Вопросы:

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

2.  Количественная мера информации. Свойства количества информации.

3.  Обобщенная модель канала передачи информации.

4.  Эталонная модель взаимосвязи открытых систем.

5.  Цели и виды преобразования информации. Кодирование, модуляция.

6.  Условия взаимной однозначности алфавитного кодирования. Эффективное кодирование.

7.  Арифметическое кодирование при известной статистике сообщения.

8.  Метод Хаффмена для сжатия при неизвестной статистике сообщения

9.  Неравенство Крафта.

10.  Средняя длина кодового слова и энтропия.

11.  Первая теорема Шеннона.

12.  Сжатие данных, эффективность сжатия.

13.  Математическая модель системы связи.

14.  Блочные коды. Избыточность кода. Расстояние Хэмминга. Вес слова

15.  Помехозащитные коды (совершенные, квазисовершенные, полиномиальные и циклические).

16.  Основы теории защиты информации.

Ответы:

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

Три составляющие теории информации:

1)     Теория эффективного кодирования

2)     Теории помехоустойчивого кодирования

3)     Теория криптографического кодирования

Задачи теории информации: при известной статистике шумов выбрать такое множество передаваемых сигналов, чтобы правдоподобие правильного декодирования, принимаемого сообщения, было максимальным.

Методы ТИ используются:

1) Для защиты данных в памяти вычислительных устройств.

2) Передачи данных:

·   В вычислительных системах

·   В цифровых оптических дисках

·   В системах со сжатием данных

·   В системах связи с ограничением на передаваемую информацию( в системах где увеличение мощности обходится очень дорого)

·   В системах цифрового телевидения и обработки изображения

·   В системах передачи информации разного назначения

Информация (в широком смысле) – абстрактное понятие , имеющие много значений в зависимости от контекста.

Информация – сведения независимо от формы и представления(сообщения, данные и т.д).

Информация в информатике – осознанные сведения об окружающем мире , кот. является объектом хранения, преобразования, передачи и использования.

Основные виды информации по форме представления, способов ее кодирования и хранения:

1)     Графическая или изобразительная;

2)     Звуковая (акустическая);

3)     Текстовая;

4)     Числовая;

5)     Видеоинформация.

Качественные свойства информации:

1)     Объективность информации;

2) Достоверность информации. Информация может быть недостоверной из-за:

·   Преднамеренного искажения;

·   Искажения в результате воздействия помех;

3) Полнота информации;

4) Точность информации;

5) Актуальность информации;

6) Полезность информации.

Сообщение – это совокупность знаков, отражающих ту или иную информацию.

Сигнал – это физический процесс, отображающий (несущий) передаваемое сообщение.

2.  Количественная мера информации. Свойства количества информации.

Формула для информации: I = -log p.

Энтропия источника информации – это средняя информация, полученная для всех возможных событий. Определяется как математическое ожидание количества информации.

   Е = - ∑ pi\*logpi;

   Pi – частота элемента.

         Log2 – результат в битах.

         Ln – результат  в натах.

         Log10 – результат в дитах.

         Свойства энтропии:

1)     Всегда неотрицательна(>= 0).

2) Энтропия сообщения с одним элементом всегда равна нулю.

3) График энтропии всегда выпуклый.

4) Для дискретного источника без памяти с равновероятностными элементами энтропия увеличивается с увеличением размера алфовита.

Энтропия показывает хаотичность информации, чем она выше,тем выше неопределенность системы.

Емкость дискретного источника:

H0 = log2m; m – кол-во уникальных символов

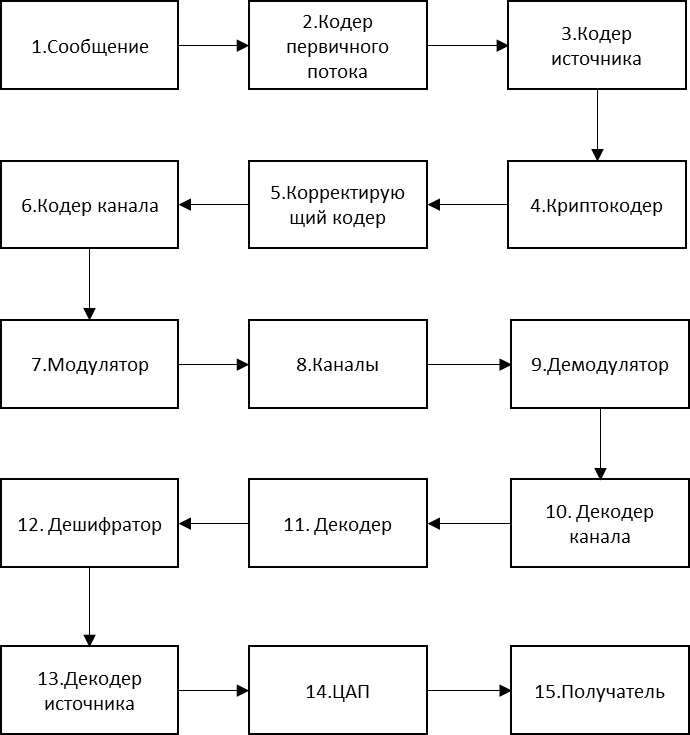
         Относительная избыточность:

                     r = 1 – (E/H0) , где E – энтропия , H0 – емкость дискретного источника.

         Обычная избыточность:

                     R = H0 – E , где E – энтропия , H0 – емкость дискретного источника.

3.  Обобщенная модель канала передачи информации.



1) Источник сообщений – формирует поток непрерывных или дискретных сообщений.

