

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по выполнению лабораторных работ
по дисциплине
«Основы информационной безопасности информационно-коммуникационных систем»
для студентов направления подготовки
11.03.02 «Инфокоммуникационные технологии и системы связи»
направленность (профиль) «Интеллектуальная обработка данных в инфокоммуникационных системах и сетях»

Ставрополь,

2026 г.

Содержание

Введение.....	4
Теоретические сведения.....	5
Лабораторная работа № 1. Использование классических криптоалгоритмов подстановки и перестановки для защиты текстовой информации	21
Лабораторная работа № 2. Исследование различных методов защиты текстовой информации и их стойкости на основе подбора ключей	32
Лабораторная работа № 3. Стандарт симметричного шифрования <i>Aes Rijndael</i>	45
Лабораторная работа № 4. Изучение устройства и принципа работы шифровальной машины Энигма.....	56
Лабораторная работа № 5. Генерация простых чисел, используемых в асимметричных системах шифрования	70
Лабораторная работа № 6. Электронная цифровая подпись.....	84
Лабораторная работа № 7. Шифрование методом скользящей перестановки	95
Лабораторная работа № 8. Изучение программных продуктов ЗИ. Программа PGP	114
Лабораторная работа № 9. Корректирующие коды. Коды Хемминга.....	128
Лабораторная работа № 10. Корректирующие коды. Циклические коды...	153
Лабораторная работа № 11. Методы сжатия по Шеннону и Хаффмену.....	158

Введение

В настоящее время во всем мире резко повысилось внимание к проблеме информационной безопасности. Это обусловлено процессами стремительного расширения потоков информации, пронизывающих все сферы жизни общества.

Информация давно перестала быть просто необходимым для производства вспомогательным ресурсом или побочным проявлением всякого рода деятельности. Она приобрела ощутимый стоимостной вес, который четко определяется реальной прибылью, получаемой при ее использовании, или размерами ущерба, с разной степенью вероятности наносимого владельцу информации. Однако создание индустрии переработки информации порождает целый ряд сложных проблем. Одной из таких проблем является надежное обеспечение сохранности и установленного статуса информации, циркулирующей и обрабатываемой в инфокоммуникационных системах и сетях.

В данных методических указаниях разработаны лабораторные работы, направленные на исследование принципов построения и параметров основных узлов, обеспечивающих защиту информации.

1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

1.1. Проблемы защиты информации

В настоящее время во всем мире резко повысилось внимание к проблеме информационной безопасности. Это обусловлено процессами стремительного расширения потоков информации, пронизывающих все сферы жизни общества.

Информация давно перестала быть просто необходимым для производства вспомогательным ресурсом или побочным проявлением всякого рода деятельности. Она приобрела ощутимый стоимостной вес, который четко определяется реальной прибылью, получаемой при ее использовании, или размерами ущерба, с разной степенью вероятности наносимого владельцу информации. Однако создание индустрии переработки информации порождает целый ряд сложных проблем. Одной из таких проблем является надежное обеспечение сохранности и установленного статуса информации, циркулирующей и обрабатываемой в информационно-вычислительных системах и сетях.

Появление глобальных компьютерных сетей сделало простым получение доступа к информации, как для отдельных пользователей, так и для больших организаций. Но легкость и высокая скорость доступа к данным при помощи компьютерных сетей, таких как *Internet*, также сделали значительными следующие угрозы безопасности данных при отсутствии мер их защиты:

- неавторизованный доступ к информации;
- неавторизованное изменение информации
- неавторизованный доступ к сетям и сервисам;
- другие сетевые атаки, например, повтор перехваченных ранее транзакций и атаки типа "отказ в обслуживании".

При обработке любой значимой информации при помощи отдельного компьютера, а тем более в сети, возникает вопрос о ее защите от

несанкционированного доступа и использования. Наиболее распространенный в компьютерных системах способ защиты – использование паролей – более пригоден для защиты доступа к вычислительным ресурсам, нежели для защиты информации. Это своеобразный экран, отгораживающий законных пользователей системы от посторонних, пройдя сквозь который, квалифицированный пользователь получает доступ практически ко всей информации.

В настоящее время исключительно важное значение в разных областях приобрели вопросы, связанные с сохранением и передачей конфиденциальной информации. Возникающие при этом задачи решает *криптография* – наука о методах преобразования информации в целях ее защиты от незаконных пользователей.

Ретроспективный взгляд на историю развития криптографии, как специфическую область человеческой деятельности, позволяет выделить три основных периода. Первый, наиболее продолжительный, – это эпоха "ручной криптографии". Ее начало теряется в глубокой древности, а окончание приходится на 30-е годы XX века. Криптография прошла путь от магического искусства до вполне прозаической прикладной специальности чиновников дипломатических и военных ведомств.

