Оглавление

[1) Базовые понятия теории информации: информация, сообщение, сигнал. Виды и свойства информации 2](#_Toc92713105)

[2) Дискретные и непрерывные сообщения. Алфавит сообщений. Сигналы и знаки. Классификация сигналов. Цифровой сигнал 3](#_Toc92713106)

[3) Количественная мера информации. Свойства количества информации 4](#_Toc92713107)

[4) Энтропия. Свойства энтропии. Энтропия объектов с дискретным и непрерывным множеством состояний. Среднее количество информации. 5](#_Toc92713108)

[5) Понятие канала связи. Информационные характеристики каналов связи. Скорость передачи и пропускная способность канала 6](#_Toc92713109)

[6) Цели и виды преобразования информации. Кодирование, модуляция 7](#_Toc92713110)

[7) Равномерное и неравномерное кодирование. Префиксные коды 8](#_Toc92713111)

[8) Метод Шеннона-Фано для сжатия при известной статистике сообщения 9](#_Toc92713112)

[9) Метод Хафмена для сжатия при известной статистике сообщения 9](#_Toc92713113)

[10) Арифметическое кодирование при известной статистике сообщения 10](#_Toc92713114)

[11) Метод Хафмена для сжатия при неизвестной статистике сообщения 11](#_Toc92713115)

[12) Словарные методы сжатия сообщения. Метод LZ77 12](#_Toc92713116)

[13) Словарные методы сжатия сообщения. Метод LZSS 13](#_Toc92713117)

[14) Словарные методы сжатия сообщения. Метод LZ78 14](#_Toc92713118)

[15) Особенности работы программ-архиваторов. Сжатие информации с потерями 15](#_Toc92713119)

[16) Информационный канал его составляющие и характеристики 17](#_Toc92713120)

[17) Способы кодирования двоичной информации 19](#_Toc92713121)

[18) Помехозащитное кодирование. Коды с обнаружением ошибок. Коды с исправлением ошибок 19](#_Toc92713122)

[19) Блочные коды. Избыточность кода. Расстояние Хэмминга. Вес слова 20](#_Toc92713123)

[20) Матричное помехозащитное кодирование 21](#_Toc92713124)

[21) Совершенные помехозащитные коды 22](#_Toc92713125)

[22) Квазисовершенные помехозащитные коды 23](#_Toc92713126)

[23) Полиномиальные помехозащитные коды 23](#_Toc92713127)

[24) Циклические избыточные коды 24](#_Toc92713128)

[25-26) Шифрование данных. Примеры простых методов шифрования. Шифр-перестановка и шифр-смещение 25](#_Toc92713129)

[27) Криптосистемы без передачи ключей 27](#_Toc92713130)

[28) Криптосистемы с открытым ключом 28](#_Toc92713131)

[29) Электронная подпись 29](#_Toc92713132)

[30) Системы передачи дискретных сообщений. Виды сигналов 31](#_Toc92713133)

[31) Виды каналов связи. Их характеристики. Фильтр нижних частот 32](#_Toc92713134)

[32) Способы передачи и обработки сигналов в системах ПДС 33](#_Toc92713135)

[33) Сигналы, используемые в технике ПДС. Свойства сигналов 35](#_Toc92713136)

[34) Амплитудная модуляция и ее характеристики. Схема амплитудного модулятора 36](#_Toc92713137)

[35) Частотная модуляция и ее характеристики. Схема частотного модулятора 38](#_Toc92713138)

[36) Фазовая модуляция и ее характеристики. Схема фазового модулятора 40](#_Toc92713139)

[37) Цифровая генерация и обработка сигналов в УПС 42](#_Toc92713140)

[38) Представление чисел в устройствах ЦОС. Эффекты квантования в устройствах ЦОС 48](#_Toc92713141)

[39) Основные понятия открытых систем 54](#_Toc92713142)

[40) Эталонная модель взаимодействия открытых систем 57](#_Toc92713143)

[41) Технология передачи информации в модели взаимодействия открытых систем 58](#_Toc92713144)

[42) Физический уровень взаимодействия открытых систем 59](#_Toc92713145)

[43) Канальный уровень взаимодействия открытых систем 61](#_Toc92713146)

[44) Сетевой уровень взаимодействия открытых систем 63](#_Toc92713147)

[45) Транспортный уровень взаимодействия открытых систем 64](#_Toc92713148)

[46) Сеансовый уровень взаимодействия открытых систем 67](#_Toc92713149)

[47) Представительный уровень взаимодействия открытых систем 69](#_Toc92713150)

[48) Прикладной уровень взаимодействия открытых систем 70](#_Toc92713151)

[49) Технология поиска информации в Интернет. Поисковые инструменты 72](#_Toc92713152)

[50) Технология поиска информации в Интернет. Поисковые машины 73](#_Toc92713153)

[51) Технология поиска информации в Интернет. Каталоги, подборки ссылок 74](#_Toc92713154)

[52) Технология поиска информации в Интернет. Базы данных адресов 75](#_Toc92713155)

[53) Технология поиска информации в Интернет. Поиск в архивах Gopher и система поиска FTP файлов 75](#_Toc92713156)

[54) Технология поиска информации в Интернет. Система поиска в конференциях, системы мета-поиска, системы поиска людей 76](#_Toc92713157)

# 1) Базовые понятия теории информации: информация, сообщение, сигнал. Виды и свойства информации

*Информация* – это:

* нематериальная сущность, при помощи которой можно с любой точностью описывать реальные, виртуальные и понятийные сущности;
* сведения, являющиеся объектом передачи, преобразования, хранения и использования;
* противоположность неопределённости.

*Сообщение* – это форма представления информации (зрительная, звуковая, вербальная, тактильная и т.д.).Могут быть непрерывными (принимают любые значения из определённого интервала) и дискретными (состоят из конечного набора возможных значений).

*Сигнал* – это физический процесс отображения сообщения.

*Виды информации:*

1. дискретная (цифровая) – характеризуется последовательными точными значениями некоторой величины;
2. непрерывная (аналоговая) – характеризуется непрерывным процессом изменения некоторой величины.

Также различают виды инф-и по способу передачи и по восприятию. Это:

* визуальная (передаётся видимыми образами, символами)
* аудиальная (передаётся звуками)
* машинная (выдаётся\воспринимается ЭВМ)
* текстовая
* числовая и т.д.

*Свойства информации:*

1. достоверность(отражает истинное положение дел)
2. полнота(ее достаточно для понимания и принятия решений)
3. актуальность (важность при работе в постоянно изменяющихся условиях)
4. ценность(определяется по тем задачам, которые можно решить с ее помощью).

# 2) Дискретные и непрерывные сообщения. Алфавит сообщений. Сигналы и знаки. Классификация сигналов. Цифровой сигнал

*Сообщение* – это форма представления информации. Могут быть непрерывными (принимают любые значения из определённого интервала) и дискретными (состоят из конечного набора возможных значений).

*Алфавит сообщения* – это совокупность символов, используемых для записи информации.

*Сигнал* – это физический процесс отображения сообщения.

Сигнал характеризуется своим уровнем и моментами изменения данного уровня. Типы сигналов:

1. непрерывные сигналы непрерывного времени (непрерывные или аналоговые) – уровни сигнала могут изменяться в любой момент времени и любым образом (принимать любые значения);
2. непрерывные сигналы дискретного времени – могут принимать произвольные значения, но изменяются только в определённые моменты времени;
3. дискретные сигналы непрерывного времени – могут изменяться в произвольные моменты, но принимают только разрешённые дискретные значения;
4. дискретные сигналы дискретного времени (дискретные) – уровни меняются только в дискретные моменты и принимают только дискретные значения.

Сигналы, формируемые на выходе преобразователя дискретного сообщения в сигнал, являются по информационному параметру дискретными. Это *цифровые сигналы данных (ЦСД).*

*Элемент ЦСД* – часть цифрового сигнала, отличающаяся от других частей значением одного из инф. параметров.

*Значащая позиция* – это фиксируемое значение состояния инф. параметра.

*Значащий момент* – это момент, в кот.происходит смена значений сигнала.

*Значащие интервалы* – это интервалы времени между двумя соседними зн.моментами. Минимальный интервал τ0 наз. единичным.

Сигналы могут быть:

* изохронными (любой ЗИ равен τ0 или целому их числу)
* анизохронными (произвольная длительность, но не менее чем τ0)

# 3) Количественная мера информации. Свойства количества информации

Если производится измерение с некоторой погрешностью, то чем больше произведено измерений, тем больше информации об измеряемом объекте.

Пусть передатчик передаёт сообщение, ассоциируемое с СВX. В канале связи накладывается аддитивный шум Z. Т.е. на принимающей стороне получаютY=X+Z. В этой схеме можно говорить о количестве информации, содержащейся в СВY по сравнению с СВ X. Чем ниже уровень помех (дисперсия Z мала), тем больше информации можно получить из Y.

Мера измерения количества инф-и, которую несёт в себе элемент :

Пусть CВX и Y заданы законами распределения P(X=xi)=pi, P(Y=yj)=qj, где ; j=, и совместным распределениемP(x=xi, y=yj)=pij.

Оценка количества информации, содерж. В X относительно Y, определяется как:

Для непрерывных СВXиY,заданных плотностями распреде­ления вероятностей*,*и*,* аналогичная формула имеет вид:

Свойства меры информации:

1) *I(X, Y) ≥ 0*,*I(X, Y) = 0* если X и Y независимы;

2) *I(X,Y)= I(Y,X);*

3) *I(X, Y) = HX + HY - H(X, Y)*, где*H(X, Y) = -*

4) *I(X, Y) ≤I(X, X)*. Если *I(X, Y) = I(X, X),* то X — функция от Y.

# 4) Энтропия. Свойства энтропии. Энтропия объектов с дискретным и непрерывным множеством состояний. Среднее количество информации.

**Энтропия. Свойства энтропии**

ПустьCВ X и Y заданы законами распределения P(X=xi)=pi, P(Y=yj)=qj,где ; j=, и совместным распределениемP(x=xi, y=yj)=pij.Оценка количества информации, содерж. В X относительно Y, определяется как:

Отсюда можно получить оценку I(X,X) –инф-ю сообщения о себе самом:



=

Данная величина наз. энтропией и обозначается H(X).

*Энтропия ДСВ* – это минимум количества бит, которые необходимо передать по каналам связи о текущем значении данной ДСВ.

Свойства энтропии:

1) *H(X) = 0* ->X — константа;

2) *I(X, Y) = HX + HY - H(X, Y)*, где*H(X, Y) = -*

3) *I(X, Y) ≤I(X, X)*. Если *I(X, Y) = I(X, X),* то X — функция от Y.

**Энтропия объектов с дискретным и непрерывным множеством состояний. Среднее количество информации**

а) Для непрерывных случайных величин, заданных плотностями распреде­ления вероятностей *,*и :

б) Для ДСВ, заданных законами распределения P(X=xi)=pi, P(Y=yj)=qjи совместным распределениемP(x=xi, y=yj)=pij:



Тогда энтропия: =

Энтропия также определяется как *среднее количество информации*, приходящееся на одно сообщение источника, вырабатывающего независимые сообщения.Чем больше энтропия, тем большую информацию в среднем несет одно сообщение источника.

# 5) Понятие канала связи. Информационные характеристики каналов связи. Скорость передачи и пропускная способность канала

*Канал связи* – это среда и средства передачи информации.

При передаче информации в канале связи могут возникнуть помехи – шумы (гармонические, импульсные, флуктуационные, мультипликативные).

*Информационный канал* — это совокупность устройств, объединенных линиями связи, предназначенных для передачи инф-ии от источника инф-ии (начальногоустр-ва канала) до ее приемника (конечного устр-ва канала).

Линии связи обеспечивают прохождение инф. сигналов между устр-вами канала. Информация обычно передается при помощи эл. тока (по проводам), света (по оптоволокну), э/м волн радиодиапазона (в пространстве), звука (в плотной среде: атмосфере, воде и т.п.) и прочих.

Устройства канала — это, как правило, репитеры, передающие усиленным принятый сигнал.

Технические характеристики канала определяются принципом действия входящих в него устр-в, видом сигнала, св-вами и составом физической среды, в которой распространяются сигналы, св-вами применяемого кода.

Эффективность канала хар-тся скоростью и достоверностью передачи информации, надежностью работы устр-в и задержкой сигнала во времени.Задержка сигнала во времени — это интервал времени от отправки сигнала передатчиком до его приема приемником. Достоверность и надёжность напрямую связана с наличием шумов.

В соответствии со структурой вх. и вых. сигналов выделяют каналы:

* *дискретные* (сигналы на входе и выходе представляют собой последовательность символов некоторых алфавитов);
* *непрерывные* (входные и выходные сигналы представляют собой функции от непрерывного параметра (времени)).

Математической моделью шумов является совокупность условных вероятностей: ->канал без шумов

Способность канала передавать информацию характеризуется *ёмкостью* или *пропускной способностью канала*:

N (T) — число всех возможных сигналов

Для вычисления максимальной скорости передачи в канале с шумом при известной ёмкости канала и энтропии источника используют теорему Шеннона: .

# 6) Цели и виды преобразования информации. Кодирование, модуляция

*Преобразование информации -*изменение формы и вида сведений*.*Среди видов преобразования информации - кодирование и модуляция.

*Кодирование* – преобразование информации с одной из следующих целей: шифрование, сжатие, защита от шума.

Цель сжатия — уменьшение количества бит, необходимых для хранения или передачи заданной информации. Это дает возможность передавать сообщения более быстро и хранить более экономно и оперативно (операция извлечения данной инф-ии с устр-ва ее хранения будет проходить быстрее).

Помехозащитное кодирование имеет целью борьбу с шумами.

Цель шифрования – сделать сообщение непонятным для посторонних лиц.

*Модуляция -*  процесс изменения одного или нескольких параметров высокочастотного несущего колебания по закону низкочастотного информационного сигнала (сообщения).В результате модуляции спектр низкочастотного управляющего сигнала переносится в область высоких частот. По каналам связи передаются, как правило, модулированные сигналы. Модуляция позволяет решить проблему согласования частот.

Модуляции делятся на:  
• модуляции гармонических колебаний (амплитудная, частотная, фазовая);  
• импульсные модуляции.

# 7) Равномерное и неравномерное кодирование. Префиксные коды

*Кодирование* – преобразование информации с одной из следующих целей: шифрование, сжатие, защита от шума.

*Алфавит кода* – это некое множество А, состоящее из d элементовA={a1,a2,…ad}, например A={0,1}.*Кодовое слово* – последовательность n символов из множества A.*Код* – множество кодовых слов.

*Равномерным* наз. код, у которого все кодовые слова имеют одинаковую длину. Если хотя бы одно кодовое слово имеет отличную от других длину, то код *неравномерный*. Равномерным кодом является таблица ASCII.

Размерность равномерного кода:.

Эффективность кода определяется по скорости кодирования:

, где *l*–кол-во символов, кодируемых одним элементом.

При *неравномерном кодировании* коды определяются с учётом вероятности появления каждого символа в сообщении.

*Префиксным* наз. код, в котором ни одно более короткое слово не является началом другого, более длинного.

# 8) Метод Шеннона-Фано для сжатия при известной статистике сообщения

Пусть известна статистика сообщения, т.е. .

