Федеральное агентство связи ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования «Московский технический университет связи и информатики»

Кафедра «‎Информатика»

Курсовая работа по дисциплине «‎Основы программирования»

На тему «Основы теории информации и криптографии»

Выполнил студент

Группы БСТ2001

Уитман Э.У.

Проверил доц. кафедры

«‎Информатика» Гуриков С.Р.

Москва 2020

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc54884307)

[**ТЕХНИЧЕСКОЕ ЗАДАНИЕ** 5](#_Toc54884308)

[**Глава 1. Теоретическая часть** 7](#_Toc54884309)

[**1.1** **Основные понятия** 7](#_Toc54884310)

[**1.2** **Система передачи информации** 9](#_Toc54884311)

[**1.3** **Постулаты теории информации** 11](#_Toc54884312)

[**1.4** **Основные принципы кодирования** 12](#_Toc54884313)

[**1.5** **Принципы кодирования** 13](#_Toc54884314)

[**1.6** **Алгоритмы сжатия без потерь** 18](#_Toc54884315)

[**1.7** **Методы сжатия источников без памяти** 19](#_Toc54884316)

[**1.8** **Методы сжатия источников c памятью** 22](#_Toc54884317)

[**1.9** **Основные термины и понятия криптологии** 24](#_Toc54884318)

[**1.10** **Основные понятия криптоанализа** 27](#_Toc54884319)

[**1.11** **Шенноновские модели криптографии** 27](#_Toc54884320)

[**1.12** **Система шифрования RSA** 30](#_Toc54884321)

[**1.13** **Вывод** 33](#_Toc54884322)

[**Cписок использованных источников** 34](#_Toc54884323)

**ПОДПУНКТОВ ТОЛЬКО ТРИ ДОЛЖНО БЫТЬ!**

# **ВВЕДЕНИЕ**

**Актуальность темы.** В настоящее время невозможно представить повседневный образ жизни человека без высокоскоростных модемов, телефонных каналов, глобальной сети Интернет, лазерных компакт-дисков, flashкарт и съемных винчестеров большой ёмкости для персональных компьютеров, мобильных сотовых телефонов, кредитных карт и много другого, что связано с передачей или хранением информации.

**АБЗАЦЫ БУДУТ??Характерно и то**, что источники информации являются дискретными, информация кодируется, хранится и декодируется в цифровой форме. **Кроме того**, необходимо отметить, что большие объёмы информации не передаются по каналам связи в первозданном виде, а должны эффективно кодироваться (сжиматься), а иногда и шифроваться.

Понимание того, как осуществляется передача информации, ее сжатие и шифрование важно для любой работы в информационном обществе, особенно в ИТ секторе. Именно по причинам, перечисленным выше, освоение теории информации и криптографии как никогда актуально в настоящий момент. Я НЕ ВИЖУ, ЧТО ИСПОЛЬЗОВАЛСЯ ОБРАЗЕЦ ВВЕДЕНИЯ

**Объектом исследования**, приведенного в рамках данной курсовой работы, являются основы теории информации и криптографии.

**Предметом исследования** является сам предмет, методы, а также функции основ теории информации и криптографии.

**Цели работы и задачи исследования.** Цель работы – рассмотреть сущность предмета, методов и функций основ теории информации и криптографии. Поставленная цель определила следующие задачи исследования:

1. Произвести анализ сущности и понятия основ теории информации и криптографии;
2. Произвести анализ предмета, методов и функций основ теории информации и криптографии;
3. Рассмотреть этап формирования, развития и определения основ теории информации и криптографии;

Разработать программный проект, содержащий электронное пособие и программу для проверки знаний по вышеуказанной теме. СКОЛЬКО ЗАДАЧ ДОЛЖНО БЫТЬ НА СЕГОДНЯ???

**ЧТО ВООБЩЕ ПРО НИХ ГОВОРИЛИ**

**Методы исследования.** В ходе выполнения данной курсовой работы был использован теоретический метод исследования, позволяющий более явно продемонстрировать актуальность темы. Фундаментом теоретического метода исследования послужила работа Майстренко Н. В. и Майстренко А. В. в области теории информации и криптографии.

# **ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

* 1. **Основания для разработки**

Основанием для разработки является задание, выполненное в соответствии с заданием, полученным от кафедры «Информатика» Московского технического университета связи и информатики и утвержденное научным руководителем доц. Кафедры «Информатика» к.п.н. Гуриковым С.Р. Дата утверждения 02.10.2020.

* 1. **Назначение разработки**

Программный продукт предназначен для ознакомления с основными теоретическими положениями по теме курсовой работы и проверки знаний пользователя с помощью тестовой программы.

* 1. **Требования к программе**
     1. Требования к функциональным характеристикам

Разработанный программный продукт должен обеспечить выполнение следующих функций:

* Возможность вывода результатов исследования для пользователя
* Ознакомление с теоретической частью
  + 1. **Требования к надежности**

Разрабатываемое программное обеспечение должно иметь устойчивую работу, в соответствии с алгоритмом программы, выдавать сообщение об ошибках, поддерживать диалоговый режим, в рамках представляемых пользователю возможностей.

* + 1. **Требование к составу и параметрам технических средств**

Минимальные и рекомендуемые системные требования для ПК.

* + 1. **Требования к информационной и программной совместимости**

Программа должна легко устанавливаться, функционировать и корректно работать при наличии следующего ПО: OC Windows XP и более поздние версии.

* + 1. **Требования к транспортированию и хранению**

Программа поставляется на usb-флеш-накопителе. Программная документация поставляется в электронном и печатном виде.

* + 1. **Требования к программной документации**

В ходе разработки программы должны быть подготовлены: текст программы, описание программы, методика испытаний, руководство пользователя.