2)  Кодер первичного потока – позволяет представить информацию в форме, позволяющий упростить дальнейшую обработку.

3) Кодер источника сообщений – предназначен для устранения информационной избыточности (позволяет более эффективно использовать частотный ресурс и повысить скорость передачи информации).

4) Криптокодер – преобразует информацию с целью защиты от несанкционированного (случайного) доступа.

5)  Корректирующий кодер – вводит информационную избыточность, с целью обнаружения и/или исправления ошибок сравнительно небольшой кратности.

6) Кодер канала – предназначен для минимизации влияния помех на передаваемую информацию.

7) Модулятор – преобразует множество дискретных сигналов канального кодера в непрерывные сигналы.

11) Декодер – восстанавливает ту избыточность, которая была ранее устранена на передающей стороне.

В канале формируются сигналы и помехи вида:

Y(t)=X(t)\* μ(t)+ n(t), где:

·   X(t) – передаваемый непрерывный сигнал

·   μ(t) – мультипликативная помеха

·   n(t) – аддитивная помеха

4.  Эталонная модель взаимосвязи открытых систем.

7-уровневая архитектура модели канала:

1. На физическом уровне реализуется канал.

2. На канальном уровне реализуются процедуры по кодированию, сжатию и шифрованию.

3. На сетевом уровне реализуется передача информации от источника к адресату.

4. Транспортный уровень управляет сквозной передачей пакетов с коррекцией ошибок.

5. Сеансовый уровень контролирует соединение между оконченными системами.

6. На уровне представления выполняются операции сжатия данных,  защиты информации, преобразования форматов, для обеспечения эффективного и безопасного взаимодействия.

7. Прикладной уровень предоставляет различные сетевые службы.

5.  Цели и виды преобразования информации. Кодирование, модуляция.

Преобразование информации - изменение формы и вида сведений.

Основными целями преобразования информации являются:

1. Обеспечение интерфейса, необходимого для реализации информационного взаимодействия различных функциональных элементов информационных систем (например, согласование источника информации с характеристиками канала связи).
2. Извлечение полезной информации, содержащейся в сигнале.
3. Устранение исходя из требований по качеству измененных данных, содержащихся в сигнале (редукция). Обеспечение специальных свойств, позволяющих повысить скорость передачи, помехоустойчивость, секретность (конфиденциальность).

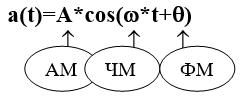
Основные виды преобразования информации. К основным видам преобразования относятся:

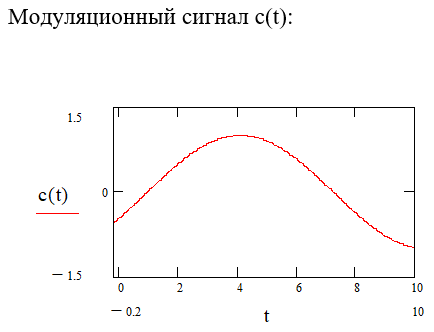
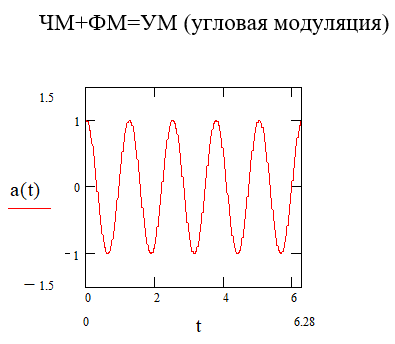
1. Функциональное преобразования (изменение характеристик сигнала без потерь полезной информации).
2. Квантование сигнала по уровню. Дисперсия шума квантования Dδ. Если δ1=…=δi=…=δk=δ, то дисперсия шума определяется формулой: Dδ=δ2/12
3. Дискретизация сигнала во времени.

Теорема Котельникова: Если ширина спектра сигнала равна *f*c, то сигнал без потерь информации может быть передан своими дискретными отсчетами через интервал времени: Δt=½\**f*c

1. Кодирование (преобразование информации с одной из следующих целей: шифрование, сжатие, защита от шума):
   1. интерфейсное;
   2. статистическое;
   3. помехоустойчивое (цель - борьба с шумами).
2. 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):
   1. модуляции гармонических колебаний (амплитудная, частотная, фазовая);
   2. импульсная модуляция ( амплитудно-импульсная, широтно-импульсная, частотно-импульсная, фазово-импульсная).
3. Сжатие (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 информации с устройства ее хранения будет проходить быстрее)).

Импульсно-кодовая модуляция предполагает дискретизацию сигнала во времени и квантование по уровню.





6.  Условия взаимной однозначности алфавитного кодирования. Эффективное кодирование.

Кодирование связано с преобразованием выходных символов дискретного источника в последовательность символов заданного кодового алфавита.

Модель системы передачи информации с кодированием:

1. Источник информации
2. Кодер
3. Канал
4. Декодер
5. Получатель

Код источника - это множество дискретных последовательностей всех событий, представленных символами кодового алфавита.

Кодовыми словами (векторами) называются последовательности символов.

Длина кода (N) - число символов кодового слова.

Параметр N определяют следующие виды кодов:

равномерный (N = const) / неравномерный (N = var).

Правила кодирования должны отвечать следующим требованиям:

1) Необходимо добиваться высокой вероятности однозначного (правильного) декодирования исходной информации дискретного источника по закодированной последовательности.

2) Число символов кода, требуемых на 1 символ источника, должно быть минимальным.

Код является эффективным, если он имеет наименьшую среднюю длину кодового слова (математическое ожидание = энтропии).