Алгоритм построения кода:

1. Упорядочить множество возможных значений сообщения по убыванию вероятностей;
2. Данное множество разделить на d подмножеств так, чтобы сумма вероятностей каждого подмножества была примерно одинаковой;
3. Пронумеровать полученные подмножества символами из алфавита A;
4. Шаги 2 и 3 повторять до тех пор, пока не останется 1 элемент сообщения.

Код считывается слева направо.

# 9) Метод Хафмена для сжатия при известной статистике сообщения

Пусть известна статистика сообщения, т.е. .

Алгоритм построения кода:

1. Упорядочить множество возможных значений сообщения по убыванию вероятностей;
2. d самых маловероятных объединить в 1 (d=2, т.к. A={0,1}) и вычислить их суммарную вероятность;
3. Каждому из объединённых эл-тов поставить в соответствие символ из алфавита А (0 или1);
4. Повторять, пока не останется 1 элемент.

Код считывается справа налево.

По сравнению с методом Шеннона-Фано, метод Хафмена более практичен и никогда не уступает ему по степени сжатия.

# 10) Арифметическое кодирование при известной статистике сообщения

Одной из лучших схем кодирования, позволяющих кодировать некоторые символы менее чем 1 битом, является арифметическое кодирование, появлявшееся в 70-х гг. 20 в.

По исходному распределению вероятностей для выбранной для кодирования д. с. в. строится таблица, состоящая из пересекающихся только в граничных точках отрезков для каждого из значений этой д.с.в.; объединение этих отрезков должно образовывать отрезок [0,1], а их длины должны быть пропорциональны вероятностям соответству­ющих значений д. с. в.

Алгоритм кодирования заключается в постро­ении отрезка, однозначно определяющего данную последовательность значений д.с.в.

Затем для построенного отрезка находится число, при­надлежащее его внутренней части и равное целому числу, деленному на минимально возможную положительную целую степень двойки. Это число и будет кодом для рассматриваемой последовательности. Все воз­можные конкретные коды - о числа строго большие нуля и строго меньшие одного, поэтому можно отбрасывать лидирующий ноль и де­сятичную точку, но нужен еще один специальный код-маркер, сигна­лизирующий о конце сообщения. Отрезки строятся так. Если имеется отрезок для сообщения длины (n-1), то для построения отрезка для сооб­щения длины n, разбиваем его на столько же частей, сколько значений имеет рассматриваемая д.с.в. Это разбиение делается совершенно так­же как и самое первое (с сохранением порядка). Затем выбирается из полученных отрезков тот, который соответствует заданной конкретной последовательности длины n.

Принципиальное отличие этого кодирования от рассмотренных ра­нее методов в его непрерывности, т. е. в ненужности блокирования. Эффективность арифметического кодирования растет с ростом длины сжимаемого сообщения (для кодирования Хаффмена или Шеннона-Фэно этого не происходит). Хотя арифметическое кодирование дает обычно лучшее сжатие, чем кодирование Хаффмена, оно пока используется на практике сравнительно редко, т.к. оно появилось гораздо позже и тре­бует больших вычислительных ресурсов.

При сжатии заданных данных, например, из файла все рассмотрен­ные методы требуют двух проходов. Первый для сбора частот симво­лов, используемых как приближенные значения вероятностей символов, и второй для собственно сжатия.

Получение исходного сообщения из его арифметического кода про­исходит по следующему алгоритму.

Шаг 1. В таблице для кодирования значений д.св. определяется ин­тервал, содержащий текущий код - по этому интервалу одно­значно определяется один символ исходного сообщения. Если этот символ - это маркер конца сообщения, то конец.

Шаг 2. Из текущего кода вычитается нижняя граница содержащего его интервала, полученная разность делится на длину этого же интервала. Полученное число считается но|вым текущим значением кода. Переход к шагу 1.

# 11) Метод Хафмена для сжатия при неизвестной статистике сообщения

Адаптивный алгоритм сжатия Хаффмена является практичным, однопроходным, не требующим передачи таблицы кодов. Его суть в использовании адаптивного алгоритма, т.е. алгоритма, который при каждом сопоставлении символу кода, кроме того, изменяет внутренний ход вычислений так, что в следующий раз этому же символу может быть сопоставлен другой код, т.е. происходит адаптация алгоритма к поступающим для кодирования символам. При декодировании происходит аналогичный процесс.

В начале работы алгоритма дерево кодирования содержит толь­ко один специальный символ, всегда имеющий частоту 0. Он необхо­дим для занесения в дерево новых символов: после него код символа передается непосредственно. Обычно такой символ называют escape-символом (ESC). При построении дерева кодиро­вания необходимо для возможности правильного декодирования как-то упорядочивать структуру дерева. Расположим листья дерева в порядке возрастания частот и затем в порядке возрастания стандартных кодов символов. Узлы собираются слева направо без пропусков. Левые ветви помечаются 0, а правые - 1.

# 12) Словарные методы сжатия сообщения. Метод LZ77

Алгоритм LZ77 был опубликован в 1977 г. Разработан израильскими математиком Якобом Зивом и Авраамом Лемпелом.

Одной из причин популярности алгоритмов LZ77 является их исключительная простота при высокой эффективности сжатия.

Основная идея LZ77 состоит в том, что второе и последующее вхождения некоторой строки символов в сообщении заменяются ссылками на её первое вхождение.

LZ77 использует уже просмотренную часть сообщения как словарь. Чтобы добиться сжатия, он пытается заменить очередной фрагмент сообщения на указатель в содержимое словаря. LZ77 использует скользящее по сообщению окно, разделённое на две неравные части. Первая, большая, словарь, включает уже просмотренную часть сообщения. Вторая, намного меньшая, является буфером, содержащим ещё незакодированные символы входного потока. Алгоритм пытается найти в словаре фрагмент, совпадающий с содержимым буфера.

Алгоритм LZ77 выдаёт коды, состоящие из трёх элементов:

1) смещение в словаре относительно его начала подстроки, совпадающей с началом содержимого буфера;

2) длина этой подстроки;

3) первый символ буфера, следующий за подстрокой.

Недостатки LZ77:

1) с ростом словаря скорость алгоритма-кодера пропорционально замедляется;

2) кодирование одиночных символов очень неэффективно.

# 13) Словарные методы сжатия сообщения. Метод LZSS

В 1982 г. Сторером и Шиманским на базе LZ77 был разработан алгоритм LZSS, который отличается от LZ77 производимыми кодами.

Основная идея LZ77 состоит в том, что второе и последующее вхождения некоторой строки символов в сообщении заменяются ссылками на её первое вхождение.

LZ77 использует уже просмотренную часть сообщения как словарь. Чтобы добиться сжатия, он пытается заменить очередной фрагмент сообщения на указатель в содержимое словаря. LZ77 использует скользящее по сообщению окно, разделённое на две неравные части. Первая, большая, словарь, включает уже просмотренную часть сообщения. Вторая, намного меньшая, является буфером, содержащим ещё незакодированные символы входного потока. Алгоритм пытается найти в словаре фрагмент, совпадающий с содержимым буфера.

Код, выдаваемый LZSS, начинается с однобитного префикса, различающего собственно код от незакодированного символа. Код состоит из пары: смещение и длина, такими же как и для LZ77. В LZSS окно сдвигается ровно на длину найденной подстроки или на 1, если не найдено вхождение подстроки из буфера в словарь. Длина подстроки в LZSS всегда больше нуля, поэтому длина двоичного кода для длины подстроки – это округлённый до большего целого двоичный логарифм от длины буфера.

LZ77 и LZSS обладают следующими недостатками:

1) невозможность кодирования подстрок, отстоящих друг от друга на расстоянии, большем длины словаря;

2) длина подстроки, которую можно закодировать, ограничена размером буфера.

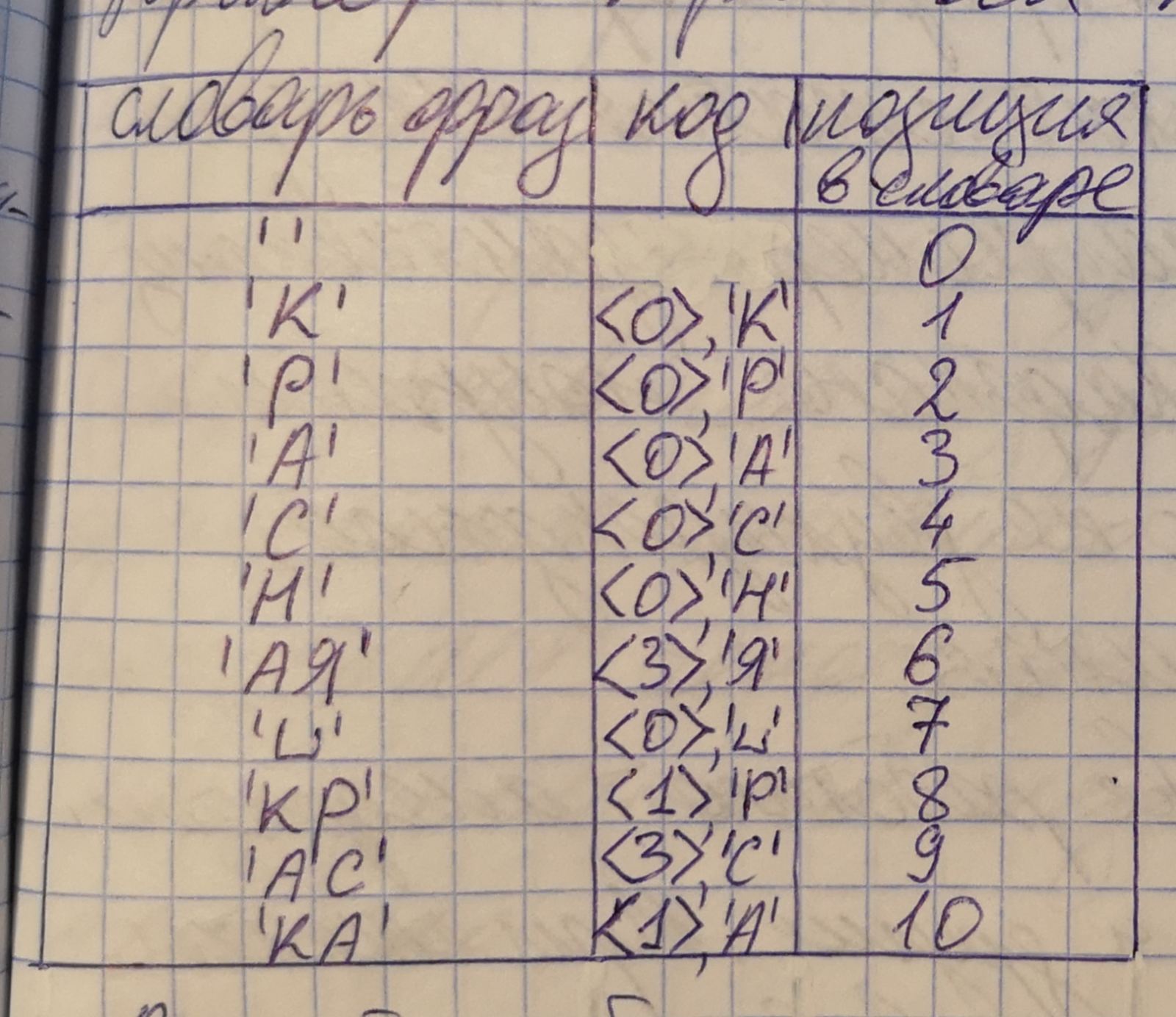
Если механически увеличивать размеры словаря и буфера, то это приведёт к снижению эффективности кодирования, т.к. с ростом этих величин будут расти и длины кодов для смещения и длины, что сделает коды для коротких подстрок недопустимо большими. Кроме того, резко увеличится время работы алгоритма-кодера.

14) Словарные методы сжатия сообщения. Метод LZ78

Авторы LZ77 в 1978 году предложили алгоритм LZ78.

LZ78 – не имеет скользящего окна. Он хранит словарь из просмотренных фраз. В начале этот словарь содержит одну пустую строку. Алгоритм считывает символы сообщения до тех пор, пока эта подстрока входит целиком в одну из фраз словаря. Как только строка перестанет соответствовать хотя бы одной из фраз словаря, алгоритм генерирует код, состоящий из индекса строки в словаре и символа, нарушившего совпадение. И новая подстрока добавляется в словарь. Если словарь заполнен, из него удаляется фраза с наименьшей частотой применения.

Пример: «Красная краска»



Пусть словарь содержит до 16 фраз (от 0 до 15). Для кодирования номера достаточно 4 бита. В файл было передано 10 элементов. Длина кода сообщения: 120 бит. Декодирование сообщения закодировано LZ78.

Строится словарь фраз, заданной длины. Изначально словарь имеет одну пустую строку в позиции 0. Считывается блок битов длины равной длине кода и длине символа. Число равное длине кода определяет номер фразы в словаре, а код символа определяет очередной элемент фразы. Вновь полученная фраза добавляется в словарь.

15) Особенности работы программ-архиваторов. Сжатие информации с потерями

Если коды алгоритмов типа LZ передать для кодирования (адап­тивному) алгоритму Хаффмена или арифметическому, то полученный двухшаговый (конвейерный, а не двухпроходный) алгоритм даст ре­зультаты сжатия подобные широко известным программам: GZIP, ARJ, PKZIP.

Наибольшую степень сжатия дают двухпроходные алгоритмы, ко­торые исходные данные последовательно сжимают два раза, но они работают до двух раз медленнее однопроходных при незначительном увеличении степени сжатия.

Большинство программ-архиваторов сжимает каждый файл по от­дельности, но некоторые сжимают файлы в общем потоке, что дает уве­личение степени сжатия, но одновременно усложняет способы работы с полученным архивом, например, замена в таком архиве файла на его более новую версию может потребовать перекодирования всего архива. Примером программы, имеющей возможность сжимать файлы в общем потоке, является RAR.

Сжатие RLE (кодирование переменной длины) - это простейший метод сжатия, в общем случае очень неэффективный, но дающий неплохие результаты на типичной графической информации. Оно основано в основном на выделении специального кода-маркера, указывающего сколько раз повторить следующий байт.

Сжатие и распаковка в реальном времени используется в программах-драйверах для "уплотнения" носителей информации, позволяющих увеличить емкость носителя приблизительно в 2 раза. Наиболее известной программой такого рода является DriverSpace для MS-DOS и Microsoft Windows.

Все ранее рассмотренные алгоритмы сжатия информации обеспечивали возможность полного восстановления исходных данных. Но иногда для повышения степени сжатия можно отбрасывать часть исходной информации, т. е. производить сжатие с потерями. В тех случаях, когда сжимается информация, используемая лишь для качественной оценки (это, как правило, аналоговая информация), сжатие с потерями является очень подходящим.

Сжатие с потерями используется в основном для трех видов данных: полноцветная графика (224 ~ 16 млн. цветов), звук и видеоинформация.

Сжатие с потерями обычно проходит в два этапа. На первом из них исходная информация приводится (с потерями) к виду, в котором ее можно эффективно сжимать алгоритмами 2-го этапа сжатия без потерь.