* 1. **Стадии и этапы разработки ЗДЕСЬ И ДАЛЕЕ НАДО ЧИТАТЬ ГОСТ В ЧАСТИ ОФОРМЛЕНИЯ ТАБЛИЦ**

Таблица 1 – стадии и этапы разработки

|  |  |  |  |
| --- | --- | --- | --- |
| Номер этапа | Название этапа | Срок | Отчетность |
| 1 | Утверждение темы | До 02.10.2020 |  |
| 2 | Написание введения | До 12.10.2020 |  |
| 3 | Составление ТЗ и написание теоретической части | До 29.10.2020 |  |

# **Глава 1. Теоретическая часть ЗДЕСЬ И ДАЛЕЕ НАДО ЧИТАТЬ ГОСТ В ЧАСТИ ОФОРМЛЕНИЯ ЗАГОЛОВКОВ**

* 1. **Основные понятия**

Информация – это совокупность сведений, подлежащих хранению, передаче, обработке и использованию в человеческой деятельности. Несмотря на то, что данный термин широко применяется в биологии, лингвистике, психологии и некоторых других науках, в различных областях знаний в него вкладывают разный смысл. Большой интерес к разнообразным информационным процессам повлёк за собой многочисленные толкования определений понятия «информация». То же касается и определений количества информации, которых существует довольно много. Наиболее адекватным постановке задачи считается шенноновское определение информации как меры неопределённости. **Как следствие**, основной целью передачи информации является снятие данной неопределённости.

Все подходы к определению количества информации условно делят на пять видов:

1) энтропийный;

2) алгоритмический;

3) комбинаторный;

4) семантический;

5) прагматический.

Энтропийный, алгоритмический и комбинаторный подходы могут дать количественное установление сложности объекта или явления, которое подвергалось наблюдению.

Семантический подход позволяет описать содержательную составляющую и новизну передаваемого сообщения именно для конкретного получателя сообщения.

Полезность уже полученного сообщения для конечного пользователя позволяет оценить прагматический подход к измерению информации.

Информационное взаимодействие можно представить величиной, состоящей из следующих компонентов:

1) физической;

2) сигнальной;

3) лингвистической;

4) семантической;

5) прагматической.

Если рассматривать процесс передачи информации на примере устной речи, то **можно, прежде всего, отметить** многокомпонентность данного процесса.

Физической компонентой в данном случае является необходимость ВЫРАВНИВАНИЕ??? наличия физической среды для распространения акустических колебаний (воздуха), источника акустического сигнала (голосовых связок человека) и приёмника колебаний (уха).

Роль второй, сигнальной, компоненты выполняют акустические колебания, амплитудно- и частотно-модулированные.

Третья, синтаксическая, компонента отображается в необходимости знания собеседниками хотя бы одного общего языка.

Семантическая компонента – в передаваемом сообщении необходимо присутствие содержательного описания объекта или явления, которое до этого было неизвестно получателю информации.

И, **наконец**, для эффективного процесса приёма/передачи сообщения необходимо желание и мотивация – прагматическая компонента.

Для того чтобы информацию можно было обрабатывать с использованием технических средств, её чаще всего необходимо представлять в неком формализованном виде. Информацию в таком виде принято называть данными.

Обычно выделяют этапы обращения информации:

1) восприятие информации;

2) подготовка информации;

3) передача и хранение информации;

4) обработка информации;

5) отображение информации;

6) воздействие информации.

* 1. **Система передачи информации**

Модель системы передачи (и хранения) информации приведена на рис. 1. Опишем данную модель.

1. В качестве источника информации или сообщения можно представить физический объект, систему или явление, которые формируют передаваемое сообщение. Первичные сообщения – это, **как правило**, изображения, музыка, речь, измерения параметров окружающей среды и т.д., представляющие собой функции времени неэлектрической природы. Преобразование первичных сообщений в электрический сигнал необходимо для их передачи по каналу связи. **ЗДЕСЬ И ДАЛЕЕ НАДО ЧИТАТЬ ГОСТ В ЧАСТИ ОФОРМЛЕНИЯ РИСУНКОВ**

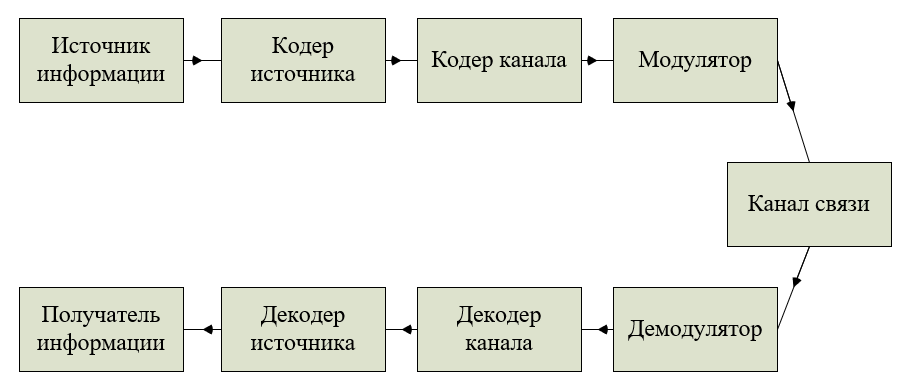


Рисунок 1 – Модель системы передачи информации

2. Кодер источника. Исходные сообщения в виде изображений, речи, музыки и т.п. предназначены прежде всего для восприятия органами чувств приёмника сообщения (человека). Такие сообщения (λ(t) или Λ) крайне неэффективно передавать в первозданном виде по каналам связи, поэтому они чаще всего кодируются. Кодирование в общем случае – это процесс преобразования алфавита сообщения в алфавит кодовых символов, которые выбираются некоторым образом. Кодировка источника будет означать уменьшение объёма информации для того, чтобы увеличить скорость её передачи или уменьшить частотную полосу, необходимую для передачи данных.

3. Кодер канала. В каналах связи часто бывают помехи, **вследствие чего** при передаче информации по такому каналу в принятых данных можно фиксировать ошибки. Для оптимизации количества ошибок, которые возникают при передаче сообщения по каналу с помехами, применяют кодирование в канале. Общий их принцип основан на следующем: в передаваемые сообщения заносится некоторая специально подобранная избыточность, которая позволяет при приёме сообщения (в декодере канала) обнаружить и исправить возникающие ошибки.