Условия однозначности взаимного кодирования связано с преобразованием выходных символов дискретного источника, последовательных символов заданного кодового алфавита.

Код источника - множество дискретных последовательностей всех событий, представленных символами кодового алфавита.

В качестве кодового алфавита часто используются символы двоичного (0 и 1) или бинарного (-1 и 1).

Многие алгоритмы эффективного кодирования в качестве однозначного декодирования кода используют префиксные моментальные коды.

Если процесс однозначного декодирования каждого кодового слова осуществляется сразу после приёма всех символов кодового слова и принимается решение о соответствующем символе источника, то код с таким символом называется моментальным.

Для ответа на вопрос, будет ли предлагаемое множество кодовых слов кода при декодировании точно соответствовать исходной информации источника применяется неравенство Крафта.

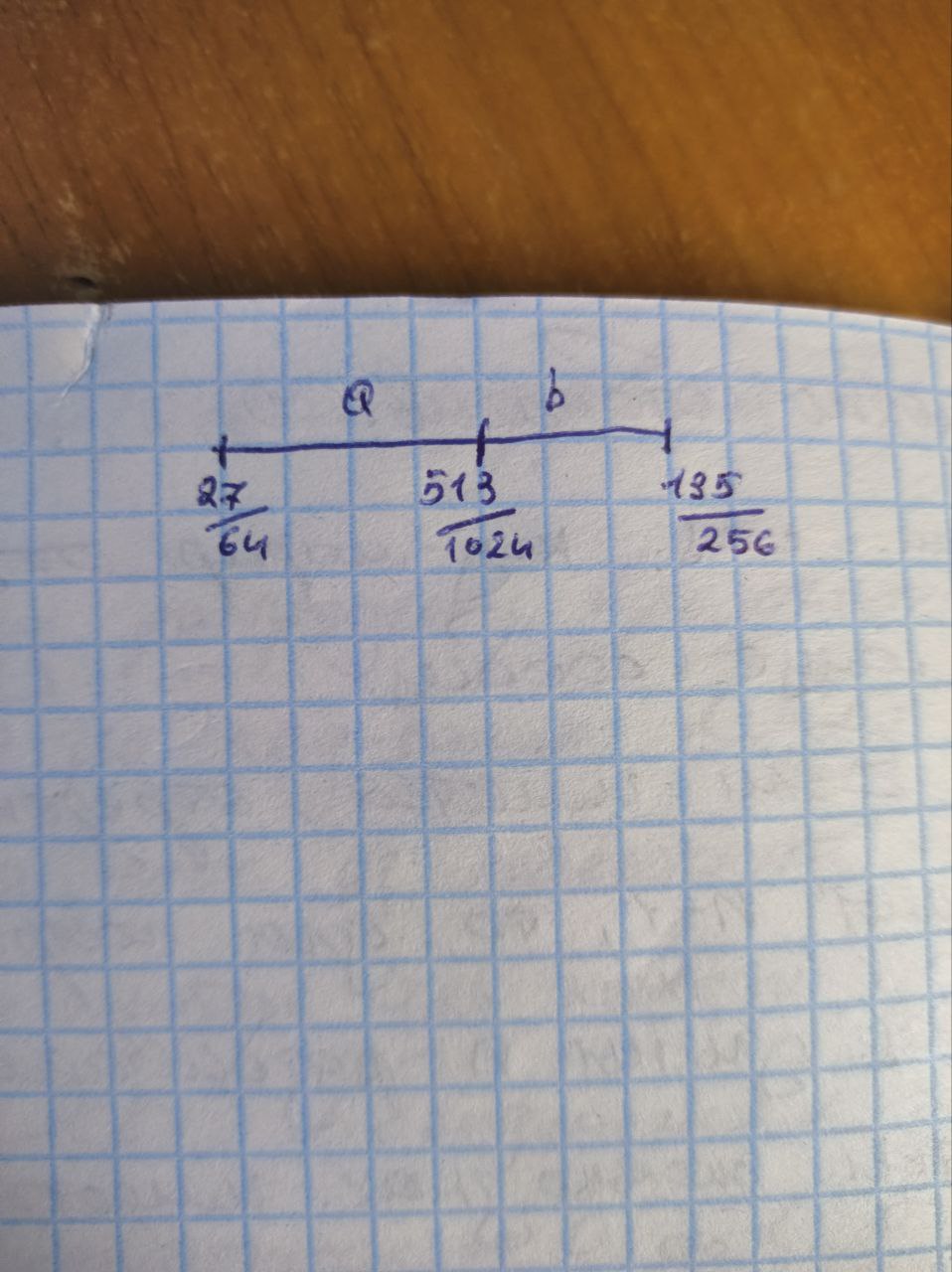
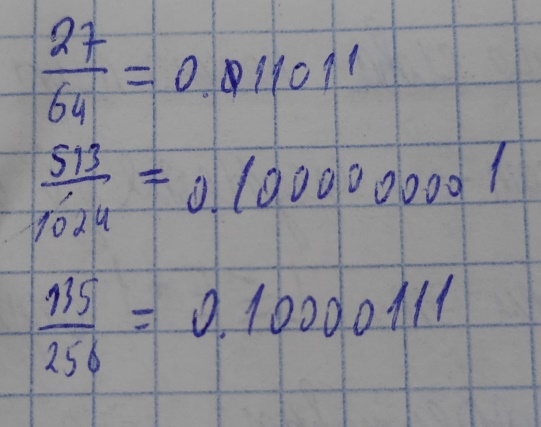
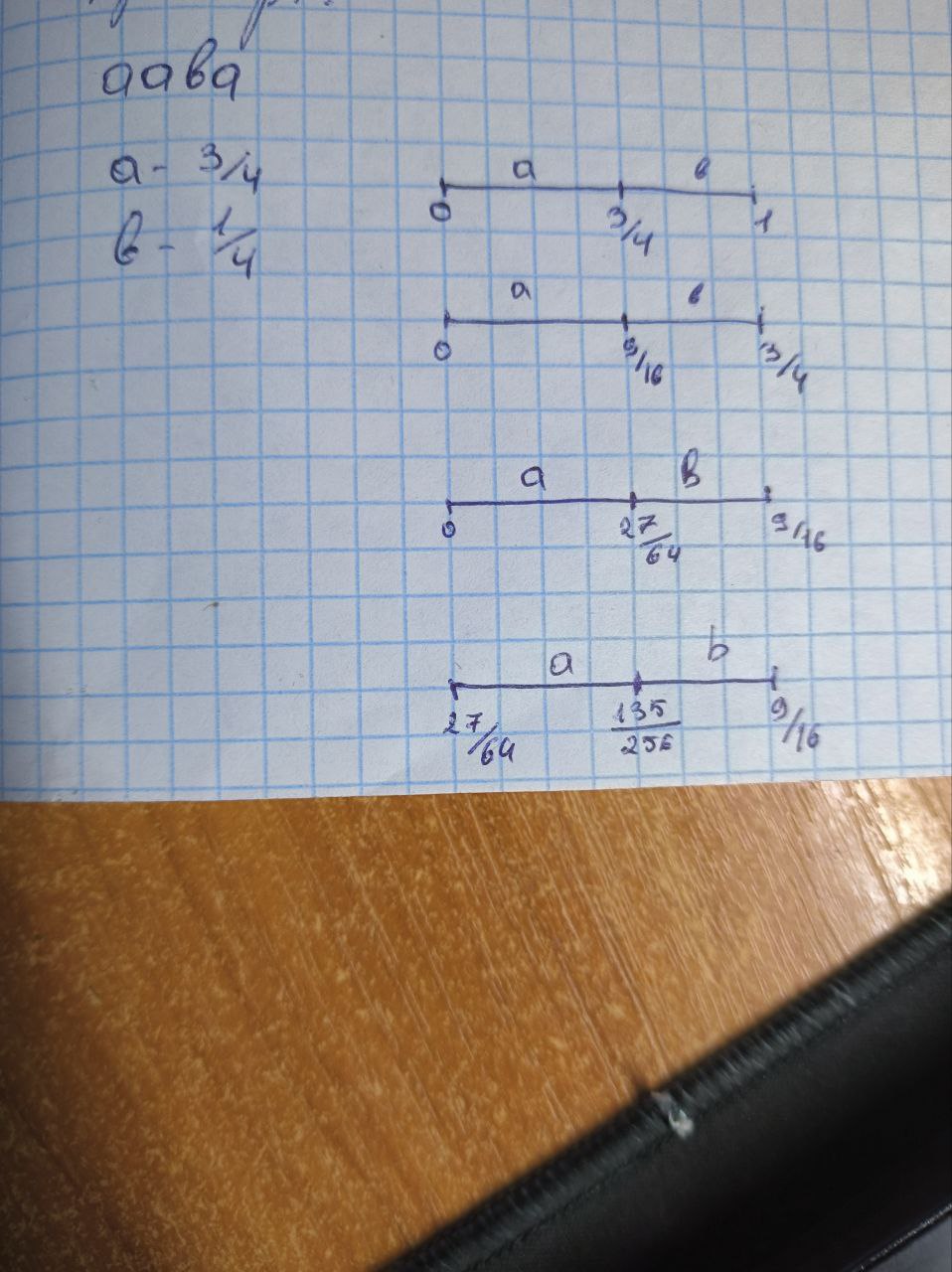
7.  Арифметическое кодирование при известной статистике сообщения.