Основная идея сжатия графической информации с потерями заключается в следующем. Каждая точка в картинке характеризуется тремя равноважными атрибутами: яркостью, цветом и насыщенностью. Но глаз человека воспринимает эти атрибуты не как равные. Глаз воспринимает полностью только информацию о яркости и в гораздо меньшей степени о цвете и насыщенности, что позволяет отбрасывать часть информации о двух последних атрибутах без потери качества изображения.

Для сжатия графической информации с потерями в конце 1980-х установлен стандарт - формат JPEG. В этом формате можно регулировать степень сжатия, задавая степень потери качества.

Сжатие видеоинформации основано на том, что при переходе от одного кадра фильма к другому на экране обычно почти ничего не ме­няется. Таким образом, сжатая видеоинформация представляет собой запись некоторых базовых кадров и последовательности изменений в них. При этом часть информации может отбрасываться. Сжатую подоб­ным образом информацию можно далее сжимать и другими методами. Наиболее распространенными являются стандарты MPEG. Видеоинформацию можно сжать до 100 и более раз

Для сжатии звуковой информации с потерями существует несколь­ко стандартов. Наиболее широко используемый из них - MPEG без видеоданных. Стандарт LPC использует­ся для сжатия речи. Алгоритм LPC пытается промоделировать рече­вой тракт человека и выдает на выходе буквально текущее состояние участвующих в формировании звуков органов.

# 16) Информационный канал его составляющие и характеристики

Канал информационный - это совокупность устройств, объеди­ненных линиями связи, предназначенных для передачи информации от источника информации (начального устройства канала) до ее прием­ника (конечного устройства канала).

Линии связи обеспечивают прохождение информационных сигна­лов между устройствами канала. Информация обычно передается при помощи электрического тока (по проводам), света (по оптоволокну), электромагнитных волн радиодиапазона (в пространстве) и, редко, зву­ка (в плотной среде: атмосфере, воде и т.п.) и прочих.

Устройства канала это, как правило, репитеры, просто пере­дающие усиленным принятый сигнал (пример, радиорелейные линии). К устройствам канала иногда относят и кодеры/декодеры, но в только тех случаях, когда кодирование/декодирование происходит с высокой скоростью, не требующей ее специального учета, как замедляющего фактора; обычно же кодеры/декодеры относят к источникам или при­емникам информации.

Технические характеристики канала определяются принципом действия входящих в него устройств, видом сигнала, свойствами и со­ставом физической среды, в которой распространяются сигналы, свой­ствами применяемого кода.

Эффективность канала характеризуется скоростью и достоверно­стью передачи информации, надежностью работы устройств и задерж­кой сигнала во времени.

Задержка сигнала во времени - это интервал времени от отправ­ки сигнала передатчиком до его приема приемником.

Математически канал задается множеством допустимых сообще­ний на входе, множеством допустимых сообщений на выходе и набо­ром условных вероятностей Р(у/х) получения сигнала у на выходе при входном сигнале х. Условные вероятности описывают статистические свойства "шумов" (или помех), искажающих сигнал в процессе пере­дачи. В случае, когда Р(у/х) = 1 при у = х и Р(у/х) = 0 при у≠х, канал называется каналом без "шумов ". В соответствии со структурой входных и выходных сигналов выделяют дискретные и непрерывные каналы. В дискретных каналах сигналы на входе и выходе представля­ют собой последовательность символов одного или двух (по одному для входа и выхода) алфавитов. В непрерывных каналах входной и выход­ной сигналы представляют собой функции от непрерывного параметра- времени. Бывают также смешанные или гибридные каналы, но тогда обычно рассматривают их дискретные и непрерывные компоненты раздельно. Далее рассматриваются только дискретные каналы.

Способность канала передавать информацию характеризуется числом - пропускной способностью или емкостью канала (обозначение - с).

Для случая канала без шума формула расчета емкости канала имеет вид С = limT→∞(log2 N(T)/T) число всех возможных сигналов за время Т.

17) Способы кодирования двоичной информации

Для кодирования используется уровень сигнала: низкий для 0 и высокий для 1. Недостатки этого способа проявляются в случаях, когда нужно передавать много сплошных нулей или единиц. Малейшее рассогласование синхронизации между приемником и передатчи­ком приводит тогда к неисправимым ошибкам. Кроме того, многие но­сители информации, в частности, магнитные, не могут поддерживать длительный постоянный уровень сигнала.

Для передачи информации используется обычно другой способ, ко­гда для представления 0 и 1 используются две разные частоты, от­личающиеся друг от друга ровно в два раза - это так называемая частотная модуляция. Таким образом, при таком кодировании, если сигнал 1 имеет дли­тельность τ, то 0 - 2 τ.

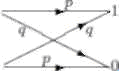
При использовании частотной модуляции на практике нули, как правило, кодируются в два раза плотнее. Это достигается тем, что учитываются не уровни сигнала, а смена уровня (полярности). Если частота v соответствует 1, то с частотой 2 v производится проверка уровня сигнала. Если он меняется, то это сигнал 1, если нет, то 0. На практике частота v - это частота синхронизации, т.е. частота импульса, который независимо от данных меняет полярность сигнала. 0 не генерирует импульса смены полярности, а 1 генерирует.

При необходимости передачи записанных с помощью некоторого кода сообщений по данному каналу приходиться преобразовывать эти сообщения в допустимые сигналы канала, т. е. производить надлежа­щее кодирование, а при приеме данных - декодирование. Кодирование целесообразно производить так, чтобы среднее время, затрачиваемое на передачу, было как можно меньше. Исходному входно­му алфавиту нужно однозначно сопоставить новый алфавит, обеспечи­вающий большую скорость передачи.

18) Помехозащитное кодирование. Коды с обнаружением ошибок. Коды с исправлением ошибок

Простейший код для борьбы с шумом - это контроль четности. Кодирование заключается в добавлении к каждому байту девятого бита таким образом, чтобы дополнить количество единиц в байте до заранее выбранного для кода четного или нечетного значения. Используя этот код, можно лишь обнаруживать большинство ошибок.

Простейший код, исправляющий ошибки, - это тройное повторение каждого бита. Если с ошибкой произойдет передача одного бита из трех, то ошибка будет исправлена, но если случится двойная или тройная ошибка, то будут получены неправильные данные. Часто коды для исправления ошибок используют совместно с кодами для обнаружения ошибок. При тройном повторении для повышения надежности три бита располагают не подряд, а на фиксированном расстоянии друг от друга. Использование тройного повторения значительно снижает скорость передачи данных.



Двоичный симметричный канал : р это вероятность безошибочной передачи бита, a q вероятность передачи бита с ошибкой. Предполагается, что в таком канале ошибки происходят независимо.

Двоичный симметричный канал реализует схему Бернулли. поэтому вероятность передачи nбит по двоичному симметричному каналу с kошибками равна Рn(k) = Cknpn-kqk.

Добиться минимальности вероятности ошибки при передаче данных можно используя специальные коды. Обычно используют систематические помехозащитные коды. Идея систематических кодов состоит в добавлении к символам исходных кодов, предназначенных для передачи в канале, нескольких контрольных символов по определенной схеме кодирования. Принятая такая удлиненная последовательность кодов декодируется по схеме декодирования в первоначально переданную. Приемник способен распознавать и/или исправлять ошибки, вызванные шумом, анализируя дополнительную информацию, содержащуюся в удлиненных кодах.

19) Блочные коды. Избыточность кода. Расстояние Хэмминга. Вес слова

По определению, блочный код заменяет каждый блок из m символов более длинным блоком из n символов. Следовательно, (m, n)-коды являются блочными. Существуют также древовидные или последовательные коды, в которых значение очередного контрольного символа зависит от всего предшествующего фрагмента сообщения. Ра­бота с древовидным шумозащитным кодом имеет сходство с работой с арифметическим кодом для сжатия информации.

Расстоянием Хэмминга между двоичными словами длины n на­зывается количество позиций, в которых эти слова различаются. Это одно из ключевых понятий теории кодирования. Если обозначить дво­ичные слова как а = а1…anи b = b1… bn - то расстояние между ними обозначается d(a,b).

Весом двоичного слова а = а1…anназывается количество единиц в нем. Обозначение w(a). Можно сказать, что w(a)=∑i=1nаi.

20) Матричное помехозащитное кодирование

Матричное *кодирование.* Для создания кодов, определяющих или исправляющих ошибки, необходимо создание кодирующих таблиц. Для хранения слов большой длины требуется много памяти. Например, для (16, 33) – кода потребуется 33\*216 = 2162688 бит. Составление таких таблиц является сложным процессом. Для автоматизации процесса кодирования можно использовать матричное кодирование, т.е. в качестве функции E используется матрица E [m x n]которая преобразует кодовые слова длиной n в длину m и наоборот.

Матрица состоит из eij={0,1}

Кодирование реализуется операцией b =aE, т.е bj=a1e1j(+)a2e2j(+)…(+)amemj; j=1,n

Рассмотрим пример. матрица [3 x 6].

E=

000→0000000 011→011100

001→001111 100→100110

010→010011 101→101001

111→111010 110→110101

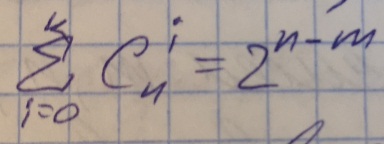
Преимущество матричного кодирования состоит в том, что надо запомнить m кодовых слов вместо 2m слов. Чтобы одно и то же кодовое слово не приписывалось различным сообщениям необходимо, чтобы m столбцов в матрице E образовывали единичную матрицу (3 первых столбца). При умножении любого вектора на единичную матрицу получается тот же самый вектор, поэтому вектора, отображающие различные сообщения будут соответствовать разные вектора систематического кода.

Матричные коды называют линейными кодами. Для линейных (n-r, n) кодов с минимальным расстоянием Хэмминга d существует нижняя граница Плоткина для минимального количества среднего количества n. R>=2d-2-Log2d.

21) Совершенные помехозащитные коды

Групповой (m,n) – код, исправляющий все ошибки веса не более k и никаких других – называется совершенным.

Свойства совершенного кода:

1. Для совершенного (m,n) кода, исправляющий ошибки не больше k, выполняется соотношение  верно и обратное.
2. Совершенный код, исправляющий все ошибки не более k, в столбцах таблицы декодирования содержит все слова отстоящие от кодовых на расстояние не больше, чем k. Верно и обратное.
3. Таблица декодирования совершенного кода, исправляющего все ошибки весом не более, чем k, имеет в качестве лидеров все строки не более, чем k единиц.

Совершенный код – лучший код, обеспечивающий максимум минимального расстояния между кодовыми словами, при минимуме длины кодовых слов.

Декодирование – каждому полученному слову однозначно ставится в соответствие ближайшее кодовое. Для чисел m,n,k, где (1<k<(n-1)/2) кодов, удовлетворяющих условию, совершенство очень мало. Совершенный код строится только в исключительных случаях. Если m,n,k не удовлетворяют условию совершенности, то лучший групповой код, который им соответствует – называют квазисовершенным, если он исправляет все ошибки кратности не более k и некоторые k+1.

Двоичный m,n код – называют оптимальным, если он минимизирует вероятность ошибочного декодирования. Совершенные и квазисовершенные коды – оптимальные.

Для любого целого, положительного числа r существует совершенный (m,n) код, исправляющий 1-у ошибку, которую называют кодом Хэмминга. В случае, когда m=2r-r-1; n=2r-1.

22) Квазисовершенные помехозащитные коды

Квазисовершенные (m,n) коды, исправляющие одну ошибку – строятся следующим образом:

Для заданного m – выбирается минимальное n, так, чтобы

. Каждое слово будет содержать r=n-m контрольных разрядов. Каждому из n разрядов слева направо присваивается номер от 1 до n.

Для заданного слова сообщения составляется r контрольных сумм S1…Sr, которые помещаются в позиции, номер которых равен n2, как и для совершенных кодов, т.е. остальные позиции содержат биты исходного слова: a=a1…am→b=S1S2a1S3a2a3a4S4a5…am.

Значения контрольных битов S1S2… вычисляется аналогично, вычислению контрольных битов для совершенных кодов.

Если m в исходном слове принадлежит множеству 1,4,11,26,53…, то в результате получим совершенный код. Можно построить код (8,4), т.к.

.

23) Полиномиальные помехозащитные коды

При полиномиальном кодировании каждое сообщение отождествляется с многочленом, а кодирование состоит в умножении на фиксированный многочлен. Полиномиальный код является блочным. Отличие состоит в алгоритме кодирования и декодирования. Пусть исходное сообщение a. Сопоставим ему многочлен a(x)=a0+a1x+a2x\*x+…+am-1\*x^m-1. Все вычисления происходят в поле классов вычетов по модулю 2. Зафиксируем многочлен по степени k g(x)=g0+g1x+…+gkx^k. g0<>0. Gk<>0.

Процедура кодирования преобразует слово a в слово b при помощи операции b(x)=a(x)\*g(x)=b0+b1x+…+bn-1x^n-1, которому соответствует кодовое слово b. Рассмотрим кодирующий элемент g(x)=1+x^2+x^3. A=01011. A(x)=x+x^3+x^4. B(x)=g(x)a(x)=x+x^5+x^7.

B=01000101 – кодовое слово, полученное в рез-те.

Пр. (3, 6) – код х3+х+1

G= 000 -> 000000 001->001101 010->011010

Полиномиальные коды явл. групповыми.

Пусть исходному сообщению соответствует полином b(x)=a(x)g(x), где g(x) – кодирующий полином

C(x)=b(x)+e(x), где e(x) – соответствует строка ошибок е(х)=е0+е1х+e2x2…

Для декодирования необходимо поделить c(x) на g(x).

c(x)/g(x), где c(x) = a(x)g(x)+e(х)

деление без остатка возможно лишь в том случае, если е(х) делится на g(х). в противном случае остается ненулевой остаток, т.е. любая ошибка, многочлен которой не делится на g(x) будет обнаружена. Если кодирующий многочлен g(x) порождающий соответствующий (m, n) – код, не явл. делителем ни одного из многочленов вида xj+1 для j<n, то минимальным расстоянием м/ду кодовыми словами не меньше 3-х, d>=3.

24) Циклические избыточные коды

*CRC*. Имеют фиксированную длину и используются для обнаружения ошибок. Наибольшее распространение получили коды CRC-16 и CRC-32, имеющие длину 16 и 32 бита соответственно. Код CRC строится по исходному сообщению произвольной длины, т. е. этот код не является блочным в строгом смысле этого слова. Но при каждом конкретном применении этот код — блочный, (m, m + 16)-код для CRC-16 или (m, m + 32)-код для CRC-32. Вычисление значения кода CRC происходит посредством деления многочлена, соответствующего исходному сообщению (полином-сообщение), на фиксированный многочлен (полином-генератор). Остаток от такого деления и есть код CRC, соответствующий исходному сообщению. Для кода CRC-16 полином-генератор имеет степень 16, а для CRC-32 — 32. Полиномы-генераторы подбираются спец. образом и для кодов CRC-16/32 стандартизированы. Для CRC-16, например, стандартным является полином-генератор x16 + x12 + x5 + 1.

Существуют быстрые алгоритмы для расчета CRC-кодов, использующие специальные таблицы, а не деление многочленов с остатком.