4. Модулятор. Основная функция модулятора заключается в согласовании сообщения источника или кодовых последовательностей, которые вырабатываются кодером со свойствами конкретного канала связи, **а также** предоставлении возможностей одновременной передачи по общему каналу связи большого числа сообщений.

5. Канал связи. В качестве каналов передачи могут использоваться воздух, проводные каналы, волоконно-оптические каналы, акустические каналы и др. **в зависимости** от вида конкретной системы передачи информации.

6. Приёмник. Основным назначением приёмника является воспроизведение с максимальной точностью по принятому колебанию на своём выходе переданного сообщения.

7. Демодулятор. Демодуляция обеспечивает оптимальный приём сигнала на фоне помех (оптимальное обнаружение, оптимальное различение двух или нескольких сигналов и т.д.).

8. Декодер канала. Задачей декодера является обнаружение и исправление найденных в сообщении ошибок – декодирование канала.

9. Декодер источника. Кодирование для более компактного или удобного представления информации источника приводит к необходимости восстановления информации к исходному виду после приёма. Декодирование источника – это процедура восстановления принятого сигнала.

* 1. **Постулаты теории информации**

Результаты решения целого ряда фундаментальных теоретических вопросов составляют проблемное поле теории информации, в частности:

– анализ сигналов с позиций средства передачи сообщений, который включает вопросы оценки переносимого этими сигналами «количества информации»;

– анализ инфохарактеристик каналов связи и источников сообщений, **а также** обоснование принципиальной возможности процессов кодирования и декодирования сообщений, которые обеспечивают максимально допустимую скорость передачи по каналу связи сообщений независимо от наличия помех.

Основные условия (постулаты) теории информации, при которых исследуются информационные системы, **можно сформулировать следующим образом**:

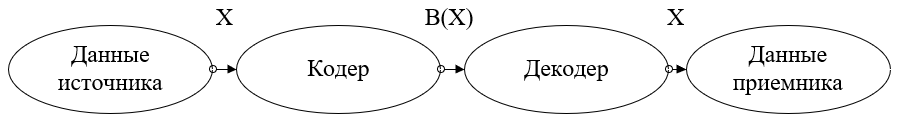
1. Источник сообщения с определённой вероятностью производит выборку сообщения из некоторого множества.
2. Сообщения по каналу связи могут передаваться в закодированном виде. При этом кодированные сообщения образуют множество, которое является взаимно однозначным отображением исходного множества сообщений. Все правила декодирования записаны в программе декодера.
3. Число сообщений неограниченно и может быть очень большим, при этом сообщения следуют друг за другом.
4. Сообщение принято верно только **в том случае, если** в результате декодирования это сообщение в точности восстановлено независимо от промежутка времени, прошедшего с момента передачи сообщения и до момента окончания декодирования, а также сложности операций кодирования и декодирования.
5. Общее количество информации не зависит ни в коей мере от смыслового содержания сообщения, его полезности и эмоционального воздействия и даже от его отношения к реальной действительности.
   1. **Основные принципы кодирования**

Под кодированием понимают преобразование алфавита исходного сообщения A{λi}, (i = 1, 2, …, K) в алфавит определённым способом подобранных кодовых символов R{xj}, (j = 1, 2, …, N). **Как правило** (но не всегда), размерность алфавита кодовых символов R{xj} значительно меньше размерности алфавита первоисточника A{λi}.

Цели у кодирования сообщений могут бать различные. Например, кодирование для засекречивания передаваемого сообщения. **В этом случае** элементарным сообщениям λi из исходного алфавита A{λi} ставятся в соответствие определённые последовательности цифр или букв из специальных кодовых таблиц, которые известны только получателю и отправителю информации.

Помехоустойчивое кодирование или кодирование в канале информации используется **в целях** снижения количества ошибок, возникающих в процессе передачи по каналу, имеющему помехи.

Ну и **наконец**, сообщения могут кодироваться для уменьшения объёма передаваемой информации и увеличения скорости передачи информации, а также сокращения полосы частот, необходимых для передачи информации. Подобное кодирование называется экономным кодированием или сжатием данных. В настоящем разделе речь будет идти как раз об этом виде кодирования. Выделяют сжатие с потерями информации и сжатие без потери информации.

В системах кодирования неразрушающего сжатия декодер регенерирует данные первоисточника абсолютно без потерь информации (рис. 2).

**Рисунок 2 - Схема сжатия без потерь: Х – последовательность длины n, состоящая из символов алфавита А; X(B) – сжатые данные, представленные в передаваемой последовательности длиной m, где m зависит от n.**

* 1. **Принципы кодирования**

**Рассмотрим** немного подробнее суть процедуры кодирования.

Любое дискретное сообщение λi из алфавита источника A{λi} объёмом в K символов может быть закодировано последовательностью определённым образом выбранных кодовых символов xj из алфавита R{xj}.

Так, любое число может быть следующим образом записано в заданной позиционной системе счисления:

λi = M = xn−1mn–1 + xn–2 mn-2 +… + x0m0, (6.1)

где m – основание системы счисления; x0, …, xn–1 – коэффициенты при степенях m; x ⊂ 0, m – 1.

Например, значение λi = M = 59. Код этого числа по основанию m = 8, будет следующим

M = 59 = 7·81 + 3·80 =738.

А код этого же числа, но по другому основанию, например m = 4, будет выглядеть таким образом:

M = 59 = 3⋅42 + 2⋅41+ 3⋅40 = 3234.

Если же основание кода принять равным m = 2, то

M = 59 = 1⋅25 + 1⋅24 + 1⋅23 + 0⋅22 + 1⋅21 + 1⋅20 = 1110112.

Как видим, числа 73, 323 и 111011 могут выступать, соответственно, как восьмеричный, четверичный и двоичный код числа M = 59.

**Таким образом**, основание кода может быть любым, но наибольшее распространение получили коды с основанием 2. Двоичные коды достаточно легко формируются и далее передаются по любым каналам связи. **А также** бесспорным преимуществом двоичных кодов является тот факт, что они служат внутренним языком всех цифровых ЭВМ, т.е. без каких-либо дополнительных преобразований могут обрабатываться цифровыми средствами электроники.