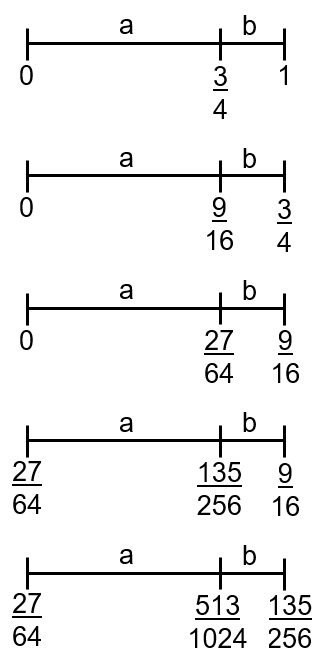
По исходному распределению вероятностей для выбранной для кодирования ДСВ строится таблица, состоящая из пересекающихся только в ограниченных точках отрезках для каждого из значений этого ДСВ. Объединением этих отрезков должен образовываться отрезок от 0 до 1, а их длины должны быть пропорциональны вероятностям соответствующих значений ДСВ.

Алгоритм кодирования заключается в построении отрезка однозначно определяющего данную последовательность значений ДСВ. Затем для построения отрезка находится число принадлежащее его внутренней части и равное целому числу, отделённому на минимально возможную положительную степень 2. Это число и будет кодом для рассматриваемой последовательности. Все возможные конкретные коды этого числа >0 и <1, поэтому можно отбрасывать 0 и “,”. Но нужен ещё один специальный код-маркер, сигнализирующий о конце сообщения.

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 ДСВ. Это разбиение делается так же, как и первое. Затем выбирается из полученных отрезков тот, который соответствует заданной конкретной последовательности длины n.