CRC-коды способны обнаруживать одиночную ошибку в любой позиции и, кроме того, многочисленные комбинации кратных ошибок, расположенных близко друг от друга. При реальной передаче или хранении информации ошибки обычно группируются на некотором участке, а не распределяются равномерно по всей длине данных. Таким образом, для реальных систем эти коды являются очень полезными.

Коды CRC используются очень широко: модемами, телекоммуникационными программами, программами архивации и проверки целостности данных и многими другими программными и аппаратными компонентами вычислительных систем.

25-26) Шифрование данных. Примеры простых методов шифрования. Шифр-перестановка и шифр-смещение

Шифрование — способ преобразования открытой информации в закрытую и обратно. Применяется для хранения важной информации в ненадёжных источниках или передачи её по незащищённым каналам связи.

В зависимости от структуры используемых ключей методы шифрования подразделяются на

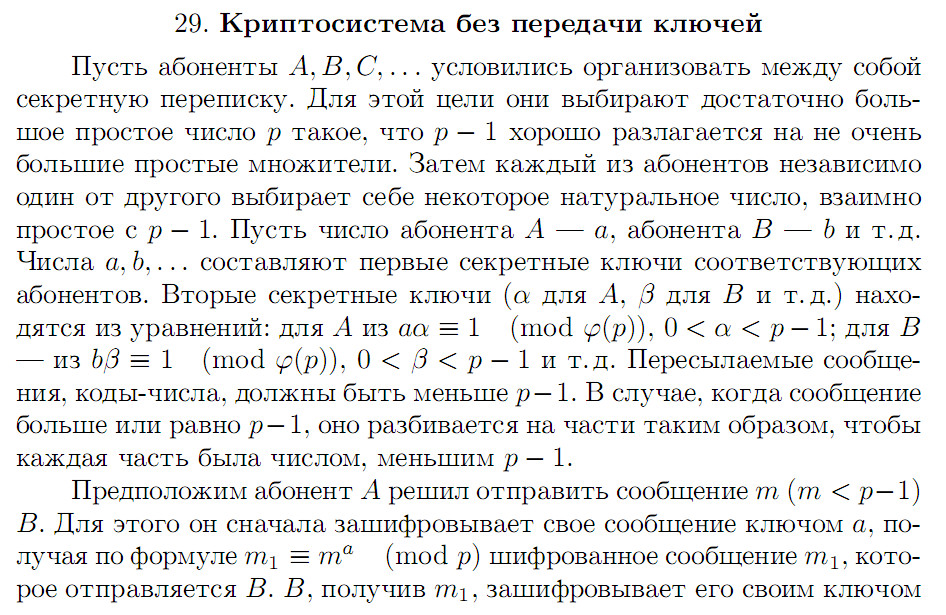
* симметричное шифрование: посторонним лицам может быть известен алгоритм шифрования, но неизвестна небольшая порция секретной информации — ключа, одинакового для отправителя и получателя сообщения;
* асимметричное шифрование: посторонним лицам может быть известен алгоритм шифрования, и, возможно, открытый ключ, но неизвестен закрытый ключ, известный только получателю.

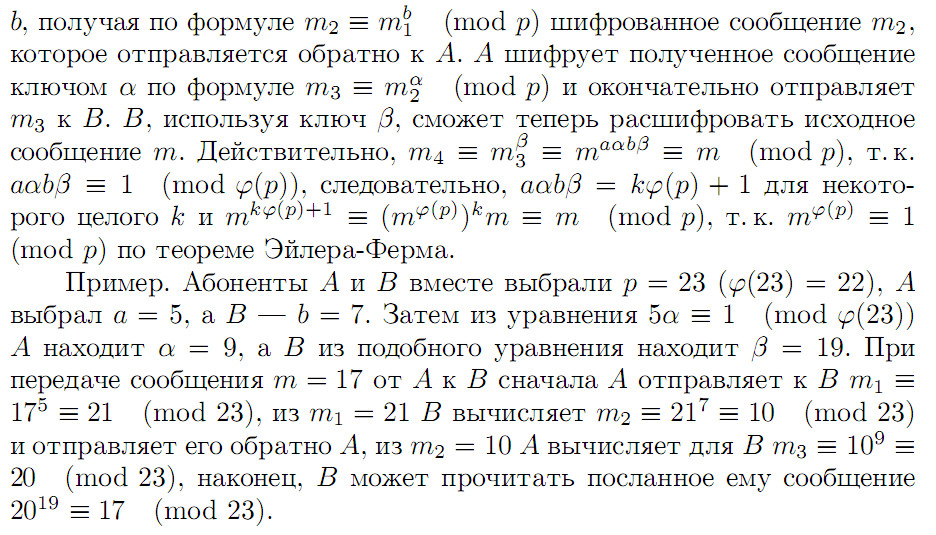
Одним из простейших методов шифрования является **шифр простой замены** – класс методов шифрования, которые сводится к созданию по определённому алгоритму таблицы шифрования, в которой для каждой буквы открытого текста существует единственная сопоставленная ей буква шифра-текста. Само шифрование заключается в замене букв согласно таблице. Для расшифровки достаточно иметь ту же таблицу, либо знать алгоритм, по которой она генерируется. Частным случаем данного метода является **шифр Цезаря** – это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется буквой находящейся на некоторое постоянное число позиций левее или правее него в алфавите. Например, в шифре со сдвигом 3, А была бы заменена на Г, Б станет Д, и так далее.

**Шифр-перестановка.** В шифре перестановки буквы открытого текста не замещаются на другие, а меняется сам порядок их следования. Например, в шифре простой колонной перестановки исходный открытый текст записывается построчно (число букв в строке фиксировано), а шифр-текст получается считыванием букв по колонкам. Расшифрование производится аналогично: шифр-текст записывается поколонно, а открытый текст можно затем прочесть по горизонтали. Также возможно использование ключа. Количество букв в строке есть количество букв в ключе.Каждой букве ключа ставиться порядковый номер(порядок следования определяет алфавит). Если первая буква (слева направо) имеет порядковый номер, например,n, то первым считывается столбец n (номер слева направо) и т.д.

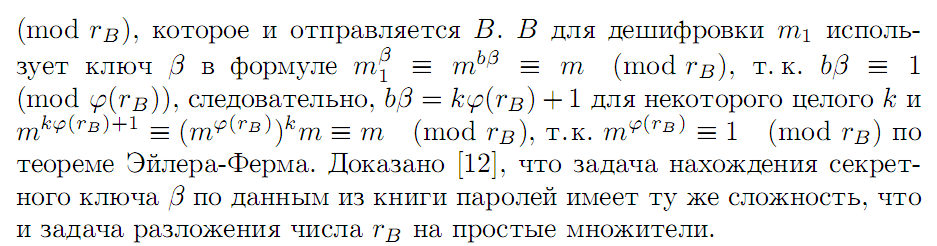
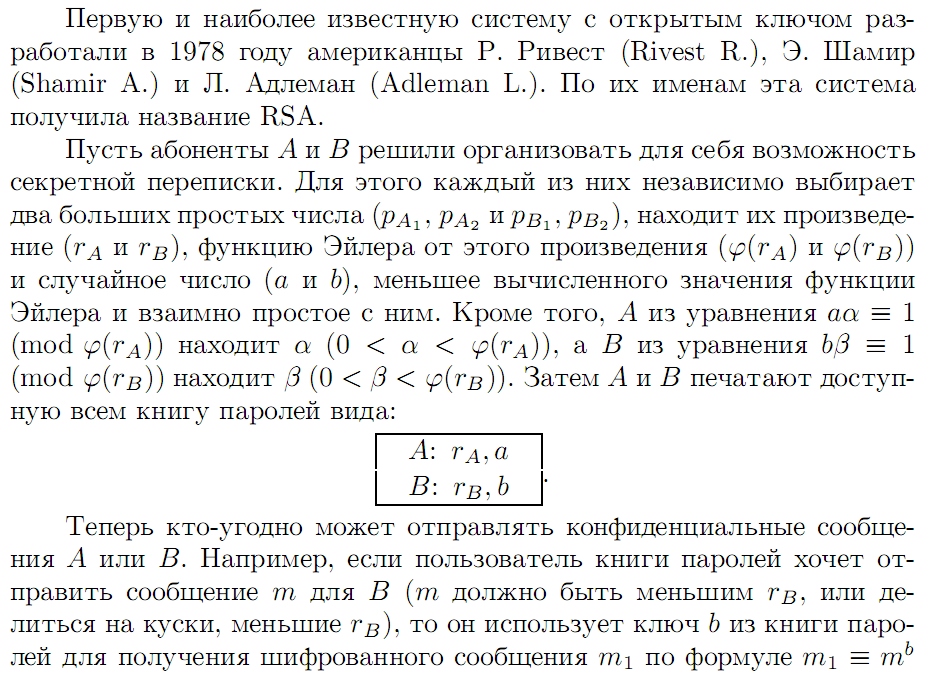
**Шифр-смещение.** Для данного метода шифрования требуется построение таблицы кодов (например, номер следования буквы в алфавите, т.е.буква А будет иметь код 1, Б – 2, В – 3 и т.д.). В сообщении под каждым символом исходного сообщения пишется кодовое слово. Затем код каждого символа исходного сообщения складывается с кодом находящимся под ним символа ключа и полученный в результате код образует символ. Если сумма кодов превосходит максимальный код, то из этой суммы вычитается размер максимального кода.

27) Криптосистемы без передачи ключей

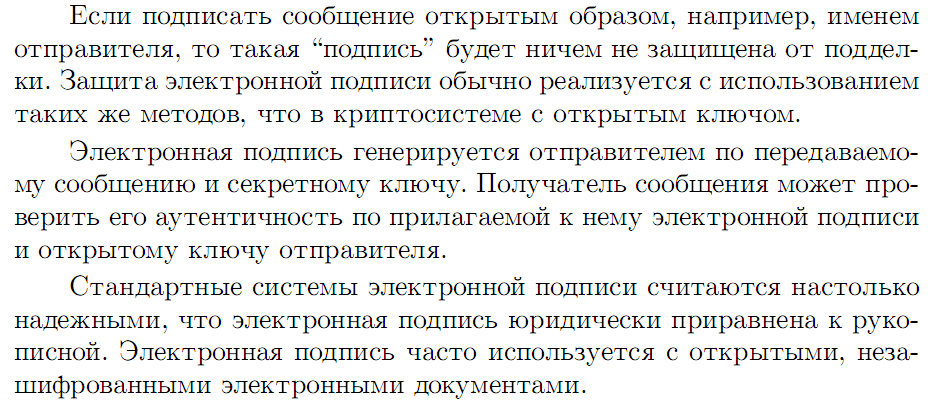
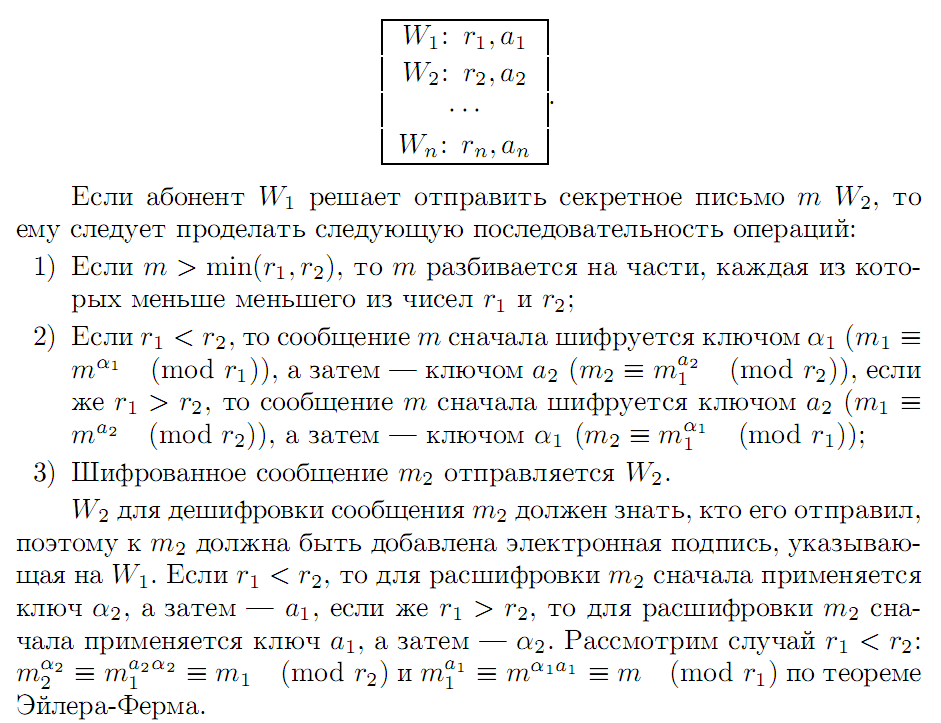
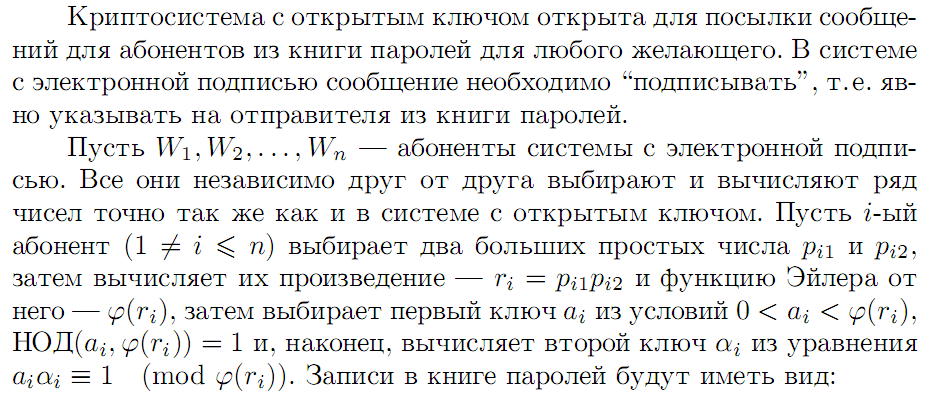




28) Криптосистемы с открытым ключом



29) Электронная подпись

****

30) Системы передачи дискретных сообщений. Виды сигналов

Дискретные сообщения как форма представления информации состоят из конечного набора возможных значений. Физический процесс отображения сообщения называется сигналом. Сигнал характеризуется своим уровнем и моментами изменения данного уровня. Виды сигналов:

1. непрерывные сигналы непрерывного времени (непрерывные или аналоговые) – уровни сигнала могут изменяться в любой момент времени и любым образом (принимать любые значения);
2. непрерывные сигналы дискретного времени – могут принимать произвольные значения, но изменяются только в определённые моменты времени;
3. дискретные сигналы непрерывного времени – могут изменяться в произвольные моменты, но принимают только разрешённые дискретные значения;
4. дискретные сигналы дискретного времени (дискретные) – уровни меняются только в дискретные моменты и принимают только дискретные значения.

Сигналы, формируемые на выходе преобразователя дискретного сообщения в сигнал, являются по информационному параметру дискретными. Это *цифровые сигналы данных (ЦСД).*

*Элемент ЦСД* – часть цифрового сигнала, отличающаяся от других частей значением одного из инф. параметров.

*Значащая позиция* – это фиксируемое значение состояния инф. параметра.

*Значащий момент* – это момент, в кот.происходит смена значений сигнала.

*Значащие интервалы* – это интервалы времени между двумя соседними зн.моментами. Минимальный интервал τ0 наз. единичным.