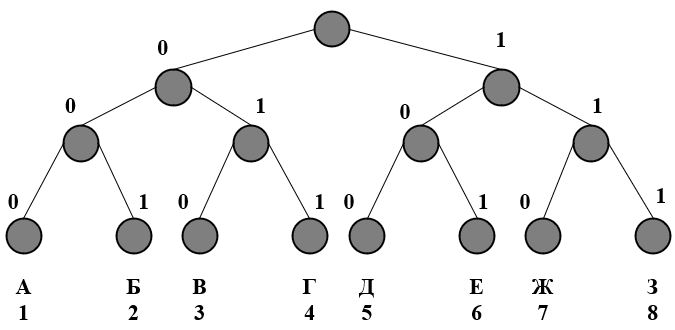
Наиболее простым способом задания кодов и их представления являются кодовые таблицы, которые ставят в соответствие сообщениям необходимые коды (табл. 2).

Таблица 2 – коды символов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Буква | Число | Код с основанием 10 | Код с основанием 4 | Код с основанием 2 |
| А | 0 | 0 | 00 | 000 |
| Б | 1 | 1 | 01 | 001 |
| В | 2 | 2 | 02 | 010 |
| Г | 3 | 3 | 03 | 011 |
| Д | 4 | 4 | 10 | 100 |
| Е | 5 | 5 | 11 | 101 |
| Ж | 6 | 6 | 12 | 110 |
| З | 7 | 7 | 13 | 111 |

Другой удобный и наглядный способ описания кодов – их представление в виде кодового дерева.

Для построения кодового дерева для выбранного кода, начиная с определённой точки – корня кодового дерева – строятся ветви – 0 или 1. В вершинах кодового дерева располагаются буквы алфавита источника. **При этом** каждой букве алфавита источника соответствуют своя вершина и свой путь от корня к вершине. Например, букве А соответствует код 000, букве В – 010, букве Е – 101 и т.д. На рисунке 3 изображён пример равномерного трёхразрядного кода.

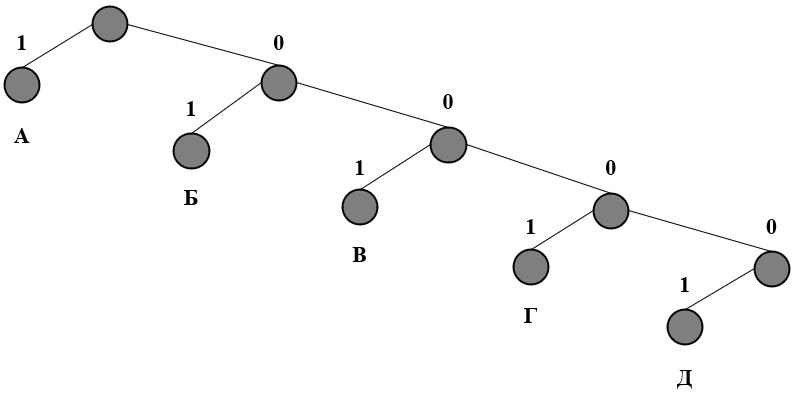


**Рисунок 3 - Дерево равномерного кода**

Одновременно с равномерными кодами применяются и неравномерные коды, у которых каждая буква из алфавита источника кодируется различным числом символов, например, А – 10, Б – 110, В – 1110 и т.д.

При использовании неравномерного кода букве А будет соответствовать код 1, Б – 0, В – 11 и т.д. **Однако** нетрудно заметить, что, закодировав произвольный текст АББА = 1001 подобным образом, мы не можем его однозначно декодировать, так как такой же код имеют фразы: ЖА = 1001, АЕА = = 1001 и ГД = 1001. Подобные коды, не позволяющие однозначно декодировать сообщение, называются непрефиксными или приводимыми кодами. Они не применяются на практике без специальных разделительных символов. **Примером** такого типа кодов **является** азбука Морзе, в которой кроме тире и точек есть специальные символы, разделяющие слова и буквы. Но это уже не двоичный код.

**Тем не менее** возможно составить неравномерные неприводимые коды, дающие возможность однозначного декодирования. Для этого необходимо поставить в соответствие всем буквам алфавита вершины кодового дерева (рис. 4). На приведённом рисунке неоднозначности декодирования не существует, так как ни одна кодовая комбинация не является началом другой, более длинной кодовой комбинации. Подобные неравномерные коды носят название префиксных.

****

**Рисунок 4 - Дерево неравномерного кода**

**Таким образом**, эффективность кодирования кодирующего устройства определяется следующими его свойствами:

1. Обеспечение безошибочной передачи информации, т.е. однозначное взаимное соответствие между Х и Y.

2. Обеспечение кодирования с минимальной избыточностью наиболее экономным образом.

Для выполнения первого требования необходимо:

а) чтобы различным буквам алфавита ставились в соответствие различные кодовые слова;

б) чтобы предусматривалась возможность разделения кодовых слов в случае их последовательной передачи. Для этого:

* вводят и применяют специальные разделяющие символы;
* используют кодовые слова равной длины;
* кодовая таблица составляется так, чтобы никакое кодовое слово не служило началом другого кодового слова.

Для выполнения второго требования нужно стремиться при кодировании к минимизации средней длины кодового слова.

* 1. **Алгоритмы сжатия без потерь**

**Выделяют** две категории методов сжатия без потерь:

1. Методы сжатия источников данных без памяти (т.е. не учитывающие последовательность символов).

2. Методы сжатия источников данных с памятью.

Рассмотрим сжатие способом кодирования серий (RLE). Кодирование серий последовательностей (Run Length Encoding – RLE) – это наиболее известный и простой подход сжатия информации обратимым путём. **Суть** методов этого подхода заключается в замене серий или цепочек повторяющихся байтов или последовательностей байтов на один кодирующий байт и счётчик числа повторений исходных байтов. RLE – первый вариант (рис. 5).