Пример:





aaba

a - 3/4

b - 1/4

27/64 = 0.11011

135/256 = 0.10000111

513/1024 = 0.1000000001

8.  Метод Хаффмана для сжатия при неизвестной статистике сообщения.

1. строится частотная таблица, которая сортируется по возрастанию;
2. все элементы выносятся как конечные узлы (листья) дерева;
3. берем два минимальных узла, получаем новый, его сумма - сумма двух узлов;
4. повторять пункт 3, пока не останется 1 узел - корень дерева;
5. после того, как дерево было построено, каждой его ветви  присваивается значение “0” или “1” (все ветви должны быть симметричны).

1. с увеличением степени блокового источника значение средней длины слова кода Хаффмана уменьшается и стремится к энтропии одиночного источника;
2. с увеличением степени блокового источника увеличивается эффективность кода или сжатия источника;
3. недостатком алгоритма Хаффмана является требование априорного знания значений появлений.

Адаптивный алгоритм Хаффмана:

Адаптивный алгоритм эффективного кодирования реализуется с помощью двух операций:

1. в начале выполняется кодирование источников предположений, когда все символы имеют равные вероятности появления;
2. по мере накопления знаний о статистических характеристиках источника выполняется кодирование по алгоритму Хаффмана.

Универсальный алгоритм сжатия

Универсальный алгоритм сжатия не требует предварительных знаний статистики символов источника (является адаптивным). Данный алгоритм основан на идее использования словаря последовательности символов образцам последовательных символов, встречающихся в текстах, изображениях, данных ставятся в соответствие кодовые слова. При этом каждое кодовое слово представлено определенным индексом. В процессе сжатия последовательности в несжатых данных заменяются кодовыми словами, которые имеются в словаре. Чем больше объем словаря, тем больше объем сжатия. Метод эффективен при архивации текстовых данных, изображений.

9.  Неравенство Крафта.

Для ответа на вопрос, будет ли предлагаемое множество кодовых слов кода при декодировании точно соответствовать исходной информации источника, применяется неравенство Крафта.

Для построения однозначно декодируемого q-ичного кода, содержащего m кодовых слов с длинами n1, n2…nm, необходимо и достаточно, чтобы выполнялось неравенство Крафта:

i = 1mq-n1 , где q - размерность алфавита.

Для двоичного алфавита q = 2, неравенство Крафта записывается как:

i = 1m2-n1

Пример:

Дан алфавит из 5-ти символов, каждый закодирован с помощью кода длиной 3. Удовлетворяет ли код неравенству Крафта?

i = 152-31 - удовлетворяет(данный код однозначно декодируемый).

10.  Средняя длина кодового слова и энтропия.

Мерой эффективности кода является его средняя длина кодовых слов:

Ln=i=1mpili , где m - число символов источника с n - кратным расширением источника одиночных символов; Pi - вероятность символа с n - кратным расширением источника одиночных символов, l - длина кода.

Очевидно, более эффективной является та информационная система, которая использует коды с минимально возможными длинами кодовых слов.

Понятие энтропии источника как среднее количество информации, передаваемой одним символом источника, отражает связь величины энтропии с величиной средней длины слов эффективного кода.

Соотношение первое: неравенство длин кода - средняя длина L двоичного однозначно декодируемого кода удовлетворяет неравенство LE. Это выражение определяет минимально достижимую среднюю длину эффективного кода.

Соотношение второе: пределы средней длины кодового слова можно получить кодированием источника двоичным кодом, у которого средняя длина кодового слова удовлетворяет выражению: ELE+1.

При этом, выражение LE+1 определяет максимально возможное значение средней длины двоичного эффективного кода.

11.  Первая теорема Шеннона.

Теорема Шеннона о кодировании для канала без шума.

Если в начале передачи информации вероятность P стремится к нулю, т.е ошибки маловероятны, основное требование к информационной системе – это представление символов источника  в максимально компактной форме .

1-ая теорема Шеннона определяет минимально достижимую длину кодового слова на символ источника. Учитывая статистические свойства источника, можно более эффективно передавать или обрабатывать информацию. Если высоковероятностным символам поставить в соответствие более короткие длины кодовых слов,  а маловероятным – слова большей длины, то скорость передачи информации увеличится.Также увеличится количество передаваемой информации.

Энтропия блокового источника: пусть имеется источник блоковых символов Xn, эти блоки имеют длину n. Первая теорема Шеннона утверждает, что если n->+∞, можно произвести кодирование блоковых символов источника Xn кодом, у которого средняя длина кодового слова будет приближаться к энтропии источника X. EMLE+1.

Пусть есть последовательность символов n, тогда перепишем формулу выше как nELnE+1 EML/nE+1/n, где ML/n - средняя длина слова на один символ источника. Таким образом, расширяя источник одиночных символов, т. е. увеличивая значение n (длину блока) при n->+∞ можно закодировать символы блокового источника кодовыми словами со средней длиной на символ источника X, приближенный к значению E источника X.

Кодируя блоковые последовательности источника без памяти X c энтропией E , можно построить q-ичный префиксный код, в котором средняя длина кодового слова удовлетворяет выражению:

где Ln -  средняя длина кодового слова.

Замечание! Составляющая    , исходящая из выражения выше, определяет максимальное значение средней длины кодового слова префиксного кода.

Вывод: эффективное кодирование информации требует использование источника с n-кратным расширением источника X.

12.  Сжатие данных, эффективность сжатия.