Сигналы могут быть:

* изохронными (любой ЗИ равен τ0 или целому их числу)
* анизохронными (произвольная длительность, но не менее чем τ0)

Передача сообщений реализуется посредством *информационных каналов* - совокупности устройств, объединенных линиями связи и предназначенных для передачи инф-ии от её источника (начальногоустр-ва канала) до приемника (конечного устр-ва канала).В *дискретных каналах*сигналы на входе и выходе представляют собой последовательность символов некоторых алфавитов.

31) Виды каналов связи. Их характеристики. Фильтр нижних частот

*Канал связи* – это среда и средства передачи информации.

При передаче информации в канале связи могут возникнуть помехи – шумы (гармонические, импульсные, флуктуационные, мультипликативные).

*Информационный канал* — это совокупность устройств, объединенных линиями связи, предназначенных для передачи инф-ии от источника инф-ии (начальногоустр-ва канала) до ее приемника (конечного устр-ва канала).

Линии связи обеспечивают прохождение инф. сигналов между устр-вами канала. Информация обычно передается при помощи эл. тока (по проводам), света (по оптоволокну), э/м волн радиодиапазона (в пространстве), звука (в плотной среде: атмосфере, воде и т.п.) и прочих.Устройства канала — это, как правило, репитеры, передающие усиленным принятый сигнал.

Технические характеристики канала определяются принципом действия входящих в него устр-в, видом сигнала, св-вами и составом физической среды, в которой распространяются сигналы, св-вами применяемого кода.

Эффективность канала хар-тся скоростью и достоверностью передачи информации, надежностью работы устр-в и задержкой сигнала во времени.Задержка сигнала во времени — это интервал времени от отправки сигнала передатчиком до его приема приемником. Достоверность и надёжность напрямую связана с наличием шумов.

В соответствии со структурой вх. и вых. сигналов выделяют каналы:

* *дискретные* (сигналы на входе и выходе представляют собой последовательность символов некоторых алфавитов);
* *непрерывные* (входные и выходные сигналы представляют собой функции от непрерывного параметра (времени)).

Математической моделью шумов является совокупность условных вероятностей: ->канал без шумов

Способность канала передавать информацию характеризуется *ёмкостью* или *пропускной способностью канала*:

N (T) — число всех возможных сигналов

Для вычисления максимальной скорости передачи в канале с шумом при известной ёмкости канала и энтропии источника используют теорему Шеннона: .

32) Способы передачи и обработки сигналов в системах ПДС

Дискретные сигналы могут передаваться последовательно и параллельно.

При последовательной передаче единичные элементы следуют в канале поочередно.

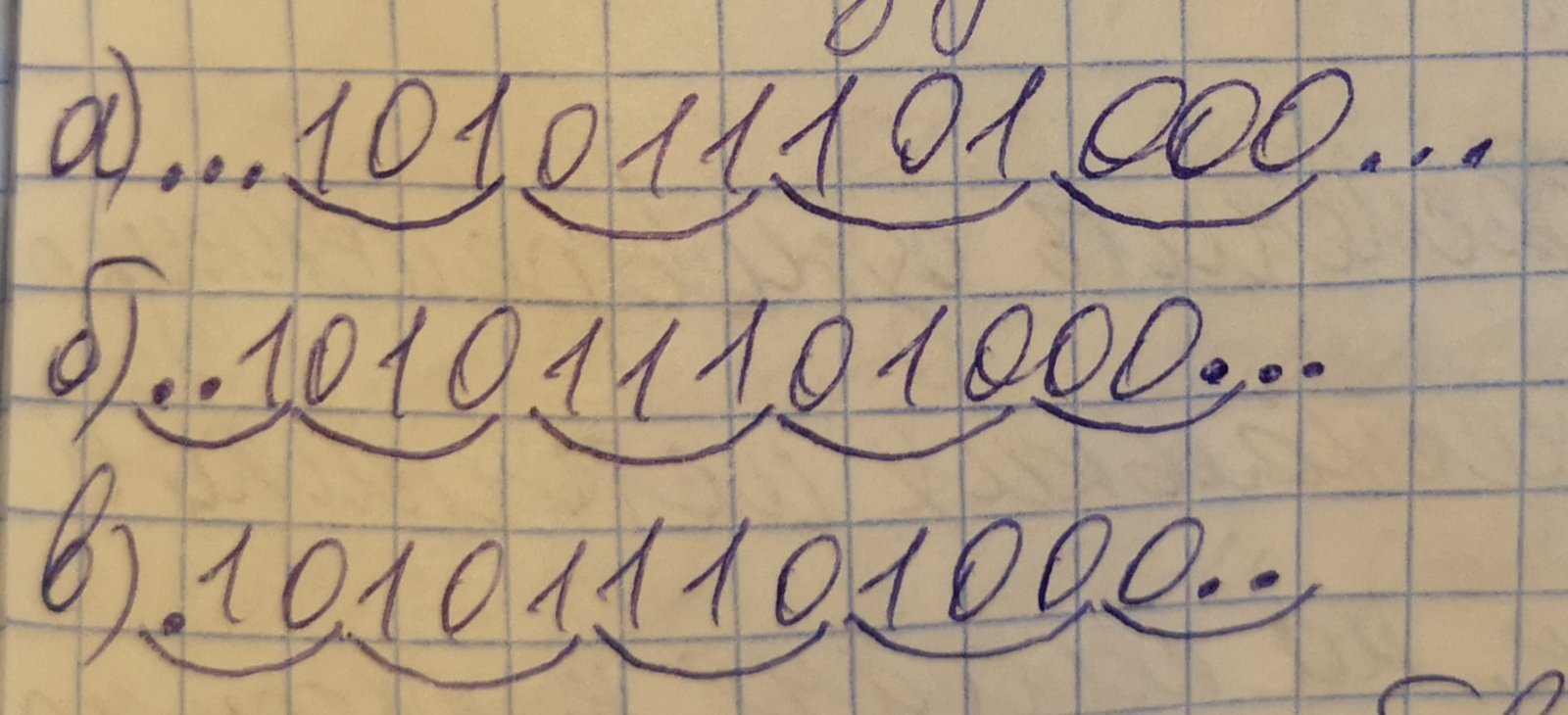
При параллельной передаче единичные элементы объединяются в группы. Элементы, составляющие группу, передаются одновременно по отдельным каналам, при использовании модулированного сигнала в разной полосе частот.

Различают синхронную и асинхронную передачу.

При синхронной передаче дискретного сигнала его значащие моменты (ЗМ) находятся в требуемом постоянном фазовом соотношении с ЗМ другого передаваемого сообщения.

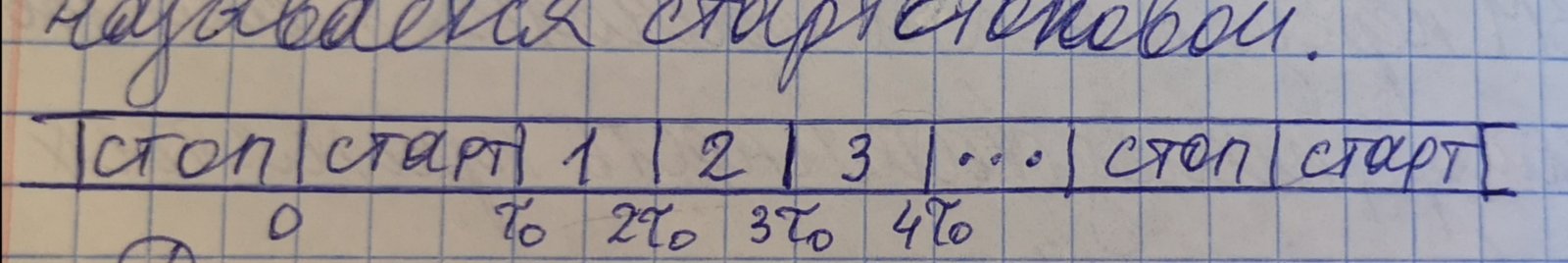
При асинхронной передаче ЗМ дискретного сигнала могут находиться в любых фазовых соотношениях с ЗМ других сигналов.

На приемной стороне УПС определяет вид элемента (0 или 1). Затем из элемента формируются кодовые комбинации, декодирование которых определяет символы, переданного сообщения. Такой метод приема дискретных сообщений – называется поэлементным. В общем виде задачу определения вида переданного элемента можно свести к задаче сравнения принятого сигнала с эталоном. Для двоичных сигналов достаточно иметь не более двух эталонов. Для обеспечения правильного приема в технике ПДС решаются различные задачи синхронизации. Синхронизация – процесс установления и поддержки определенных временных соотношений между двумя или несколькими процессами. На приеме для правильного воспроизведения элементов кодовых комбинаций – необходимо правильно отделить один элемент от другого. Для этого используют методы поэлементной синхронизации. Поэлементной – называется синхронизация переданного и принятого дискретных сигналов, при которой устанавливается и поддерживаются требуемые временные соотношения между ЗМ переданных и принятых элементов. Например, последовательность принятых элементов … 101011101000 …, состоящая из трехэлементных кодовых комбинаций на приеме может быть разбита на комбинации, следующими вариантами:



т.е. варианты а, б, в – разные, и если, вариант а – правильные комбинации, то в варианте б и в – они приняты с ошибкой.

Правильное отделение одной кодовой комбинации от другой осуществляется методами групповой синхронизации. Обычно в состав каждой комбинации информационных элементов вводят специальные элементы в начале и конце комбинации. Элемент в начале кодовой комбинации называют стартовым, элемент в конце – стоповым, и передаваемая т.о. последовательность называется стартстоповой.



Данный метод передачи относится к асинхронным, т.к. передачу любой кодовой последовательности можно начать в любой момент.

33) Сигналы, используемые в технике ПДС. Свойства сигналов

Структурная схема системы ПДС



**УПС** – устройство преобразования сигнала

**ИС** – источник сигнала **ПС** – приёмник сигнала

**ПДС** – передача дискретного сигнала

*Декодер канала* извлекает информацию с учётом коррекции ошибок. *Декодер приёмника* переводит цифровую инф-ю в аналоговую.

В технике ПДС чаще всего используются коды с основанием 2 – *бинарные, двоичные* – так как они характеризуются простотой реализации, надёжностью элементов двоичной логики, малой частотностью действия внешних помех.

С целью повышения надёжности передачи используется избыточное кодирование, позволяющее на приёме обнаруживать и, иногда, исправлять ошибки (кодер канала). На приёмном конце декодер канала выполняет обратные преобразования. С целью согласования кодера и декодера канала с непрерывным каналом связи (средой) используется УПС.

Дискретные каналы делятся на:

* *синхронные*- предназначены для передачи только изохронных сигналов, каждый ед. элемент вводится в строго определённые моменты времени;
* *асинхронные*(кодонезависимые, прозрачные)- можно передавать как изохронные, так и анизохронные сигналы.

34) Амплитудная модуляция и ее характеристики. Схема амплитудного модулятора

Наиболее часто используют в качестве несущей гармоническое колебание:

Где -амплитуда; -циклическая частота; - фаза.

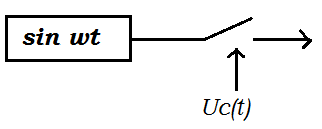
На любой из этих параметров можно воздействовать входным сигналом, получим соотв. амплитудную, частотную и фазовую модуляции.

**Амплитудная модуляция** – простейшая и наиболее раннее известная модуляция.

Частота и фаза несущей не изменяются. Для передачи двоичной информации сигнал может быть:

Сигнал в этом случаи может быть записан:

Для получения АМ можно использовать простейшую схему:



Управление запиранием и отпиранием ключа осуществляется модулиров. сигнала , которое можно представить в виде ряда Фурье:

Подставив это значение в выражение для получим:

Таким образом АМ сигнал содержит в своем составе спектральные сосатвляющие:

Если для сигнала можно выделить максимальное значение , то -нижняя боковая полоса АМ сигнала, -верхняя боковая полоса АМ сигнала.

- это ширина спектра сигнала, то АМ сигнал имеет ширину спектра в двое большую.

Любая модуляция имеет основной задачей перенос сигнальных частот в более высокую область. Таким образом АМ равно

Если задана ширина канала , то надо выбирать:

В пределах сигнала , тогда время нарастания перех. процесса амплитуда

Учитывая, что получаем, что

35) Частотная модуляция и ее характеристики. Схема частотного модулятора

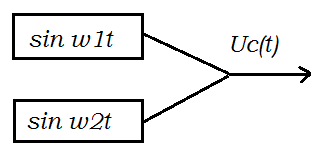
Наиболее часто используют в качестве несущей гармоническое колебание:

Где -амплитуда; -циклическая частота; - фаза.

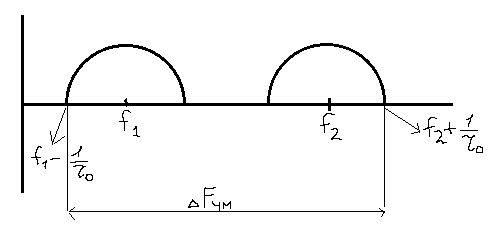
На любой из этих параметров можно воздействовать входным сигналом, получим соотв. амплитудную, частотную и фазовую модуляции.

Частотная модуляция – Является другим способом переноса спектра частичного сигнала в заданный диапазон частот, но в отличие от АМ, этот способ преобразования спектра является нелинейным. Для ЧМ амплитуда остается постоянной, а меняется частота в зависимости от управл. сигнала частота.

При двоичной частотной модуляции модулятор должен вырабатывать отрезки гармонических колебаний с частотой , соотв. передаче 0 и 1.



Сигнал на выходе модулятора может рассматриваться как суперпозиция(сумма) 2х АМ сигналов, один из которых имеет несущую f1, а другой f2(f2>f1). Соответственно спектр ЧМ сигнала может быть представлен как суперпозиция 2х АМ сигналов и , тогда ширина спектра ЧМ сигнала выражается через



Ширина спектра ЧМ сигнала шире, чем у АМ сигнала на величину равную расстоянию между несущими f2 и f1, значение равное:

Характеризует изменение частоты при передаче 1 или 0 относительно ее среднего значения:

-дивиация частоты.

Отношение девиации частоты и скорости телеграфирования(модулирования) называется индексом частотной модуляции:

Тогда учитывая определение индекса частотной модуляции и выражение для ширины ЧМ сигнала равно:

Т.е. чем больше индекс модуляции(тем выше скорость телеграфирования), тем шире при прочих равных условий спектр ЧМ сигнала.

Если ширина спектра ЧМ сигнала примерно равна ширине АМ сигнала. Время нарастания переходного процесса определяется на выходе канала постоянного тока из след. соображения:

1) Т.к.

2) , то

Также как и для АМ определяет , тогда

3) Время нарастания:

Учитывая необходимость соблюдения неравенства , получим, что

При заданном значении макс. скорость модуляции при ЧМ меньше, чем при АМ.

36) Фазовая модуляция и ее характеристики. Схема фазового модулятора

Наиболее часто используют в качестве несущей гармоническое колебание:

Где -амплитуда; -циклическая частота; - фаза.

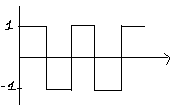
На любой из этих параметров можно воздействовать входным сигналом, получим соотв. амплитудную, частотную и фазовую модуляции.