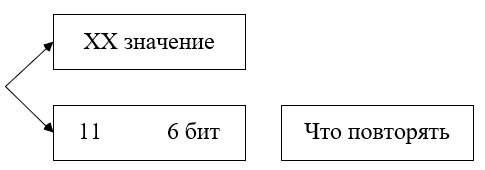


Рисунок 5 ???????????????

В описываемом алгоритме признаком счётчика являются единицы в двух верхних битах считанного сообщения, а оставшиеся 6 бит используются для записи показаний счётчика, который может принимать значения от 1 до 64. **Таким образом**, строка из 64 повторяющихся байтов превращается в два байта, т.е. сжимается в 32 раза.

Алгоритм используется для работы с деловой графикой – изображениями со значительными повторяющимися цветовыми областями. **Однако** не так уж редка ситуация, когда при использовании этого простого алгоритма файл увеличивается. Это может произойти в случае применения группового кодирования к уже обработанным цветным фотографиям.

Другой вариант RLE-алгоритма (рис. 6) имеет большую по сравнению с первым максимальную степень сжатия и в меньшей степени увеличивает исходный файл в размерах.

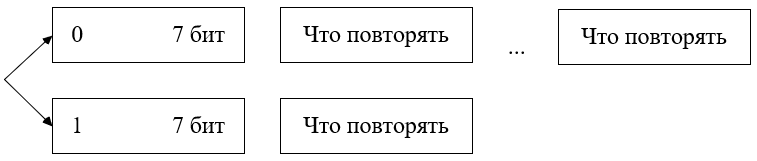


Рисунок 6

**Признаком** повтора в этом варианте алгоритма **служит** единица в старшем разряде соответствующего байта.

**Проблема** всех подобных методов состоит в определении способа, по которому распаковывающий алгоритм отличает в результирующем потоке байтов кодированную серию от других – некодированных последовательностей байтов. Эта проблема решается, как правило, простановкой специальных меток в начале кодированных цепочек. Эти методы являются достаточно эффективными для сжатия растровых графических изображений (TIF, PCX, BMP, GIF), **так как** они содержат достаточно много длинных цепочек повторяющихся последовательностей байтов. Недостатком метода RLE является довольно низкая степень сжатия или стоимость кодирования файлов с малым числом серий и малым числом повторяющихся байтов в сериях.

Описанный алгоритм и его модификации применяются в основном для работы в реальном масштабе времени и в потоке.

* 1. **Методы сжатия источников без памяти**

Самым распространенным алгоритмом сжатия без памяти является алгоритм Хаффмана. Основная идея алгоритма заключается в том, что, зная вероятность вхождения символов в сообщение, можно описать процедуру построения кодов переменной длины, состоящих из целого количества битов. Символам с большей вероятностью присваиваются более короткие коды. Коды Хаффмана обладают свойством префикса, что и позволяет однозначно их декодировать, несмотря на их переменную длину. Алгоритм Хаффмана состоит из **следующих шагов**:

1. Символы входного алфавита составляют список свободных узлов. Каждый лист имеет вес, который может быть равен либо вероятности, либо количеству вхождений символа в ожидаемое сообщение.
2. Выбираются два свободных узла дерева с наименьшими весами.
3. Создается родитель с весом, равным их суммарному весу.
4. Родитель добавляется в список свободных узлов, а двое его детей удаляются из этого списка.
5. Одной дуге, выходящей из родителя, ставится в соответствие бит 1, другой – бит 0.
6. Далее пункты повторяются, начиная со второго, до тех пор, пока в списке свободный узлов не останется только один свободный узел. Он и будет считаться корнем дерева.

**Пусть** мы имеем таблицу частот

15 7 6 6 5

A B C D E

На первом шаге выбираем из листьев дерева два листа с наименьшими весами – D и E. Присоединяем их к новому узлу-родителю, вес которого устанавливается равным 11 = 5 + 6. Затем узлы D и E удаляются из списка свободных. Узел D соответствует ветви 0 родителя, узел E – ветви 1.

**На следующем этапе** то же самое происходит с узлами B и C, так как именно они теперь имеют самый меньший вес в дереве. Создается новый узел с весом 13, а узлы B и C удаляются из списка свободных. После всего этого дерево кодирования принимает вид, представленный на рис. 7.

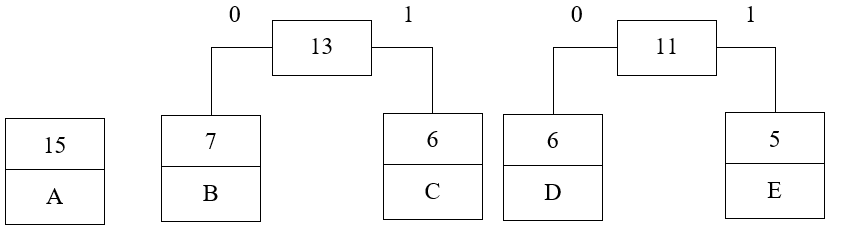


Рисунок 7

**Далее** «наилегчайшей» парой оказываются узлы B/C и D/E. Для этих узлов еще раз создается родитель, но теперь уже с весом 24. Узел B/C соответствует ветви 0 родителя, D/E соответствует ветви 1. **На последнем шаге** работы алгоритма в списке свободных остаются два узла – узел A и узел (B/C)/(D/E). В очередной раз создается родитель весом 39 и бывшие свободные узлы присоединяются к разным его ветвям.

Так как свободным остается только один узел, то алгоритм построения дерева кодирования Хаффмана на этом завершается. Н-дерево представлено на рис. 8.

