой данным алгоритмом задачи, и выделить свойства различных алгоритмов, привлекая к рассмотрению только его форму записи. Задача нахождения единообразной формы записи алгоритмов, решающих различные задачи, является одной из основных задач теории алгоритмов. В теории алгоритмов предполагается, что каждый шаг алгоритма таков, что его может выполнить достаточно простое устройство (машина). Желательно, чтобы это устройство было универсальным, т.е. чтобы на нем можно было выполнять любой алгоритм. Механизм работы машины должен быть максимально простым по логической структуре, но настолько точным, чтобы эта структура могла служить предметом математического исследования. Впервые это было сделано американским математиком Эмилем Постом в 1936 (машина Поста) еще до создания современных вычислительных машин и (практически одновременно) английским математиком Аланом Тьюрингом (машина Тьюринга).

2.25 Классификация языков программирования. Краткая история языков программирования.

Языки программирования - это формальная знаковая система, предназначенная для описания алгоритмов в форме, которая удобна для исполнителя. Язык программирования предназначен для описания данных и алгоритмов их обработки на вычислительной машине. Языки программирования занимают промежуточное положение между естественными и формализованными языками. С естественными языками языки

программирования роднит грамматический строй, с формализованными языками — символы и понятия, а главное — строгие, точно описанные правила построения текстов. Каждый язык программирования предназначен для решения определенного класса задач: Фортран — старейший язык программирования, предназначен для решения математических задач.

Кобол — для решения экономических задач
Бейсик, Pascal — для обучения
Java (джава) — язык сетевого программирования.
Для системного программирования наиболее подходят языки C, C++ и Ассемблер. C и — язык разработанный для написания операционной системы UNIX (обычно ядро операционных систем писали на Assembler).

Языки нижнего уровня: Машинно-ориентированные языки (Ассамблер). **Языки высокого уровня:** Алгоритмические языки.

1)Машинные коды. Настоящее программирование в современном понимании началось с момента создания первой электронной вычислительной машины. Первые ЭВМ позволяли выполнять программы на машинном языке, который является единственным языком, понятным ЭВМ. Машинные команды пишутся в двоичном коде. Логика работы подобных программ очень трудно понять из-за того, программа представляет собой сплошной набор нулей и единиц. Прочитать и разобраться, как работает программа, написанная в двоичных кодах, было очень сложно, не говоря уже о том, чтобы найти и исправить в ней ошибку.

2)Мнемокоды. программисты придумали мнемокоды — буквенные обозначения машинных двоичных команд, которые проще запомнить, чем последовательности нулей и единиц. Для упрощения работы с ячейками памяти стали использовать понятие переменной. Для перевода мнемокодов в машинные инструкции и имен переменных в адреса ячеек памяти использовалась специальная программа — транслятор. Языки мнемокодов получили название ассемблеров.

3)Структурное программирование. Постепенно программисты приходили к выводу, что все вычисления сводятся к таким элементарным действиям, как ввод данных, вывод данных, выполнение различных операций над данными, выбор ветви выполнения программы на основе принятого решения, повторение группы операций. Появились языки, основанные на этих базовых операциях, которые стали называть структурными или языками высокого уровня.

4)Современное программирование. В настоящее время существуют множество языков программирования: как достаточно универсальных, так и очень специфических. Многие программисты старались и стараются придумать свой язык обладающий теми или иными преимуществами. Можно лишь условно разделить языки по определенным критериям. Например, по типу решаемых задач (язык системного или прикладного назначения), по степени ориентации на решение узкого круга задач (проблемно-ориентированные или универсальные).

2.26 Мультимедиа-презентации

Мультимедийные презентации могут иметь разные свойства и вид в зависимости от целей, которые необходимо достигнуть с использованием той или иной презентации. Например, презентация может рекламировать новый продукт компании или использоваться в образовательном процессе.

Слово Презентация — (лат. praesentatio) — представление чего-либо нового. Получается, что мультимедийная презентация — это представление чего-либо нового с использованием мультимедийных технологий. Существуют разные подходы для

создания, представления и распространения мультимедийных презентаций, которые мы рассмотрим далее.

Элементы мультимедийной презентации

Мультимедийная презентация отличается от стандартной тем, что она, как правило, содержит комбинации следующих основных элементов:

- Видео.
- Аудио.
- 3D модели.
- Рисунки, фотографии.
- Текст.
- Анимация.
- Навигация.

Данные элементы могут быть представлены в совершенно разных комбинациях. Какие то из представленных элементов могут отсутствовать и это вполне нормально.