В настоящее время большая часть передаваемых, хранимых и преобразуемых данных соответствуют звуковой, графической или видеоинформации. В этом случае реализация современных инфокоммуникационных систем неизбежно усложняется: увеличиваются технические затраты на хранение данных, предъявляются более высокие требования по экономии канального частотного ресурса.

Сжатие данных позволяет уменьшить объем данных, используется для представления информации. Даже при постоянном росте емкости хранилищ данных сжатие остается важным компонентом информационных технологий.

Различают два основных метода кодирования данных, используемых для сжатия: с потерями и кодирование без потерь.

Идея сжатия основывается на возможности устранения или уменьшения избыточности данных.

Сигнал, несущий информацию можно сжать, удалив из него имеющуюся избыточность.

Различают 2 вида избыточности:

1) видео/аудио избыточность (субъективная), которую можно устранить с некоторой потерей информации, сравнительно мало влияющей на качество воспроизводимого изображения или звука.

2) статическая избыточность, связанная с корреляцией и предсказуемостью обработки данных. Такая избыточность может быть полностью устранена без потери информации и исходные данные могут быть полностью восстановлены.

Технология сжатия осуществляется за счёт применения эффективного кодирования данных из источника. В этом случае под избыточностью понимается частое повторение символов информационного сообщения. Например: кодирование с помощью алгоритмов Хаффмана, Шеннона-Фано.

Эффективное сжатие оценивается следующим коэффициентом: K = N - MN, где N - затраты на передачу/хранения без сжатия, M - затраты на передачу / хранения со сжатием.

Пример: исходное изображение N = 512 байт, сжатое с помощью MPEG2 - M = 128 байт, тогда K = (512 - 128)/512 = 0,75

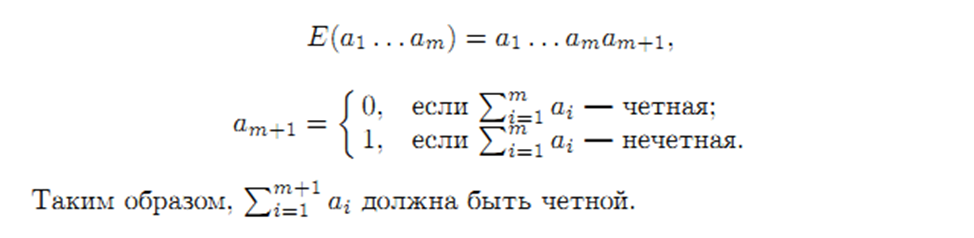
Преобразуем формулу: KN=N-M N=KN+MNM=KNM+1

Отношение вида NM - характеризует выигрыш в записи данных.

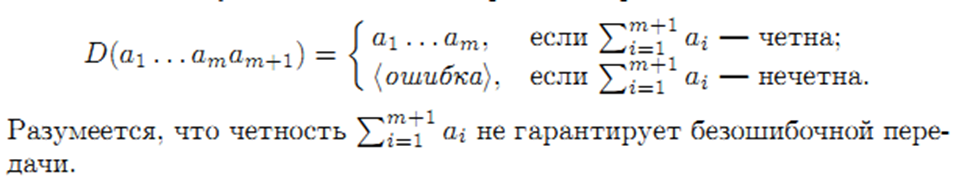
Эффективностью кода (фактор сжатия) называется отношение Е к средней длине кода.

13.  Математическая модель системы связи.

Коды делятся на два больших класса. Коды с исправлением ошибок имеют целью восстановить с вероятностью, близкой к единице, посланное сообщение. Коды с обнаружением ошибок имеют целью выявить с вероятностью, близкой к единице, наличие ошибок.

Простой код с обнаружением ошибок основан на схеме проверки четности, применимой к сообщениям a1 . . . am любой фиксированной длины m. Схема кодирования определяется следующими формулами:  

Соответствующая схема декодирования тривиальна:

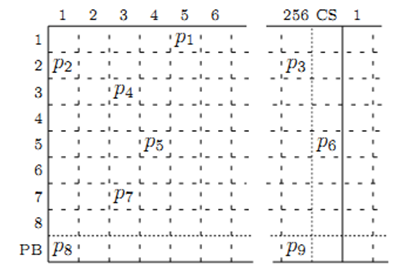


Рассмотрим (m, 3m)-код с тройным повторением. Коды с повторениями очень неэффективны, но полезны в качестве теоретического примера кодов, исправляющих ошибки. Любое сообщение разбивается на блоки длиной m каждое и каждый блок передается трижды — это определяет функцию E. Функция D определяется следующим образом. Принятая строка разбивается на блоки длиной 3m. Бит с номером i    в декодированном блоке получается из анализа битов с номерами i, i+m, i+2m в полученном блоке: берется тот бит из трех, который встречается не менее двух раз. Вероятность того, что бит в данной позиции будет принят трижды правильно равна p3. Вероятность одной ошибки в тройке равна  3p2q. Поэтому вероятность правильного приема одного бита равна p3 + 3p2q. Аналогичным образом получается, что вероятность приема ошибочного бита равна q3 + 3pq2.

Тройное повторение обеспечивает исправление одной ошибки в каждой позиции за счет трехкратного увеличения времени передачи.

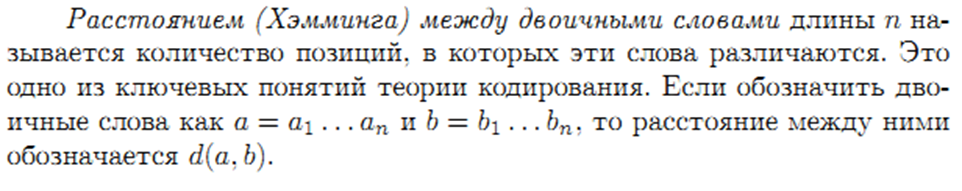
14.  Блочные коды. Избыточность кода. Расстояние Хэмминга. Вес слова.