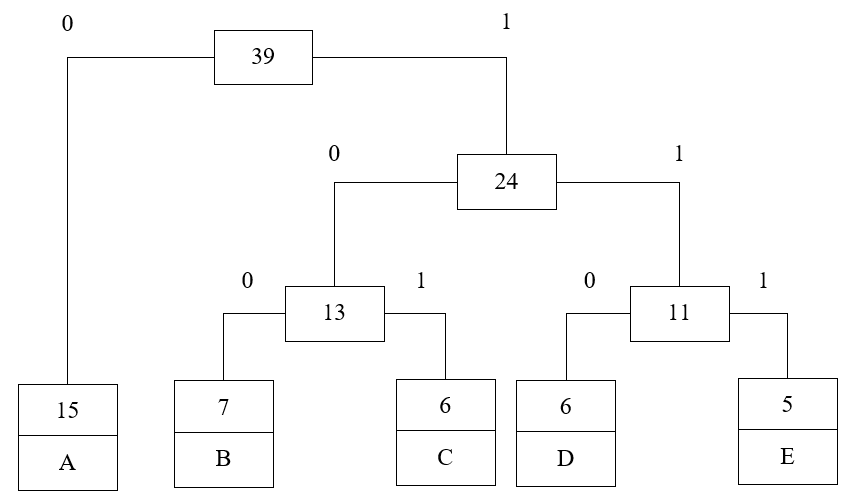


Рисунок 8

Для определения кода для каждого из символов, входящих в сообщение, необходимо пройти путь от листа до дерева, соответствующего этому символу, до корня дерева, накапливая биты при перемещении по ветвям дерева. **Полученная подобным образом** последовательность битов будет являться кодом данного символа, которая записана в обратном порядке. Коды Хаффмана для приведенной выше таблицы символов будут выглядеть следующим образом:

A 0

B 100

C 101

D 110

E 111

Так как ни один из полученных кодов не является префиксом другого, они однозначно декодируются при чтении их из потока. **Более того**, наиболее частый символ сообщения A закодирован наименьшим количеством битов, а наиболее редкий символ E – наибольшим.

Недостатком приведенного способа кодирования является то, что вместе с закодированным сообщением требуется передача построенной таблицы кодов (дерева), что понижает величину сжатия.

* 1. **Методы сжатия источников c памятью**

Входную последовательность символов в сообщении можно рассматривать в виде последовательности строк, которые содержат произвольное количество символов. **Основная идея** словарных методов **заключается** в замене строк символов на коды, которые можно трактовать как индексы строк некоторого словаря.

**Таким образом**, происходит попытка преобразовать исходную последовательность символов сообщения так, что его «буквы» являются фразами словаря, состоящими в общем случае из произвольного количества символов входной последовательности.

Словарь – это набор отдельных фраз, которые, **предположительно**, будут появляться в обрабатываемой последовательности. Индексы фраз строятся таким образом, чтобы их представление в среднем занимало меньше места, чем требуется для замещаемых строк. Это и позволяет осуществлять сжатие.

Рассмотрим классические алгоритмы Зива–Лемпела. Алгоритмы словарного сжатия Зива–Лемпела были разработаны во второй половине 1970х годов. К ним относятся алгоритмы LZ77 и LZ78, созданные совместно Зивом (Ziv) и Лемпелом (Lempel) а также их модификации.

LZ77 и LZ78 являются универсальными алгоритмами сжатия, в которых словарь формируется адаптивно на основании уже обработанной некоторой части входного потока. **Принципиальным отличием** алгоритмов друг от друга является лишь способ формирования фраз.

LZ77 – является «родоначальником» целого семейства словарных схем – так называемых алгоритмов со скользящим словарем или скользящим окном. В LZ77 в качестве словаря используется блок уже закодированной последовательности. **Как правило**, по мере выполнения обработки положение этого блока относительно начала последовательности постоянно меняется, словарь «скользит» по входному потоку данных.

LZW. Название алгоритм получил по первым буквам фамилий его разработчиков — Lempel, Ziv и Welch и является модификацией алгоритма LZ78. За счет предварительного занесения в словарь всех символов алфавита входной последовательности результат работы LZW состоит только из последовательности индексов фраз словаря. Из-за устранения необходимости регулярной передачи одного символа в явном виде LZW обеспечивает лучшее сжатие, чем LZ78.

Таблица словаря для LZW состоит из 4096 строк (рис. 8):

коды 256 и 257 являются служебными.

258 … 4095 содержат непосредственно сжимаемую информацию.

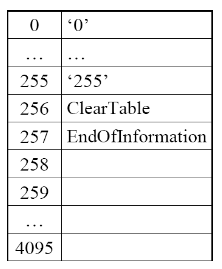


Рисунок 9

* 1. **Основные термины и понятия криптологии**

Наука криптология, занимающаяся проблемами защиты информации за счёт её преобразования, состоит из двух разделов: криптография и криптоанализ.

Криптография как область практической и научной деятельности до 70х годов прошлого века была связана с разработкой и применением различного рода шифросистем. Сегодня эта область человеческой деятельности связана с разработкой, анализом и применением систем криптографической защиты информации.

**Основными функциями** криптографических систем **являются** обеспечение аутентичности и конфиденциальности различных сторон информационного взаимодействия.

В качестве источников угроз при постановке и решении различных криптографических задач выступают преднамеренные действия недобросовестного участника или противника информационного взаимодействия, но не случайное изменение информации, вызванное помехами, отказами и т.п.

Конфиденциальность заключается в защищённости информации от возможности ознакомления с её содержанием лицами, не имеющими права доступа к ней.

Криптоанализ – наука о способах и методах анализа криптографических отображений информации в целях её раскрытия. **Основные составляющие части** криптографии показаны на рис. 9.

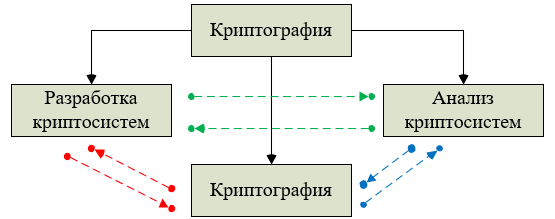


Рисунок 10 – Схема взаимодействия составляющих криптографии

Криптосистема – система, предназначенная для обеспечения безопасности защищенной сети с использованием криптографических средств. Подсистемой криптосистемы **является** ключевая система, обеспечивающая функционирование других подсистем: шифрования, имитозащиты, цифровой подписи, идентификации и др.