2.27 Модели данных

Модель данных - это некоторая абстракция, которая будучи приложена к конкретным данным, позволяет пользователям и разработчикам трактовать их уже как информацию, то есть сведения, содержащие не только данные, но и взаимосвязь между ними. Модель, которая выражает информацию о предметной области в виде независимом от используемой СУБД, называется инфологической или семантическая модель (используется на ранних стадиях проектирования приложений). **Даталогические** – служат для описания структур данных в виде зависимом от используемой СУБД. **Физическая** - оперирует категориями, касающимися организации внешней памяти и структур хранения, используемых в данной операционной среде. Классификация БД. Обычно системы БД классифицируются в зависимости от модели данных, которая лежит в их основе. В общепринятой классификации в настоящее время известны следующие модели данных: **иерархическая**(Иерархической модели данных соответствует дерево – связной ориентированный граф, который не содержит циклов. **Сетевая**, В сетевой модели данных также как и в иерархической используется графический способ представления данных, но ограничения на количество связей не накладывается. **Реляционная**, В реляционной модели рассматриваются три аспекта данных – структура данных, целостность данных и обработка данных.**Объектно-ориентированная**, представляет собой расширенную реляционную модель, в которой отменено требование атомарности(неделимости) атрибутов. **Объектно-реляционная**. Объектно-ориентированную и объектно-реляционную модели часто называют постреляционными моделями данных БД . Соответственно говорят о сетевых, иерархических, реляционных, объектно-ориентированных, объектно-реляционных и постреляционных БД. На ПК самое широкое применение нашли реляционные модели данных.

2.28 Основные этапы компьютерного моделирования

Моделирование заключается в имитации изучаемого явления. Точность имитации определяется путем сравнения полученного при воспроизведении результата с его прототипом, объектом исследования, и оценки степени их сходства. В целом моделирование включает в себя четыре этапа:
1. Постановка задачи(Задача формулируется в общем виде, в форме словесных описаний, достаточных для того, чтобы определить цель моделирования. При этом модель

и ее характеристики напрямую зависят от поставленной исследователем цели.)

2. Разработка модели(На основе анализа результатов, полученных на предыдущем этапе, формируется информационная модель. На данном этапе исходные предположения переводятся на однозначный (обычно математический) язык количественных отношений и устраняются нечеткие, неоднозначные высказывания или определения. Завершается этап получением информационной модели объекта.)

3. Компьютерный эксперимент.

4. Анализ результатов моделирования(Анализ результатов моделирования заключается в установлении адекватности модели объекту исследования, то есть в определении степени сходства модели с ее оригиналом).

2.29 Развитие сетевых межкомпьютерных коммуникаций в России.

Классификация компьютерных сетей.

Компьютерные сети принято классифицировать по типам передачи данных (широковещательные, сети с передачей от узла к узлу) и по размеру (локальные, муниципальные и глобальные сети).

Классификация компьютерных сетей по типу передачи данных:

1)широковещательные сети(Широковещательные сети обладают единым каналом связи, совместно используемым всеми машинами сети. Короткие сообщения, называемые в некоторых случаях пакетами, которые посылаются одной машиной, получают все машины.)

2)сети с передачей от узла к узлу.(подразумевает наличие изолированного канала между двумя компьютерами. В этом случае отпадает необходимость в каждом сообщении оповещать, для какого именно компьютера оно предназначено.)

Классификация по типу коммутации между узлами: Физический уровень(определяет способ соединения узлов ,относящиеся к аппаратному обеспечению например модем). Логический уровень(определяет взаимодействие физическисоединенных узлов, согласно определенному порядку обмена сигналами) По типу коммутации между узлами различают сети с коммутацией каналов и пакетов.

♣ Сеть с коммутацией каналов(на время передачи данных между двумя узлами образуется цепь взаимосвязанных последовательных отрезков пути передачи данных, которая образует канал.)

Классификация по среде передачи данных: **Проводные:**

-Проводные соединения компьютеров реализуются при помощи медного провода.

-Кабельные соединения реализуются при помощи групп проводов, объединенных общей оболочкой.

-Кабельные коаксиальные соединения реализуются при помощи одножильного кабеля с экранирующей оплеткой, предназначенного для передачи переменного электрического тока высокой (радио) частоты.

-Оптоволоконные соединения реализуются при помощи кабеля, проводящего не электрический ток, а световой луч. В центре такого кабеля лежит специальный материал, называемый оптоволоконном.

Беспроводные:

-Радиочастотные наземные каналы представлены каналами наземной мобильной связи, технологиями Вай-Фай и Блутус.

-В случае радиочастотных спутниковых каналов данные передаются через искусственные спутники земли.