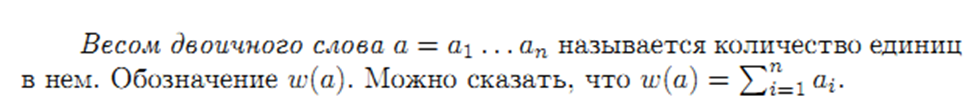
Рассмотрим (2048, 2313)-код, используемый при записи данных на магнитофонную ленту компьютерами Apple II. К каждому байту исходных данных прибавляется бит четности и, кроме того, после каждых таких расширенных битом четности 256 байт добавляется специальный байт, также расширенный битом четности. Этот специальный байт, который называют контрольной суммой (check sum), есть результат применения поразрядной логической операции “исключающее ИЛИ” (XOR) к 256 предшествующим расширенным байтам. Этот код способен как обнаруживать ошибки нечетной кратности в каждом из отдельных байт, так и исправлять до 8 ошибок в блоке длиной 256 байт. Исправление ошибок основано на том, что если в одном из бит одного из байт 256 байтового блока произойдет сбой, обнаруживаемый проверкой четности, то этот же сбой проявится и в том, что результат операции “исключающее ИЛИ” над всеми соответствующими битами блока не будет соответствовать соответствующему биту контрольной суммы. Сбойный бит однозначно определяется пересечением сбойных колонки байта и строки бита контрольной суммы. На рисунке изображена схема участка ленты, содержащего ровно 9 ошибок в позициях, обозначенных p1, p2, . . ., p9. Расширенный байт контрольной суммы обозначен CS, а бит паритета (в данном случае четности) — PB (parity bit). Ошибка в позиции p1 может быть исправлена. Ошибки в позициях p4, p5, p6, p7 можно обнаружить, но не исправить. Ошибки в позициях p2, p3, p8, p9 невозможно даже обнаружить.



Приведенные ранее примеры простейших кодов принадлежат к классу блочных. 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.

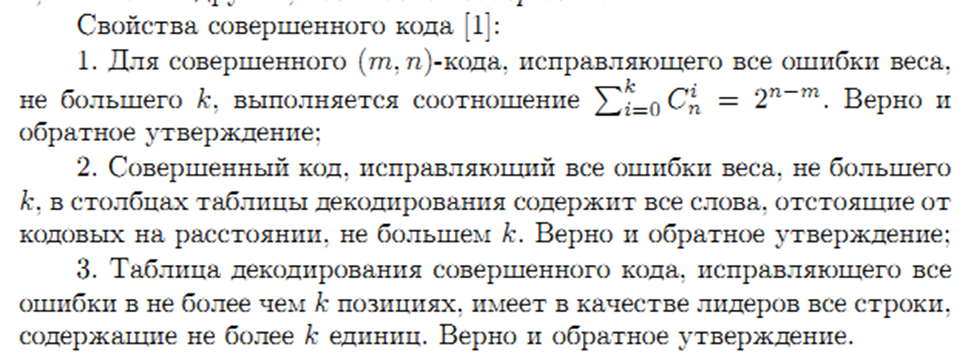
Понятие избыточности означает, что фактическая энтропия кода или сообщения (E) меньше, чем максимально возможная энтропия (Emax), т. е. число символов в сообщении или элементов в символе кода больше, чем это требовалось бы при полном их использовании.





15.  Помехозащитные коды (совершенные, квазисовершенные, полиномиальные и циклические).

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



Совершенный код — это лучший код, обеспечивающий максимум минимального расстояния между кодовыми словами при минимуме длины кодовых слов. Совершенный код легко декодировать: каждому полученному слову однозначно ставится в соответствие ближайшее кодовое.

Если m, n и k не удовлетворяют условию совершенности, то лучший групповой код, который им соответствует называется квазисовершенным, если он исправляет все ошибки кратности, не большей k, и некоторые ошибки кратности k + 1. Квазисовершенных кодов очень мало, как и совершенных.

Двоичный блочный (m, n)-код называется оптимальным, если он минимизирует вероятность ошибочного декодирования. Совершенный или квазисовершенный код — оптимален. Общий способ построения оптимальных кодов пока неизвестен.

При полиномиальном кодировании каждое сообщение отождествляется с многочленом, а само кодирование состоит в умножении на фиксированный многочлен. Полиномиальные коды — блочные и отличаются от рассмотренных ранее только алгоритмами кодирования и декодирования.

Циклический избыточный код (Cyclical Redundancy Check—CRC) имеет фиксированную длину и используется для обнаружения ошибок. Наибольшее распространения 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.

16.  Основы теории защиты информации.

Криптография (тайнопись) — это раздел математики, в котором изучаются и разрабатываются системы изменения письма с целью сделать его непонятным для непосвященных лиц. Известно, что еще в V веке до нашей эры тайнопись использовалась в Греции. В современном мире, где все больше и больше услуг предоставляется через использование информационных технологий, проблема защиты информации методами криптографии имеет первостепенное значение. Сегодня большая часть обмена информацией проходит по компьютерным сетям и часто (в бизнесе, военным и прочее) нужно обеспечивать конфиденциальность такого обмена. Теоретические основы классической криптографии впервые были изложены Клодом Шенноном в конце 1940-х годов.