Рассмотрим пример. **Пусть** некий отправитель желает отправить письмо получателю, но так, чтобы при перехвате это сообщение никто не смог бы прочесть. Это передаваемое сообщение носит название открытого текста, преобразование этого сообщения – шифрования, обратный процесс преобразования сообщения в открытый текст – дешифровки, а зашифрованное сообщение – шифротекста.

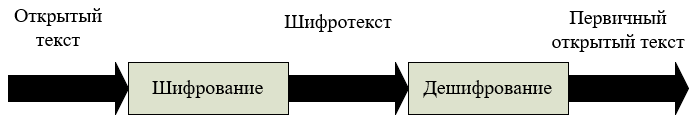


Рисунок 11 - Процесс шифрования/дешифрования

Введём обозначения:

* + M (message) – открытый текст (например, текстовый файл, поток битов и т.п.);
  + C (ciphertext) – шифротекст;
  + Е – функция шифрования, E(M) = C;
  + D – функция дешифрования, D(C) = M;

**При этом необходимо** выполнение: D(E(M)) = M.

Криптографический алгоритм есть некоторые математические функции, которые используются для шифрования/дешифрования.

Современные средства криптографии для шифрования используют ключ k. Этот ключ может принимать любое значение, выбираемое из большого множества K. Всё множество возможных ключей называется пространством ключей.

Если в процессе шифрования и дешифрования применяется один и тот же ключ, то имеют дело с симметричным шифрованием. **Конечно же**, этот ключ обязан храниться в секрете, отсюда другое название подобных систем – системы с секретным ключом.

В случае несимметричного шифрования (иначе алгоритм с открытым ключом) различают два типа ключей: k1 – открытый ключ, который доступен всем участникам информационного обмена, и k2 – закрытый ключ, доступный только отправителю сообщения, определить ключ k2 дешифрования по k1 практически невозможно.

Безопасность алгоритмов подобных типов полностью основана на ключах, т.е. эти алгоритмы шифрования возможно опубликовать и проанализировать.

**Достоинством** симметричных алгоритмов **являются** их высокая скорость работы, небольшие размеры этих ключей и более высокие гарантии обеспечения криптостойкости.

**В отличие** от симметричных, в несимметричных криптосистемах применяются более удобные протоколы.

Криптографический протокол – это заданная последовательность действий, которые выполняются для решения криптографической задачи. Отличием алгоритма от протокола является то, что для реализации протокола требуется взаимодействие нескольких сторон протокола. К одной из наиболее важных криптографических задач относится задача распределения ключей.

* 1. **Основные понятия криптоанализа**

Попытка криптоанализа носит название вскрытия. В XIX веке Д. А. Керкхоффс впервые сформулировал основное предположение криптоанализа, **заключающееся в том**, что безопасность в полной мере определяется ключом и, что криптоаналитик имеет полное описание алгоритма шифрования, а также его реализацию.

Безусловно безопасным называется такой алгоритм шифрования, в котором независимо от объёма шифротекстов, имеющихся у криптоаналитика, недостаточно данных для получения открытого текста.

* 1. **Шенноновские модели криптографии**

**Рассмотрим** математические модели, характерные для элементарных криптосистем:

1. Подстановка: пусть имеется открытое сообщение [x1, …, xn] в алфавите А и система отображений Φ={ φ1, …, φn} символов из алфавита А в алфавит шифрованного текста В. Тогда зашифрованный текст предстанет в виде Y = [φ1 (x1), …, φn (xn)].

2. Перестановка: пусть имеется открытое сообщение [x1, …, xn]; перестановка имеет вид Ψ = {ψ1, …, ψn}. Тогда зашифрованный текст будет выглядеть следующим образом: Y = {xψ1, …, xψn}.

Недостатками представленной модели являются:

* часто встречаемые символы в X и Y будут совпадать;
* ограничения по размеру ключа, которые приводят к его многократному применению.

Для усиления криптостойкости этого криптопреобразования **целесообразно использовать** композицию из нескольких перестановок, отличающихся длинами блоков.

3. Шифр Виженера и его модификация: **пусть** имеется некоторое открытое сообщение [x1, …, xn]; тогда шифрованное сообщение будет получаться путём преобразования: yt = (xt + kτ)mod |X|, в котором t = (i-1)T+ τ, *i* = 1, 2, …; τ = , где T – длина блока. Ключ k есть фиксированный набор символов, принадлежащих алфавиту открытого сообщения.

Тогда обратное преобразование будет иметь вид

xt = (yt - kτ + |X|) mod|X|.

Частными случаями будут являться: шифры Цезаря, Бофора.

4. Криптопреобразование Вернама (или поточный шифр): является частным случаем шифра Виженера, в котором T = n. Ключ k здесь будет называться гаммой, одноразовым блокнотом или бегущей строкой.

5. Биграммная подстановка: данная подстановка использует принцип простой подстановки, **однако** вместо подстановки типа «один символ – один символ», происходит одновременная подстановка типа «m символов – m символов».

Шеннон К. теоретически доказал существование криптосистемы, которая обладает совершенной безопасностью (т.е. шифр, кроме длины, не даёт никакой другой информации об открытом сообщении). Такое возможно в случае, если число возможных ключей будет таким же большим, что и число возможных сообщений (другими словами, ключ не должен дважды использоваться и быть короче открытого сообщения).

Основными показателями криптостойкости являются:

1) количество всех возможных ключей;

2) среднее время, затрачиваемое на криптоанализ;

3) количество и качество (или достоверность) всех криптограмм, которые были перехвачены;

4) шифросистема обладает высокой криптографической стойкостью, когда выражается экспоненциально через длину ключа;

5) количество материала, необходимого для анализа при вскрытии шифра (к примеру, полный перебор ключей криптосистемы);

6) стоимость процесса дешифрирования системы (например, разработка новой вычислительной системы).