Классификация по территориальному охвату

-Персональная сеть создается в случае, когда несколько компьютерных устройств находятся в персональном пользовании одного человека, причем требуется передавать данные с одного устройства на другое (с наладонника или ноутбука на настольный стационарный компьютер, с мобильного телефона на наладонный компьютер, с коммуникатора на беспроводную гарнитуру).

-Локальная сеть в зависимости от назначения и расположения может быть и предельно простой, и очень сложной. отличительным признаком можно считать то, что она обычно не требует сложного телекоммуникационного оборудования.

-Муниципальная сеть прокладывается в масштабах города.

-Глобальные сети планируются и строятся в масштабах стран или регионов. Их признаком являются наличие высоко- и сверхвысокоскоростных магистралей.

-Интернет объединяет основные магистрали глобальных сетей, обеспечивая передачу данных из одной глобальной сети в другую и тем самым создавая единое информационное пространство для всего населения планеты.

Классификация по скорости передачи данных

<input type="checkbox"/>	низкоскоростные	(до	10	Мбит/с);
<input type="checkbox"/>	среднескоростные	(до	100	Мбит/с);
<input type="checkbox"/>	высокоскоростные	(до	1	Гбит/с);
<input type="checkbox"/>	сверхвысокоскоростные (до 10 Гбит/с).			

2.30 Защита информации в информационных системах.

Государственные стандарты по информационной безопасности.

Основные задачи государственной системы защиты информации:

- проведение единой технической политики, организация и координация работ по защите информации в оборонной, экономической, политической, научно-технической и других сферах деятельности;
- исключение или существенное затруднение добывания информации техническими средствами разведки, а также предотвращение ее утечки по техническим каналам, несанкционированного доступа к ней, предупреждение преднамеренных специальных программно-технических воздействий на информацию с целью ее разрушения, уничтожения, искажения или блокирования в процессе обработки, передачи и хранения;
- принятие в пределах компетенции нормативно-правовых актов, регулирующих отношения в области защиты информации;
- общая организация сил, создание средств защиты информации и средств контроля эффективности ее защиты;
- контроль за проведением работ по защите информации в органах государственного управления, объединениях, на предприятиях, в организациях и учреждениях (независимо от форм собственности).

Структура государственной системы защиты информации РФ

В соответствии с Законом Российской Федерации "О федеральных органах правительственной связи и информации", к основным функциям Федерального агентства правительственной связи и информации при Президенте Российской Федерации (ФАПСИ) в рассматриваемой области относится:

- осуществление координации деятельности по вопросам безопасности информационно-аналитических сетей, комплексов технических средств баз данных;
- осуществление координации деятельности в области разработки, производства и поставки *шифровальных средств и оборудования специальной связи*, по обеспечению криптографической и инженерно-технической безопасности шифрованной связи.

Федеральным законом от 03.04.95 г. N 40-ФЗ "*Об органах Федеральной службы безопасности в Российской Федерации*" к компетенции ФСБ в рассматриваемой области отнесены следующие вопросы:

- участие в разработке и реализации мер по защите сведений, составляющих государственную тайну;
- осуществление контроля за обеспечением сохранности сведений, составляющих государственную тайну, в государственных органах, воинских формированиях, на предприятиях, в учреждениях и организациях независимо от форм собственности;
- осуществление мер, связанных с допуском граждан к сведениям, составляющим государственную тайну.

3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПОДГОТОВКЕ К ЗАНЯТИЯМ

3.1 Лабораторные работы по теме «Информация и информационные процессы.

Представление информации. Информационно-логические основы построения персонального компьютера. Состав и структура ЭВМ и ПЭВМ. Программное обеспечение персонального компьютера»

При подготовке к занятию необходимо обратить внимание на следующие моменты.

Информатика как наука. Информация: определение, формы представления. Передача информации.

Сигналы и данные. Основные структуры данных. Файлы.

Виды информации. Свойства информации.

Количество информации (вероятностный и объёмный подходы). Единицы измерения информации.

Кодирование информации. Кодирование различных форм представления информации (числовой, текстовой).

Кодирование информации. Кодирование различных форм представления информации (графической, звуковой).

Качество информации.

Представление информации в компьютере. Позиционные и непозиционные системы счисления. Двоичный алфавит.

Двоичная, восьмеричная, шестнадцатеричная системы счисления. Перевод чисел из двоичной, восьмеричной, шестнадцатеричной систем счисления в десятичную и обратно.

Компьютер: определение, состав, минимальная и расширенная конфигурация.

Устройства ввода и вывода информации.

Процессор: назначение, устройство, важнейшие характеристики.

Структура памяти компьютера.

Внутренняя память.