Простейшая система шифрования — это замена каждого знака письма на другой знак по выбранному правилу. Юлий Цезарь, например, заменял в своих секретных письмах первую букву алфавита на четвертую, вторую — на пятую, последнюю — на третью и т.п., т.е. A на D, B на E, Z на C и т.п. Подобные шифры, называемые простой заменой или подстановкой, описаны в рассказах “Пляшущие человечки” А. К. Дойла, “Золотой жук” Э. По и других.

Шифры простой замены легко поддаются расшифровке, при знании исходного языка сообщения, т.к. каждый письменный язык характеризуется частотой встречаемости своих знаков. Например, в английском языке чаще всего встречается буква E, а в русском — О. Таким образом, в шифрованном подстановкой сообщении на русском языке самому частому знаку будет с большой вероятностью соответствовать буква О. Вероятность будет расти с ростом длины сообщения.

Усовершенствованные шифры-подстановки используют возможность заменять символ исходного сообщения на любой символ из заданного для него множества символов, что позволяет выровнять частоты встречаемости различных знаков шифра, но подобные шифры удлиняют сообщение и замедляют скорость обмена информацией.

В шифрах-перестановках знаки сообщения специальным образом переставляются между собой, например, записывая сообщение в строки заданной длины и беря затем последовательность слов в столбцах в качестве шифра.

Шифры-перестановки в общем случае практически не поддаются дешифровке. Для их дешифровки необходимо знать дополнительную информацию. Крупный недостаток подобных шифров в том, что если удастся каким-то образом расшифровать хотя бы одно сообщение, то в дальнейшем можно расшифровать и любое другое. Модификацией шифров-перестановок являются шифры-перестановки со словом ключом, которое определяет порядок взятия слов-столбцов.

Если в качестве ключа использовать случайную последовательность, то получится нераскрываемый шифр. Проблема этого шифра —это способ передачи ключа

Extra. Алгоритмы говна по жопе

Алгоритм эффективного кодирования Лемпеля-Зива (LZ77)

Метод LZ77 является основой алгоритмов сжатия zip, arch, а также в растровом формате png.

Процесс кодирования LZ77 начинается сразу после поступления входных источников на вход кодера:

1. передающая сторона записывает в специальный буфер поиска то, что было уже отправлено, принимающая - то, что было уже получено для осуществления декодирования;
2. при подготовки следующего фрагмента текста передающая сторона находит в ранее переданном объекте образцы;
3. далее идет процесс передачи на самих образцов, а только информации об этих образцах в виде ссылок;
4. ссылка записывается в форме трех указателей: xyz, x - указывает относительный адрес образца в буфере поиска, y - обозначает длину образца, z - следующую букву в буфере.

Пример

Строка: декодирование кода

|  |  |  |
| --- | --- | --- |
| 1) | декодирование кода | 0, 0, д |
| 2) д | екодирование кода | 0, 0, е |
| 3) де | кодирование кода | 0, 0, к |
| 4) дек | одирование кода | 0, 0, о |
| 5) деко | дирование кода | 4, 1, и |
| 6) декоди | рование кода | 0, 0, р |
| 7) декодир | ование кода | 4, 1, в |
| 8) декодиров | ание кода | 0, 0, а |
| 9) декодирова | ние кода | 0, 0, н |
| 10) декодирован | ие кода | 6, 1, е |
| 11) декодирование | [пробел]кода | 0, 0, [пробел] |
| 17) декодирование кода |  | 12, 3, а |

Метод кодирования LZ77 приводит к сжатию тогда, когда затраты становятся меньше, чем код ASCII.

Декодирование LZ77 осуществляется декодированием каждого кодового слова по идентичному слову словаря.

|  |  |  |
| --- | --- | --- |
| 1) 0, 0, д |  | 0 |
| 2) 0, 0, е | д | де |
| 3) 0, 0, к | де | дек |
| 4) 0, 0, о | дек | деко |
| 5) 4, 1, и | деко | декоди |

Пример LZSS

Строка: карабас-барабас

|  |  |  |
| --- | --- | --- |
| словарь | буфер | код |
| 1) | карабас-барабас | 0, к |
| 2) к | арабас-барабас | 0, а |
| 3) ка | рабас-барабас | 0, р |
| 4) кар | абас-барабас | 1, (2, 1) |
| 5) кара | бас-барабас | 0, б |
| 6) караб | ас-барабас | 1, (2, 1) |
| 7) караба | с-барабас | 0, с |
| 8) карабас | -барабас | 0, - |
| 9) карабас- | барабас | 1, (4, 2) |
| 10) карабас-ба | рабас | 1, (8, 5) |

Пример LZ78

Строка: карабас-барабас

|  |  |  |
| --- | --- | --- |
| позиция словаря | словарь | код |
| 0 | “” | - |
| 1 | к | 0, к |
| 2 | а | 0, а |
| 3 | р | 0, р |
| 4 | аб | 2, б |
| 5 | ас | 2, с |
| 6 | - | 0, - |
| 7 | б | 0, б |
| 8 | ар | 2, р |
| 9 | аба | 4, а |
| 10 | с | 0, с |

Пример LZW

Строка: карабас-барабас

|  |  |  |
| --- | --- | --- |
| входная фраза | код | позиция словаря |
| “”алфавит-[пробел] |  | 0-35 |
| ка | 12 | 36 |
| ар | а | 37 |
| ра | 18 | 38 |
| аб | 1 | 39 |
| ба | 2 | 40 |
| ас | 1 | 41 |
| с- | 19 | 42 |
| -б | 34 | 43 |
| бар | 40 | 44 |
| раб | 38 | 45 |
| бас | 40 | 46 |
| с | 0, c |  |