При фазовой модуляции (ФМ), амплитуда и частота несущей остаются неизменными, а меняется лишь фаза. ФМ сигнал можно записать в виде:

Где ; - девиация фазы.

Будем полагать что .

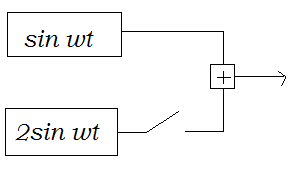
При передаче бинарного сигнала, пусть

**

*-* в зависимости от того какой бит передан.

Для лучшего различия необходимо, чтобы они максимально отличались друг от друга().

Т.е. при передаче 1 выходной сигнал совподает по фазе с несущим, а при передаче 0 - 180º



Такой ФМ сигнал можно представить в виде суммы 2х АМ сигналов. Для получения первого используется несущая.

*Для 2-го*

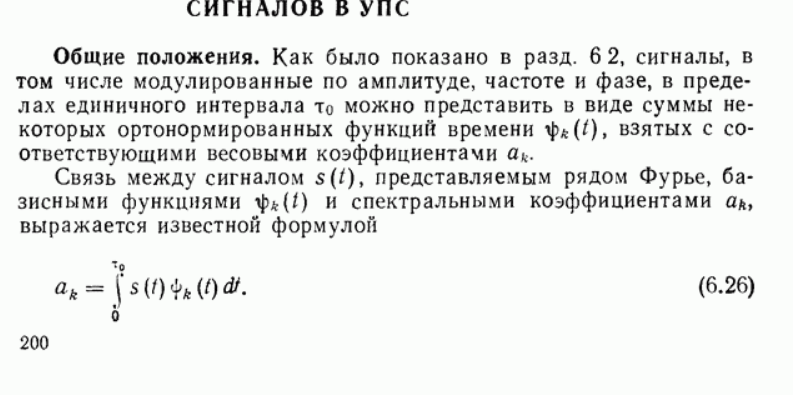
Тогда, если ключ разомкнут, поступает сигнал , если замкнут – поступает сумма сигналов

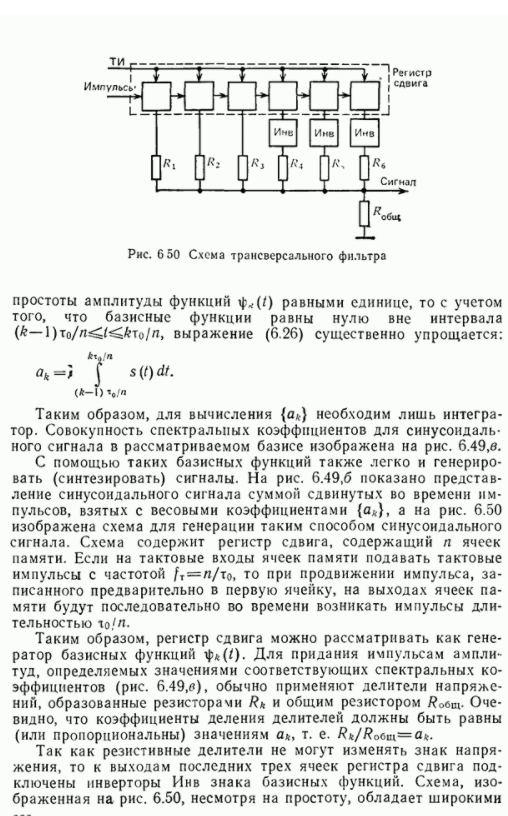
:

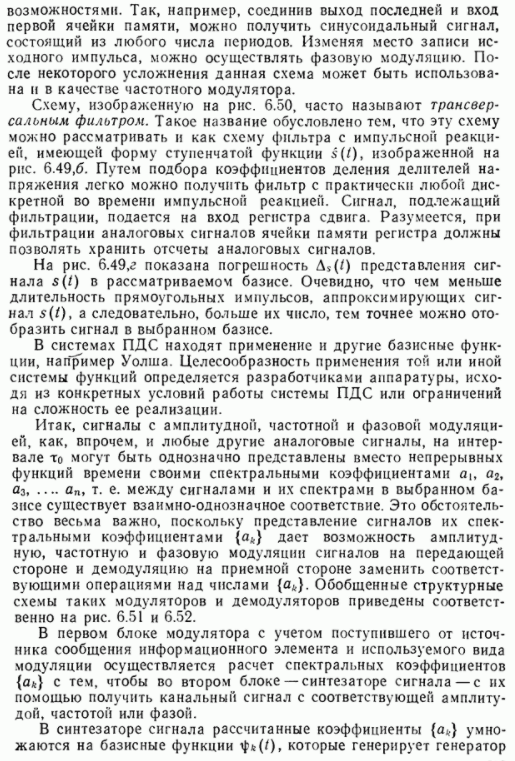
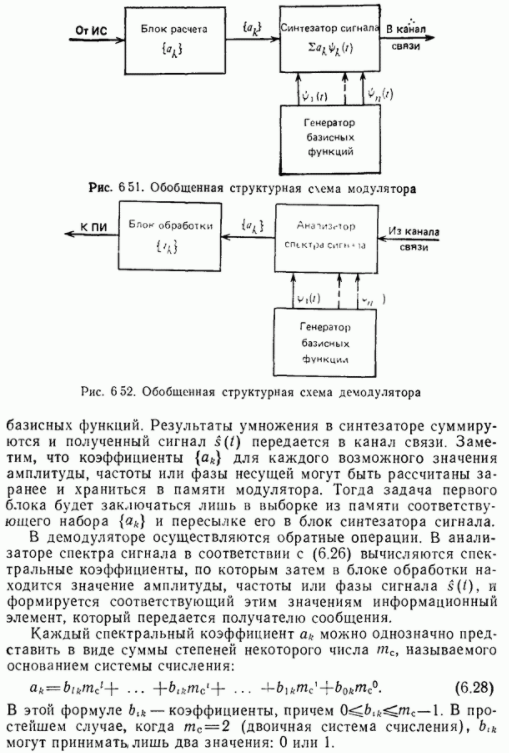
При спектр амплитуд ФМ сигнала содержит те же составляющее, что и спектр АМ сигнала, но амплитуда этих составляющих в отличие от АМ, в 2 раза больше.

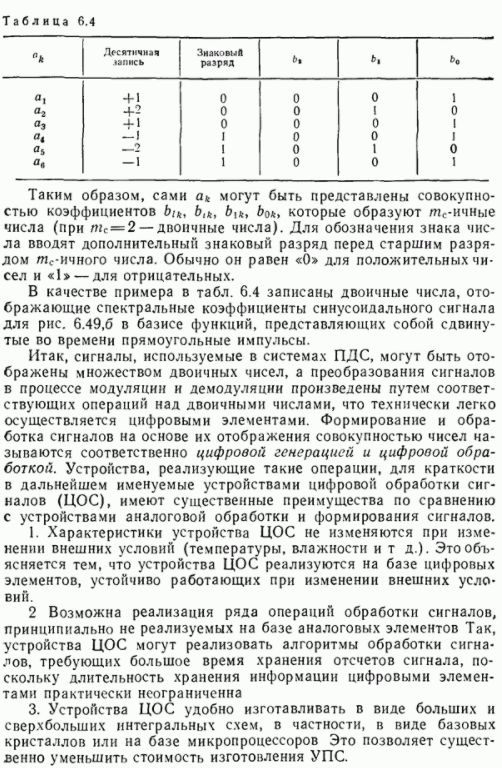
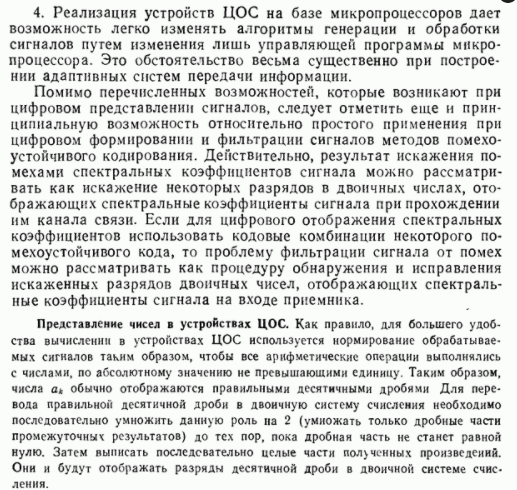
Равенство полос АМ и ФМ сигналов предполагает равенство и максимальных скоростей модуляции, но большая амплитуда спектральных составляющих обуславливает для ФМ большую помехоустойчивость.

37) Цифровая генерация и обработка сигналов в УПС

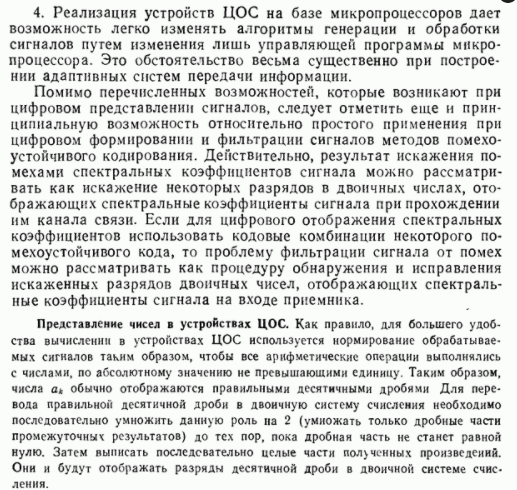
****

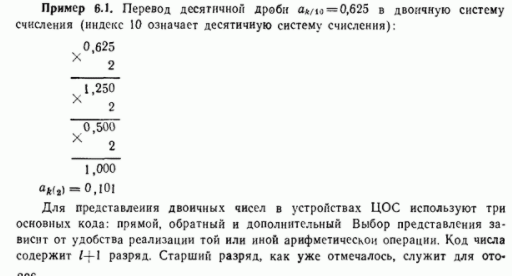
****

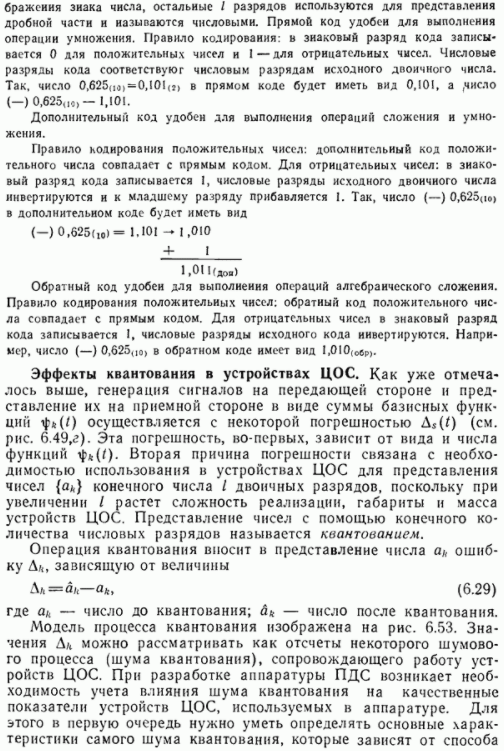
** **

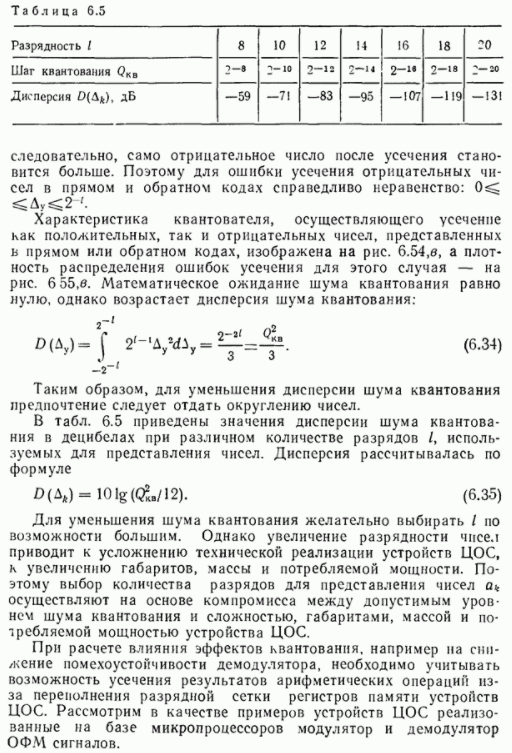
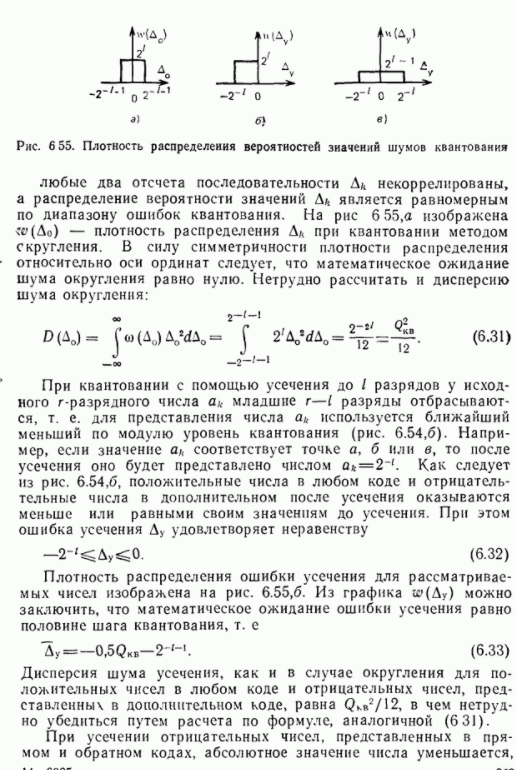
** **

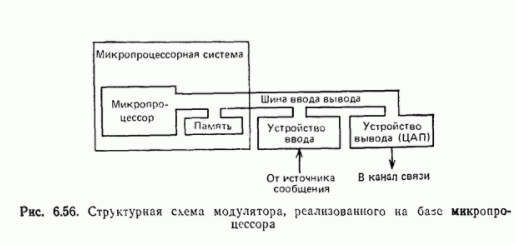
38) Представление чисел в устройствах ЦОС. Эффекты квантования в устройствах ЦОС











39) Основные понятия открытых систем

Открытая система – система, которая способна взаимодействовать с другой системой, посредством реализации международных стандартных протоколов.

Протокол – набор правил, определяющих взаимодействие устройств, программ, систем обработки данных, процессов или пользователей.

Технологии открытых систем является одним из основных направлений информационных технологий и реализуется всеми ведущими фирмами поставщиками, средств техники, передачей информации и программного обеспечения.

Открытыми системами могут быть как конечные так и промежуточные системы, отвечающие следующим требованиям:

1. Возможность переноса прикладных программ с минимальными изменениями на широкий диапазон систем.
2. Совместная работа с другими прикладными системами на локальных и удаленных платформах.
3. Взаимодействие с пользователем, облегчающим переход от системы к системе.

Свойства открытых систем:

1. Переносимость прикладного ПО и повторная применимость. Переносимость это перенос всего соответствующего данному приложению ПО на другие платформы. Повторная применимость – перенос в новые приложения некоторой части работающей программы.
2. Переносимость данных. Означает возможность переноса на новые платформы данных, хранящихся во внешней памяти. Это свойство обеспечивается применением в открытых системах стандартов, строго регламентирующих форматы и способы представления данных.