Внешняя память. Основные носители информации и их важнейшие характеристики.
Форматирование диска.
Функциональная организация компьютера (магистрально-модульный принцип построения компьютера). Принцип Фон-Неймана.
Программное обеспечение компьютера. Классификация программного обеспечения.
Операционные системы. Загрузка операционной системы.
Операционная система MS DOS. Составные части MS DOS.
Начальная загрузка MS DOS. Работа с файлами и каталогами в MS DOS(команды).
История создания вычислительной техники.
Краткая история развития ЭВМ.
Важнейшие этапы компьютерной научно-технической революции.
Текстовый процессор WORD. Его возможности Создание простых текстовых документов. Установка параметров страницы, колонтитулы, нумерация страниц.
Текстовый процессор WORD Создание составных текстовых документов: работа с таблицами, диаграммы. Формулы, вычисляемые таблицы.
Текстовый процессор WORD Создание маркированных, нумерованных и многоуровневых списков. Создание оглавления.
Текстовый процессор WORD. Гиперссылки и макросы.
Электронные таблицы. Основные манипуляции с таблицами. Блоки, абсолютная и относительная адресация, запись чисел и формул в Excel.
MS Excel. Функции: математические; статистические; функции даты.
MS Excel Условные (логические) функции.
MS Excel. Сортировки данных. Фильтрация данных в списке. Промежуточные итоги.
MS Excel Сводные таблицы. Консолидация.

3.2.Лабораторные работы по теме «Основы алгоритмизации и программирования.

Базы данных. Модели решения задач. Компьютерные сети.»

При подготовки к занятию необходимо обратить внимание на следующие моменты.

Понятие алгоритма. Исполнитель алгоритма Свойства алгоритмов. Формы записи алгоритмов.
Виды алгоритмов: линейные; условные; циклические (понятие, форма записи, примеры).
Технологии программирования.
Языки программирования, их назначение, особенности.
Классификация языков программирования. Транслятор и компилятор.
Структурное программирование.
Основные понятия объектно-ориентированного программирования
Компьютерная графика. Виды компьютерной графики.
Моделирование. Информационные модели. Этапы построения компьютерной информационной модели.
Формы представления и структуры информационных моделей.
Основные понятия БД. Классификация.
СУБД Microsoft Access – основные возможности. Базовые объекты СУБД MS Access.
Типы данных в MS Access. Связи в БД. Целостность БД.
Запросы. Определение условий отбора.
Передача информации. Локальные компьютерные сети.
Классификация сетей. Сетевые топологии.

Глобальные сети. Основные возможности, предоставляемые сетью Интернет.
Технические и программные ресурсы Интернета.
Информационная система. Основные свойства информационных систем.
Лицензионные, условно бесплатные и бесплатные программы.
Защита информации. Криптографические средства защиты информации.

4. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ ИНДИВИДУАЛЬНЫХ ДОМАШНИХ ЗАДАНИЙ

Контрольная работа по информатике состоит из трех заданий направленных на освоение основных программ из пакета «Microsoft Office» - «Word»; «Excel»; «PowerPoint».

В результате выполнения контрольной работы студент должен:

получить представление:

- о развитии информационных технологий;
- о современных информационных технологиях сбора, хранения, кодирования, обработки и передачи информации;
- о фазах и моделях информационных процессов в автоматизированных системах;
- о технических и программных средствах информационных технологий.

уметь:

- записывать целые и вещественные числа в различных системах счисления,
- выполнять над ними арифметические операции,
- использовать системные оболочки и различные утилиты при работе на ПЭВМ,
- подготавливать, редактировать и оформлять текстовую документацию, графики, диаграммы и рисунки,
- обрабатывать числовые данные в электронных таблицах,
- использовать основные функциональные возможности сетевых информационных технологий,
- подготовить презентацию по заданной теме с использованием компьютерных технологий.

1 задание

Подготовить реферат на тему:

- развития информационных технологий,
- совершенствования аппаратных средств компьютерной техники,
- развития сетевых технологий;
- математические модели построения сетей;
- архитектура вычислительных сетей и т.д.

Реферат должен содержать 5-7 листов текста (формат листа А4, шрифт 14, интервал 1,5). В реферате должны быть рисунки, схемы, таблицы, формулы.

2 задание

В программе «Microsoft Excel» разработать программу по расчету:

- банковского кредита;
- заработной платы;
- какой – либо расчетной задачи.

Студент выбирает произвольную тему по решению математической задачи, показывает с помощью диаграммы динамические процессы разработанной задачи.

3 задание

Создать презентацию на тему своего реферата с использованием средств анимации и других средств программы «Microsoft PowerPoint».

Оформление контрольной работы осуществляется в соответствии с требованиями к контрольным работам, установленным в ОГАУ. К работе прикладывается электронный носитель с выполненными задачами.