Второй период – создание и широкое внедрение в практику сначала механических, затем электромеханических и электронных устройств шифрования, организация целых сетей засекреченной связи. Его началом можно считать разработку Гилбертом Вернамом (*G.Vernam*) в 1917 году схемы телеграфной шифровальной машин, использующей *одноразовую гамму*, рис.1.1.

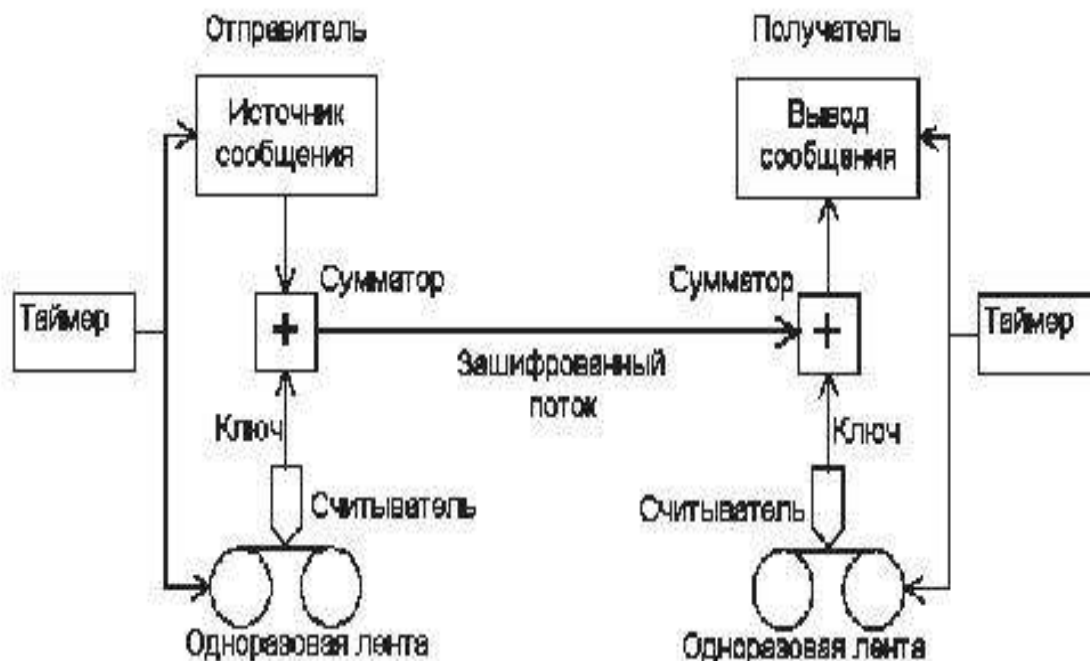


Рис. 1.1. Шифрование методом Вернама

К середине 70-х годов с развитием разветвленных коммерческих сетей связи, электронной почты и глобальных информационных систем на первый план вышли новые проблемы – проблемы снабжения ключами и проблемы подтверждения подлинности.

В 1976 году американские ученые Уитфилд Диффи (*W.Diffie*) и Мартин Хеллман (*M.Hellman*) предложили два новых принципа организации засекреченной связи без предварительного снабжения абонентов секретными ключами шифрования – принцип так называемого *открытого шифрования* и принцип *открытого распределения ключей*. Этот момент можно считать началом нового периода в развитии криптографии. В настоящее время это направление современной криптографии очень интенсивно развивается.

1.2. Из истории криптографии

Понятие “безопасность” охватывает широкий круг интересов, как отдельных лиц, так и целых государств. Во все исторические времена

существенное внимание уделялось проблеме информационной безопасности, обеспечению защиты конфиденциальной информации от озна-

комления, кражи, модификации, подмены. Решением этих вопросов занимается *криптография*.



Рис. 1.2. Джон Валлис

Криптография – тайнопись. Термин ввел Джон Валлис (*John Wallis*) (1616-1703), английский математик. Потребность шифровать и передавать шифрованные сообщения возникла очень давно. Так еще в V-VI вв до н.э. греки применяли специальное шифрующее устройство. По описанию Плутарха, оно состояло из двух цилиндрических стержней одинаковой длины и толщины. Один оставляли себе, а другой отдавали отъезжающему. Эти стержни называли *считалами*.

При необходимости передачи сообщения, длинную ленту папируса наматывали на считалу, не оставляя на ней никакого промежутка. Затем, оставляя папирус на считале, писали на нем все, что необходимо, а, написав, снимали папирус и без стержня отправляли адресату. Так как буквы оказывались разбросанными в беспорядке, то прочитать сообщение мог только тот, кто имел свою считалу такой же длины и толщины, намотав на нее папирус.