Для всех современных криптографических систем защиты информации можно сформулировать следующие общие требования:

* только имея ключ, зашифрованное сообщение может быть прочитано;
* число операций, требуемых для расшифровывания информации, используя перебор всех возможных ключей, должно характеризоваться строгой нижней оценкой и превосходить возможности современных компьютерных систем (даже с учётом всех возможностей распараллеливания и сетевых вычислений);
* количество операций, требуемых для определения уже использованного ключа шифрования по некоторому фрагменту шифрованного сообщения и соответствующего ему открытого текста, должно быть не менее общего количества возможных ключей;
* несущественное изменение ключа должно в результате приводить к значительному изменению зашифрованного сообщения;
* знание алгоритма шифрования не может влиять на надёжность защиты;
* структурные элементы, составляющие алгоритм шифрования, не должны меняться;
* длина шифрованного текста и длина исходного текста должны быть равны;
* биты, дополнительно вводимые в сообщение в ходе шифрования, должны быть надёжно и полностью скрыты в зашифрованном тексте;
* не должно быть легко и просто устанавливаемых зависимостей между ключами, которые последовательно используются в процессе шифрования;
* алгоритм шифрования должен допускать и программную, и аппаратную реализацию;
* изменение длины ключа не должно приводить к качественному ухудшению выбранного алгоритма шифрования;
* любой ключ из всего множества возможных ключей должен обеспечивать надёжную защиту всей информации.
  1. **Система шифрования RSA**

Система шифрования RSA является шифрованием с открытым ключом. **В основе** алгоритма RSA **лежит** предположение, что поиск больших простых чисел вычислительно не представляет сложности, однако практически невозможно разложить большое простое число на произведение двух таких чисел.

В основесистемы лежит следующая степенная функция **ЗДЕСЬ НАДО ЧИТАТЬ ГОСТ В ЧАСТИ ОФОРМЛЕНИЯ ФОРМУЛ**

ƒ(x) = x*e* mod *n*.

Здесь блок х открытого текста является целым числом из диапазона [0, n-1] и преобразуется с помощью вычисления: y = xe mod n в блок шифротекста из этого же диапазона. Пара (e, n), носит название открытого ключа и применяется при шифровании сообщения. Число n = pq, где q, p – различные простые случайные числа. Эти числа генерируются одной из сторон, участвующих в информационном обмене.

**В случае** расшифровывания применяется функция (x) = xd mod n: x = yd mod n. Здесь пара (d, n) носит название закрытого ключа и используется при дешифровании.

Согласованность этих преобразований **обосновывает** теорема Эйлера. (пусть 1> n, НОД (z, n) = 1; тогда: zϕ(n) ≡ 1(mod n)).

Чтобы воспользоваться теоремой Эйлера, в RSA используют такие числа e, d, для которых: ed = 1(mod ϕ(n)), ϕ(n) = (p-1) (q-1). Тогда если е известно, то d – есть решение сравнения первой степени. Или **иначе**, проверив НОД (e, ϕ(n)) = 1, с использованием расширенного алгоритма Евклида можно найти d, t: ed + tϕ(n). Проверим работоспособность формулы дешифрования:

x = yd mod n = (xe mod n)d mod n = [(xe mod n)\*…\*(xe mod n)] mod n =

= xed mod n = x1-tϕ(n) mod n = x\*x-tϕ(n) mod n =

= [x mod n\*(x ϕ(n) mod n)-t] mod n = [x mod n\*1-t] mod n = x.

**В основе** криптостойкости системы шифрования RSA **лежит** то, что, имея открытый ключ (e, n), криптоаналитик может найти d:

1. зная функции Эйлера ϕ(n). Эта задача – есть задача разложения (или факторизации) n на простые множители. **Однако** нахождение функции Эйлера по определению задача вычислительно труднореализуемая. Даже используя самый эффективный алгоритм факторизации (например, решето числового поля), число n длиной в 50 десятичных разрядов может быть разложено приблизительно за 1010 операций (суперкомпьютеры), а если 800 десятичных разрядов – примерно за 1051 операций (не раскладывается вообще в настоящее время);
2. либо найти (задача дискретного логарифма).

Пример.

Для упрощения расчётов используем небольшие числа.

Будем использовать таблицу ASCII кодов минус 64 для перевода сообщений в числа.

**Предположим**, что абонент А сгенерировал два простых числа: p = 3, q = 11 → n = 33, ϕ(n) = 20. Находим e: НОД (e, n) = 1, 1 < e <ϕ(n), например e=7.

Пара (7, 33) публикуется как открытая.

Находим далее d:7d ≡ 1(mod ϕ(n)) → d = 3.

Пара (3, 33) при этом держится абонентом А в тайне.

Пусть теперь абонент B хочет послать секретное сообщение: «CAB». В нём A→1, B→2, C→3.

Зашифруем: C1 = 37 mod 33 = 9, C2 = 17 = mod 33 = 1, C3 = 27 mod 33 = 29.

Попробуем теперь дешифровать: C1′ = 97 mod 33 = 15 (не получается!).

Абоненту A посылают: «IA».

Абонент A переводит сообщение в числовое представление и дешифрует его:

M1 = 93 mod 33 = 3, M2 = 13 mod 33 = 1, M3 = 293 mod 33 = 2. Получается сообщение: «CAB».

**Вывод**

Итак, в 1 главе курсовой работы мы произвели анализ сущности и понятия основ теории информации и криптографии, произвели анализ предмета, методов и функций основ теории информации и криптографии и рассмотрели этап формирования, развития и определения основ теории информации и криптографии;

**Cписок использованных источников**

1. Майстренко, Н. В. Основы теории информации и криптографии: учебное пособие / Н. В. Майстренко, А. В. Майстренко. — Тамбов: Тамбовский государственный технический университет, ЭБС АСВ, 2018. — 81 c
2. ГОСТ 7.32-2017 СИБИД. Отчет о НИР – Взамен ГОСТ 7.32-2001; введ. 01.07.2018 – Москва: Изд-во стандартов, 2017. – 24 с.
3. ГОСТ 19.701-90 ЕСПД – Взамен ГОСТ 19.002-80, ГОСТ 19.003-80; введ. 01.01.92 – Москва: Стандартинформ, переизд. 2010. – 24 с
4. ГОСТ 19.201-78 ЕСПД; введ. 01.01.80 – Москва: Стандартинформ, 2010 - 4 с