Интероперабельность (совместимость) – способность системы взаимодействовать с другими системами, посредством обмена информации и совместного ее использования.

1. Функциональное совместимость …. – возможность обмена данными между различными программами, в том числе между программами реализуемыми на разнородных платформах, а так же возможность совместного использования данных.
2. Функциональная совместимость (интероперабельность) управления и безопасности – унификация и целостность средств административного управления и управления информационной безопасностью с использования международных стандартов.
3. Переносимость пользователей – обеспечение возможности для пользователей избежать необходимости переобучения при работах на различных платформах.
4. Расширяемость – способность системы эволюционировать с учетом изменений стандартов, и требований пользователей
5. Маштабируемость - … в широком диапазоне параметров определяющих технические и ресурсные характеристики системы (число процессоров, число пользователей в сети, максимальное число пользователей).
6. Прозрачность реализации – способ построения системы, при котором все особенности ее реализации скрываются за стандартными интерфейсами.
7. Поддержка пользовательских требований – точная спецификация пользовательских требований, определенных в виде наборов сервисов.

Замечание: открытая система не обязательно должна быть полностью доступна другим открытым системам. Чтобы избежать необходимости защиты информации в компьютерах и средствах коммуникации. Сущность открытых систем состоит в обеспечении возможности переносимости прикладных программ и взаимодействия систем друг с другом. Эта возможность достигается за счет использования международных стандартов на все программные и аппаратные интерфейсы.

Стандарт – документированное соглашение содержащее технические условия или другие точные критерии соответствия продуктов, процессов и услуг своему назначению.

Различают стандарты де-факто и де-юре.

Стандарт де-факто означает что продукт или система какого-либо производителя захватили значительную часть рынка. Другим производителям приходится эмулировать, копировать и использовать их чтобы остаться на рынке.

Стандарт де-юре. Создается официально аккредитованными организациями по разработке стандартов.

Преимущества открытых систем:

Для пользователя открытые системы обеспечивают:

1. Новые возможности сохранения сделанных вложений;
2. Замена отдельных компонентов без перестройки всей системы;
3. Освобождение от зависимости от одного поставщик аппаратных или программных средств, возможность выбора продуктов при соблюдении стандартов;
4. Дружественность среды, мобильность персонала в процессе эволюции системы;
5. Возможность использования информационных ресурсов имеющихся в других системах (оргиназациях).

Для проектировщика информационных систем:

* возможность использования разных аппаратных платформ;
* Возможность совместного использования прикладных программ, реализованных в разных системах;
* Развитые средства инструментальных сред, поддерживающих проектирование;
* Возможности использования готовых программных продуктов и информационных ресурсов.

Для разработчиков общесистемных программных средств:

1. Новые возможности разделения труда, благодаря повторному использованию программ;
2. Развитые инструментальные среды и системы программирования;
3. Возможности модульной организации программных макросов, благодаря стандартизации программных интерфейсов.

40) Эталонная модель взаимодействия открытых систем

Перемещение информации между компьютерами различной конфигурации является очень сложной задачей. В начале 80-ых годов. ISO(международная организация стандартизации) и МККТТ (международный консультационный комитет по телеграфии и телефонии) поставили задачу создания модели сети, помогающей создавать реализации взаимодействия других сетей. Предпосылками для данной работы явились:

1. необходимость эталонной системы, обеспечивающей взаимодействие сетевых средств, созданных различными разработчиками;

2. необходимость теоретически обоснованной сетевой модели, решающей задачу перемещения информации между компьютерами различных систем;

3. разбиение общей задачи перемещения информации на более мелкие задачи.

В 1984 году международная организация разработала эталонную модель под названием OSI (open system interconnection)- взаимодействия открытых систем.

Для взаимодействия двух приложений посредством сети необходимо:

1. найти приложение, которое будет производить обмен информации;

2. установить и поддерживать связь;

3. обеспечить обработку потерь и помех при обмене.

Модель OSI разделяет задачу сетевого обмена на 7 более мелких задач, каждая из которых формулируется таким образом, что для её решения требуется минимум внешней информации, при этом каждый уровень (подзадача) имеет достаточную автономность. Уровни взаимодействуют на строго иерархической основе: каждый уровень обеспечивает сервис для вышестоящего уровня, запрашивая сервис у нижестоящего уровня.

Данная 7-ми уровнева модель и ее протоколы и услуги со временем менялись.

С появлением в 1984 г. Первого стандарта с последующим наполнением этой модели конкретными протоколами, практическая привлекательность начало быстро возрастать. Так в начале 90-х годов появились правительственные профили взаимодействия открытых систем объединивших многие развитые страны.

Система GOSIP была попытка объединения различных стран в единую промышленно-правительственную спецификацию открытых систем. Но в 90-е годы быстрый рост сети интернет с ее более простыми протоколами и др. спецификации захватили лидерство в сфере обмена информации между удаленными компьютерами.

Процесс стандартизации взаимодействия открытых систем никогда не был связан с какой-либо конкретной рабочей средой. Модель взаимодействия открытых систем описывает как информация проходит путь по сети от одной прикладной программы к другой(или от одной сети к другой.)

41) Технология передачи информации в модели взаимодействия открытых систем

Для передачи данных обе системы должны удовлетворять стандартом эталонной модели взаимодействия открытых систем.

1-ый этап: программа источник передает данные верхнему уровню системы, в которой она реализована, при этом происходит обработка данных, которое состоит в том, что к информации добаляется заголовок, содержащий служеюную информацию необходимую для адресации сообщения и выполнения контрольных функций, это дополнительная информация(заголовок) помещается перед фактическими данными и весь информационный блок передается в следующий смежные нижестоящие уровни системы.

2-ой этап: каждый уровень системы принимая информацию от предыдущего(верхнего) уровня системы добавляет свои данные, которые необходимы для функционирования этого уровня. При прохождении каждого очередного уровня данные получают новый заголовок, при этом по мере продвижения через уровни информация кодируется постепенно преобразовываясь в сигналы, которые можно передавать по каналу связи.

3-ий этап: нижний уровень системы не добавляет сообщение заголовка, он обеспечивает передачу информации виде последовательности электрических сигналов по каналу связи.

4-ый этап: на принимающей стороне информация проходит снизу вверх и на каждом уровне не соответствующая часть заголовка сообщения отделяется т.е. каждый уровень на принимающей стороне получает данные в том же виде, в котором они были отправлены соответствующим уровнем на передающей стороне т.е. полученный заголовок прочитывается и происходит обработка информации в соответствии с командами содержащимися в этом заголовке.

5-ый этап: верхний уровень принимающей системы передает данные прикладной программе для которой производился обмен информацией, при этом все дополнительные заголовки уже отделены от сообщения.

Замечание: концепция заголовков относительна и не зависит от уровня, которой в данный момент анализирует информационный блок. Не все уровни добавляют заголовки, а просто выполняют трансформацию данных чтобы, сделать их более читаемы для очередных уровней.

42) Физический уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Физический уровень.** Является базовым в иерархии протоколов моделей взаимодействия. Назначение: состоит в обеспечении механических, электрических функциональных и процедурных средств установления, поддержания и разъединения физических соединений для передачи последовательности бит.

*Функции физического уровня:*

1) Установление и разъединение физических соединений.

2) Определение характеристик физической среды передачи данных таких как полоса пропускания, помехозащищённость и т.д.

3) Определение характеристик электрических сигналов таких как уровень напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов.

4) Передачи последовательности бит в синхронном или асинхронном режиме.

В настоящее время используют два способа передачи данных:

1) аналоговый способ, который обеспечивает широкополосную передачу информации за счёт использования в одном канале различных несущих частот.

2) цифровой (узкополосный) способ передачи данных, при котором данные передаются в их естественном виде на одной частоте.

При передаче данных по аналоговым каналам связи последовательность бит на входе канала преобразуется в модемах (устройство модуляции-демодуляции) в аналоговые сигналы, параметры которых согласованы с параметрами физической среды, которые на принимающей стороне преобразуются в последовательность бит.

Для цифровых каналов связи не производится преобразование бит в аналоговые сигналы. Вместо модемов используют линейные контроллеры, обеспечивающие сопряжение оборудования обработки данных с физическим каналом.

Функции физического уровня реализованы только аппаратными средствами, на которые разработаны и широко используются международные стандарты. Данные функции реализуются во всех устройствах, подключённых к сети. На компьютере функции физического уровня выполняются оконечными сетевыми устройствами: модемом или сетевой картой.

43) Канальный уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Канальный уровень.** Обеспечивает надёжную передачу массивов данных между сетевыми открытыми системами. Назначение данного уровня состоит в управлении доступом к передающей среде и в управление передачи данных.

Функции канального уровня:

1) Приём пакетов, поступающих с сетевого уровня.

2) Подготовка пакетов к передаче.

3) Генерация стартового сигнала и организация начала передачи информации.

4) Передача информации по каналу.

5) Проверка получаемой информации и исправление ошибок.

6) Генерация сигнала, окончание передачи и перевод канала в пассивное состояние.

На физическом уровне при передаче сигналов не учитывается возможность того, что среда будет занята. Поэтому одной из задач канального уровня является проверка доступности среды передачи. Другой задачей является реализация механизмов обнаружения и коррекции ошибок. Для этого на данном уровне биты группируются в наборы, называемые кадрами. Канальный уровень помещает специальную последовательность битов в начало и конец каждого кадра (устанавливает границы кадра). Также используют контрольные суммы, добавляя их к каждому кадру. Получатель снова вычисляет контрольную сумму, сравнивает её с контрольной суммой из кадра. Если они совпадают, кадр считается правильным. Иначе фиксируется ошибка. Существует три вида протоколов канальных уровней:

1. Протокол с остановками и ожиданием.

2. Протокол с непрерывной передачей.

3. Протокол с выборочной передачей.

Возможно также использование двух режимов передачи:

1. Дупликсный режим, при котором одновременно происходит передача и приём сообщений.

2. Полудупликсный режим, когда передача и приём производится попеременно.

Протокол с остановками и ожиданием характеризуется тем, что одновременно передаётся только один кадр, после чего передающая сторона ждёт подтверждения. Если поступит отрицательное подтверждение или произойдёт рассрочка времени ожидания ответа, кадр передаётся повторно. Этот протокол подходит для полудупликсных каналов связи.

При использовании протокола с непрерывной передачей кадры передаются непрерывно без ожидания подтверждения. При получении отрицательного подтверждения неподтверждённый кадр и все последющие передаются вновь. Этот протокол более производительный и предполагает использование дупликсной связи.

Для протокола с выборочной передачей повторная передача требуется только для кадра, о котором поступило отрицательное подтверждение. На принимающей стороне требуется накопитель с перестроениями, так как кадры могут повторно передаваться не по порядку.

Замечание. Из-за увеличения стоимости реализации данный протокол пока не нашёл применения.

44) Сетевой уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Сетевой уровень.** Служит для организации транспортной системы, объединяющей несколько сетей. При этом эти сети могут использовать различные принципы передачи сообщений и обладать произвольной структурой связи.

Назначение данного уровня состоит в установлении, поддержании и разъединении сетевых соединений между объектами транспортного уровня и прокладке оптимальных маршрутов.

Функции сетевого уровня:

1) Установление сетевого соединения для передачи объектов транспортного уровня в форме блоков данных.

2) Предоставление сетевых адресов, используемых для идентификации транспортных объектов.

3) Обеспечение независимости передачи данных от используемых средств передачи.

4) Предварительное согласование параметров качества обслуживания между пользователем и поставщиком сетевой службы.

5) Управление скоростью передачи блоков данных со стороны приёмника информации.

6) Возможность передачи последовательности срочных блоков данных (срочные блоки данных имеют ограниченную длину и их передача осуществляется по правилам, отличным от правил нормальной передачи данных).

7) Обнаружение и исправление ошибок передачи данных.

8) Разъединение сетевого соединения либо пользователями, либо поставщиками сетевой службы.

Одной из важных задач сетевого уровня является **маршрутизация**. Протоколы маршрутизации выбирают оптимальные маршруты через последовательность соединённых между собой подсетей. Сети соединяются между собой специальными устройствами, называемыми маршрутизаторами. **Маршрутизаторы** – это программные или программно-аппаратные средства определения маршрута передачи данных между узлами сети. Обычно критерием по выбору маршрута является время передачи данных по этому маршруту, которое зависит от пропускной способности каналов связи и интенсивности трафика, которая может меняться с течением времени. Трафиком называют поток сообщений в сети передачи данных или рабочую нагрузку линии связи. Некоторые алгоритмы маршрутизации пытаются приспособиться к изменению нагрузки. Другие алгоритмы функционируют на основе средних показателей за длительное время. Возможны другие критерии. Например, надёжность передачи.

45) Транспортный уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Транспортный уровень.** Обеспечивает приложениям или верхним уровням модели (прикладному и сеансовому) передачу данных с заданной степенью надёжности.

Назначение транспортного уровня состоит в обеспечении надёжного последовательного обмена данными между пользователями с использованием сетевого уровня и управление потоком данных, гарантирующим правильный приём.

Функции транспортного уровня:

1) Установление транспортного соединения между пользователями.

2) Отображение транспортного адреса на сетевой.

3) Передача блоков данных без ограничения их длины и содержимого.

4) Деление длинных сообщений, поступающих от верхних уровней, на пакеты данных.

5) Управление темпом обмена.

6) Формирование первоначальных сообщений из набора пакетов, полученных через нижние уровни.

7) Разъединение транспортного соединения.

Транспортный уровень оптимизирует использование имеющейся сетевой службы для обеспечения требуемых сеансовыми объектами характеристик передачи с минимальной стоимостью. На пути от отправителя к получателю пакеты могут быть искажены или утеряны. Транспортный уровень предоставляет 5 классов сервиса, которые отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств объединения нескольких соединений между различными протоколами через общий транспортный протокол, способностью к обнаружению и исправлению ошибок передачи (искажение, потеря или дублирование пакетов).

Выбор класса сервиса транспортного уровня определятся задачами обеспечения надёжности передачи информации, например, если качество каналов передачи данных высокое, т.е. вероятность возникновения ошибок мала, то обычно используют один из облегчённых сервисов и наоборот, если транспортные средства нижних уровней не надёжны, то целесообразно использовать более развитый сервис транспортного уровня, использующий максимум средств для обнаружения и устранения ошибок. Функционирование транспортного уровня состоит из 3-х фаз:

1) Фаза установления соединения. Выполняется:

-выбор сетевого соединения, наиболее соответствующего требованиям сеансового объекта с учётом стоимости и качества обслуживания.

-решение о необходимости объединения или расщепления транспортного соединения с целью оптимизации использования сетевых соединений.

-выбор оптимального размера транспортного блока.

-выбор функций, задействованных в фазе передачи данных.

-отображение транспортных адресов в сетевые.

2) Фаза передачи данных. Производится пересылка транспортных блоков по транспортному соединению с использованием сетевых протоколов. Сетевые протоколы реализуют продвижение пакетов через сеть. Могут быть задействованы следующие функции, которые были согласованы в фазе установления соединения:

-упорядочение

-управление потоком

-обнаружение ошибок

-исправление ошибок

-передача срочных данных

-разграничение транспортных блоков

-идентификация транспортных соединений

3) Фаза разъединения соединения. Выполняются следующие функции:

-оповещение о причине разъединения

-идентификация разъединяемого транспортного соединения

Все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети, являющихся компонентами их сетевых ОС.

Протоколы нижних 4-х уровней обобщённо называют сетевым транспортом или транспортной подсистемой.

# 46) Сеансовый уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Сеансовый уровень.** Устанавливает, управляет и завершает сеансы взаимодействия между прикладными задачами.

Назначение уровня: обеспечивать управление диалогом, т.е. фиксация, какая из сторон является активной, а какая пассивной, а также предоставление средств синхронизации.

Функции сеансового уровня:

1) Формирование сквозного канала связи между взаимодействующими прикладными каналами

2) Установление и расторжение сеансовых соединений.

3) Выбор режима передачи между прикладными процессами.

4) Управление очерёдностью передачи данных по их приоритетам.

5) Обмен нормальными данными

6) Обмен срочными данными

7) Неделимая служба

8) Управление взаимодействием (маркеры)

9) Синхронизация сеанса (контрольные точки)

10) Восстановление сеанса

Неделимая служба – это услуга, посредством которой сеансовые блоки данных, посланные по сеансовому соединению, не предоставляются объекту-получателю до тех пор, пока это явно не разрешено объектом-отправителем.

Управление взаимодействием – это услуга, позволяющая взаимодействующим объектам явно управлять очерёдностью выполнения некоторых управляющих функций.

Синхронизация сеансового соединения – это услуга, позволяющая соединяемым объектам определять и идентифицировать точки синхронизации, осуществлять повторную установку сеансового соединения и согласовывать точку повторной синхронизации.

На сеансовом уровне обеспечиваются средства, необходимые для организации и синхронизации диалога между взаимодействующими объектами для управления информационным обменом между ними. Сеансовые соединения устанавливаются по запросу одного из объектов, передаваемому в сеансовой точке доступу к службе, и разъединяются либо данным объектом, либо сеансовыми объектами. В каждый момент времени между сеансовыми и транспортными соединениями существует взаимно-однозначное соответствие.

В случае отказов в транспортном соединении сеансовый уровень может выполнять действия, необходимые для повторного транспортного соединения. Сеансовые объекты при наличие соответствующей услуги оповещают взаимодействующие объекты о том, что служба была прервана и восстанавливают службу только по указанию одного из взаимодействующих объектов, что позволяет объектам провести повторную синхронизацию и продолжить функционирование с некоторого согласованного состояния.

Разъединение сеансового соединения производится по запросу взаимодействующих объектов и без потерь данных. Возможно также преждевременное разъединение сеансового соединения с потерями данных. Сеансовые протоколы могут осуществлять некоторые функции по управлению уровнем, например, контроль ошибок.

Замечание: большинство существующих приложений редко используют сеансовый уровень. Обычно функции этого уровня объединяют с функциями прикладного уровня и реализуют в одном протоколе.

# 47) Представительный уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Представительный уровень.** Отвечает за то, чтобы информация, посылаемая из прикладного уровня одной системы, была читаемой для прикладного уровня другой системы.

Назначение уровня: обеспечение независимости прикладных взаимодействующих объектов под использованием конкретного синтаксиса передаваемой информации.

Синтаксис – это совокупность правил образования правильных (допустимых) конструкций языка.

Функции уровня:

1) запрос установления сеанса

2) выбор синтаксиса

3) согласование и повторное согласование синтаксиса

4) преобразование синтаксиса

5) передача данных

6) запрос завершения сеанса

Представительный уровень имеет дело с формой представления передаваемой по сети информации, не меняя её содержание. При этом передаваемая прикладным уровнем одной системы, эта информация всегда понятна прикладному уровню другой системы. Средства данного уровня обеспечивают согласованность протоколов прикладных уровней могут преодолеть синтаксические различия. Например, различие в кодах символов. На этом уровне может выполняться шифрование и дешифрование данных, что обеспечивает секретность обмена данными сразу для всех прикладных служб. При необходимости представительный уровень осуществляет трансляцию между множеством форматов представления информации, используя общий формат представления информации. Таким образом, обеспечивается независимость прикладных объектов от используемого синтаксиса. Синтаксическая независимость может быть достигнута двумя способами:

1) Представительный уровень обеспечивает общие синтаксические элементы, используемые прикладными объектами.

2) Прикладные объекты используют любой синтаксис, а на представительном уровне в этом случае осуществляется преобразование между различными формами синтаксиса и общим синтаксисом, необходимым для связи между прикладными объектами. Это преобразование является прозрачным для других открытых систем, поэтому не оказывает влияние на стандартизацию протоколов представительного уровня.

В среде взаимодействия открытых систем не существует единого синтаксиса передачи данных. Используемый синтаксис передачи согласовывается между взаимодействующими объектами. Для каждого объекта необходимо определить как синтаксис, данного прикладного объекта, так и согласованный синтаксис передачи. Для протокола представительного уровня необходимо знать только синтаксис передачи.

# 48) Прикладной уровень взаимодействия открытых систем

Эталонная модель взаимодействия открытых систем состоит из 7 уровней:

1) Физический

2) Канальный

3) Сетевой

4) Транспортный

5) Сеансовый

6) Представительский

7) Прикладной

Все функции подразделяются на 2 группы:

1. Функции, зависящие от конкретной реализации сети (сетезависимые уровни 1-3)

2. Функции, ориентированные на работу с приложениями (сетенезависимые уровни 5-7).

4-ый уровень (транспортный) является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней.

**Прикладной уровень.** Является границей между процессами сети и прикладными (пользовательскими) процессами.

Назначение уровня: обеспечить доступ прикладных процессов к среде передачи информации для обеспечения их взаимодействия при решении общей задачи.

Функции уровня:

1) идентификация партнёров

2) определение текущей готовности партнёров

3) установление полномочий для передачи

4) согласование механизма секретности

5) аутентификация партнёров

6) определение методологии, назначение цен, достаточности ресурсов, приемлимого качества обслуживания (времени ответа, уровня ошибок и т.п.

7) синхронизация взаимодействующих приложений

8) выбор дисциплины диалога

9) передача прикладных данных

10) согласование ответственности за обнаружение ошибок

11) идентификация ограничений по синтаксису данных

Прикладной уровень содержит все функции, отсутствующие на более низких уровнях, но необходимые для взаимодействия. При начале взаимодействия прикладных процессов процесс-инициатор вызывает экземпляр прикладного объекта своей открытой системы. Устанавливается ответственность данного прикладного объекта за установление соединения с прикладным объектом отрытой системы получателя.

Прикладной объект состоит из элемента пользователя и элемента прикладной службы. Выделяют 2 типа элементов прикладной службы: общие элементы – предоставляют возможности, необходимые множеству приложений; специальные элементы – предоставляют возможности, необходимые для обеспечения дополнительных услуг конкретным приложением: банковские операции, передача файлов и т.д.

Прикладной уровень определяет сетевые прикладные программы, которые обслуживают файлы. Многие сетевые программы-утилиты являются частью прикладного уровня.

# 49) Технология поиска информации в Интернет. Поисковые инструменты

Количество ресурсов, доступных в Интернете, превышает количество людей на Земле. Это количество с каждым годом увеличивается в разы. Поэтому поиск информации в Интернете является актуальной задачей. Чтобы найти нужную информацию, необходимо найти её адрес. Для этого существуют специализированные поисковые сервера (роботы индексов поисковой системы), тематические интернет-каталоги, системы мета-поиска, службы поиска людей и так далее.

*Технология поиска.*Веб-технология является специальной технологией подготовки и размещения документов в сети Интернет. В неё входят веб-страницы, электронные библиотеки, каталоги и виртуальные музеи.

Для ориентации и поиска нужной информации используют поисковые инструменты. Поисковые инструменты – это специальное программное обеспечение, основная цель которого обеспечить наиболее оптимальный и качественный поиск информации. Поисковые инструменты размещаются на специальных веб-серверах, каждый из которых выполняет следующие функции:

1. анализ веб-страниц и занесение результатов анализа на тот или иной уровень базы данных;

2. поиск информации по запросу пользователя;

3. обеспечение удобного интерфейска для поиска информации и просмотр результатов поиска.

Введём следующее понятие:

1. интерфейс поискового инструмента – это веб-страница с гиперссылками, строкой поиска (строкой подачи запроса) и инструментами активизации запроса.

2. индекс поисковой системы – это информационная база, содержащая результат анализа;

3. запрос – ключевое слово или фраза, которое пользователь вводит в строку пользователя, при этом для запросов могу использоваться специальные символы, математические символы.

Схема поиска везде одинакова. По запросу, введённому пользователем, выдаётся список гиперссылок, соответствующих запросу.

Данный список ранжируется по определённым критериям, чтобы вверху оказались документы, наиболее соответствующие запросу пользователя. Каждый из поисковых инструментов использует различные критерии ранжирования документов, как при анализе результатов поиска, так и при формировании индекса, т.е. указав в строке поиска запросы одинаковой конструкции можно получить различные результаты поиска.

Большинство поисковых инструментов предлагают два способа поиска: простой и расширенный.

Для увеличения эффективности поиска можно использовать в запросах некоторые логические операторы. С помощью операторов пользователь связывает логический запрос для получения точного результата.

Кавычки организуют поиск запроса, указанного подряд.

Простой запрос даёт ссылки на документы.

Оператор & - указывает на то, что в документе должны быть включены все ключевые слова.

Оператор not – указывает на то, что слова должны располагаться рядом.

Оператор \* - указывает на поиск слова, по его маске.

Для поиска русскоязычной информации наиболее развиты является сервис поискового сервера Яндекс. В строке поиска можно написать то, что вы ищете. Система проанализирует и обработает ваш запрос и найдёт ссылки по теме. Операторы языка запросов Яндекс можно рассмотреть <http://help.yandex.ru/search/?id=481939>

# 50) Технология поиска информации в Интернет. Поисковые машины

Это сервера с огромной базой данных Url адресов, которые автоматически обращаются к станицам www по этим адресам, изучают содержимое этих страниц. Формируют и записывают ключевые слова в свою базу данных (индексируются ) . Роботы поисковых систем переходят по встречаемым на страницах ссылкам и переиндексируют их. Так как почти любая страница www имеет множество ссылок на другие страницы, то поисковая машина может обойти теоритически все сайты интернет. Для большей эффективности поиска необходимо учитывать следующие моменты:

1. Определение темы запроса, т.е. что именно вы хотите найти.
2. Необходимо уделять внимание языку, грамматике использованию небуквенных символов и морфологии. Поисковые системы могут незначительно различатся по используемым специальным символам или операторам. Каждая поисковая система как правило имеет раздел Help где изложены все синтаксические правила и рекоментации по поиску.
3. Рекомендуется использовать различные поисковые системы.
4. Чтобы исключить определенные документы, содержащие такие термины перед каждым таким словом ставится знак -. Для того, чтобы обязательно включались некоторые ссылки использовается знак +. Каждая ссылка в списке результатов содержит так называемый спиппет – несколько строчек из найденного документа, среди которых встречаются наши ключевые слова.
5. Поисковая система – это посредник между обладателем информации (сайтом) и пользователем. Базы данных постоянно обновляются, но реальным является и отставание.

Кроме поисковых систем существует другие источники информации в сети.

# 51) Технология поиска информации в Интернет. Каталоги, подборки ссылок

Кроме поисковых систем существуют другие источники информации в сети: каталоги, поиск по которому является движение от более общих категорий к более конкретным. При переходе по рубрикам каталога, некоторые поисковые программы позволяют вводить ключевые слова.

Каталоги – постоянно обновляющийся и изменяющийся иерархический каталог, содержащий много категорий и отдельных web-серверов с кратким содержанием web-содержимого. Поиск по каталогу явл. движением от более общих к более конкретным. При переходе по рубрикам каталога, некоторые поисковые проги позволяют вводить ключевые слова.

Подборки ссылок – это отсортированные по темам ссылки. Они достаточно сильно отличаются друг от друга по наполнению.

<http://www.relcom.ru/Internet/Treasures/>

выдает список: автомобилистам, ваш дом, ваши питомцы и т.д. выбрав первую ссылку, попадаете на подборку авто и т.д. преимущество – целенаправленность, т.к. подборки вкл. в себя редкие Интернет ресурсы подобранные конкретным web-мастером.

# 52) Технология поиска информации в Интернет. Базы данных адресов

- специальные поисковые сервера, исп. классификацию по роду деятельности, по услугам, по выпускаемой продукции и т.д. Иногда имеют поиск по алфавиту. В записях этих баз данных хранится информация о сайтах, представляющих информация об электронном адресе организации за определенную плату владельцу БД.

Каждая директория содержит ссылки на сайты, в кот. находится интересующая вас информация.

# 53) Технология поиска информации в Интернет. Поиск в архивах Gopher и система поиска FTP файлов

*Gopher* – взаимосвязанная система серверов, распределенная по Интернету (Gopher - пространство). Пространство Gopher собрана богатейшая библиотека, материалы которой не доступны для просмотра в удаленном режиме. Пользователь может только просматривать иерархически собранные данные.

*система поиска FTP файлов*. Особый тип поиска информации в Интернете, позволяющий находить файлы на «анонимных» FTP серверах. Протокол FTP предназначен для передачи в сети файлов, т.е явл. аналогом Gopher. Основным критерием поиска является название файла, задаваемое разными способами. Недостаток этого поиска – содержимое файла не учитывается при поиске. Достоинство состоит в след. Необходимо найти известную программу, название которой устоялось, то с большей вероятностью данный файл будет найден. Для этого исп. FTP-сервера, один из которых является [www.filesearrch.ru](http://www.filesearrch.ru). Данный поисковый сервер ищет файлы на FTP-серверах по именам файлов и каталогов.

# 54) Технология поиска информации в Интернет. Система поиска в конференциях, системы мета-поиска, системы поиска людей

*Система поиска в конференциях.* USENET NEWS – система телеконференций – сообщество сетей Интернет. На западе данный сервис называется NEWS- новости. С точки зрения абонента USENET представляет из себя доску объявлений, где можно найти статью на любую тему, т.к. статьи занимают место на серверах, поэтому они уничтожаются, освобождая место для новых. Наилучшим сервисом явл. сервер Google Groups – бесплатное интерактивное сообщество и служба групп обсуждений, предлагающая самый обширный в интернет архив сообщений сети USENET.

Пользователь набирает строку запроса и по этим ключевым словам формируется список конференций. Как правило требуется подписка.

*системы мета-поиска.* Исп. для быстрого поиска в БД сразу нескольких поисковых систем. Данный сервис посылает запрос на большое кол-во разных поисковых систем, обрабатывает рез-тат, удаляет повторяющиеся адреса. В рез-те получает более широкий спектр того, что имеется в Интернет. Наиболее популярная система мета-поиска: <http://www.search.com>, которая включает в себя более 2-х десятков поисковых систем. Основным недостатком данного сервиса явл. его нестабильность.

*Системы поиска людей.* – это спец. сервера, позволяющие осуществлять поиск людей в интернет. Пользователь указывает ФИО и получает. Данный сервис берет информацию из открытых источников. Одним из таких серверов явл.: hthttp://www.whowhere.com.

Сервера, обеспечивающие поиск людей, содержат БД на несколько 10 млн адресов.