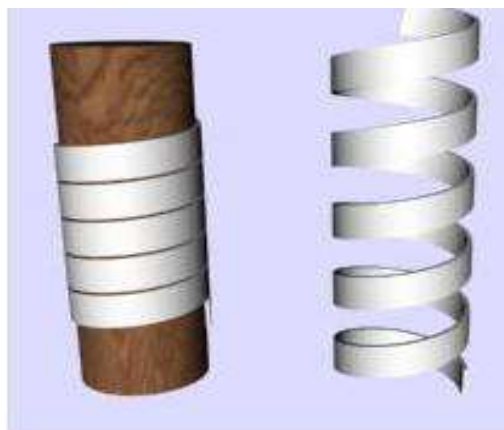


Рис 1.3. Считала

Квадрат Полибия

В Древней Греции (II в. до н.э.) был известен шифр, называемый “квадрат Полибия”.

Это устройство представляло собой квадрат 5*5, столбцы и строки которого нумеровали цифрами от 1 до 5. В каждую клетку записывалась одна буква (в греческом варианте одна клетка оказывалась пустой, а в латинском – в одну клетку помещали две буквы *I, J*). В результате каждой букве отвечала пара чисел по номеру строки и столбца.

1	2	3	4	5	
A	B	C	D	E	1
F	G	H	I, J	K	2
L	M	N	O	P	3
Q	R	S	T	U	4
V	W	X	Y	Z	5

13	34	22	24	44	34	15	42	22	34	43	45	32
----	----	----	----	----	----	----	----	----	----	----	----	----

Cogito ergo sum - лат. “Я мыслю, следовательно, существую”
Р.Декарт

Код Цезаря

В I в.н.э. Ю.Цезарь во время войны с галлами, переписываясь со своими друзьями в Риме, заменял в сообщении первую букву латинского алфавита (*A*) на четвертую (*D*), вторую (*B*) – на пятую (*E*), наконец последнюю – на третью.

¹ Полибий (200-120 гг. до н.э.) древнегреческий историк



ABCDEFGHIJKLMNOPQRSTUVWXYZ
 DEFGHIJKLMNOPQRSTUVWXYZABC

YHQL YLGL YLFL
 Veni vidi vici - "Пришел, увидел
 победил"
 Ю.Цезарь Донесение Сенату о победе
 над понтийским царем

□

Шифр Цезаря относится к так называемому классу *моноалфавитных подстановок* и имеет множество модификаций.

Решетка Кардано

Широко известны шифры, принадлежащие к классу *перестановка*, в частности "*решетка Кардано*"². Это прямоугольная карточка с отверстиями, чаще всего квадратная, которая при наложении на лист бумаги оставляет открытыми лишь некоторые его части. Число строк и столбцов на карточке четно. Карточка сделана так, что при последовательном ее поворачивании каждая клетка лежащего под ней листа окажется занятой. Карточку поворачивают сначала вдоль вертикальной оси симметрии на 180°, а затем вдоль горизонтально оси также на 180°. И вновь повторяют ту же процедуру.

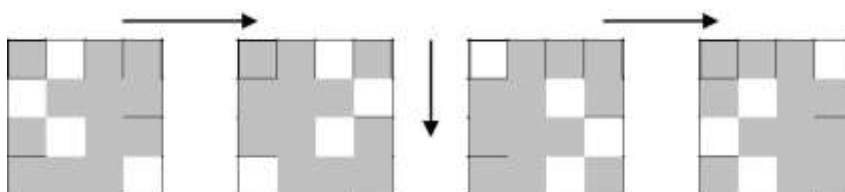


Рис. 1.4. Решетка Кардано

² Кардано Джероламо (1501-1576) - итальянский математик, философ и врач

Диск Альберти

Итальянец Альберти (XVI в.) впервые выдвинул идею "двойного шифрования" – текст, полученный в результате первого шифрования, подвергался повторному зашифрованию. В трактате Альберти был приведен его собственный шифр, который он назвал "шифром, достойным королей". Он утверждал, что этот шифр недешифруем. Реализация шифра осуществлялась с помощью шифровального диска, положившего начало целой серии *многоалфавитных подстановок*. Устройство представляло собой пару дисков – внешний, неподвижный (на нем были нанесены буквы в естественном порядке и цифры от 1 до 4) и внутренний – подвижный – на нем буквы были переставлены. Процесс шифрования заключался в нахождении буквы открытого текста на внешнем диске и замену ее на соответствующую (стоящую под ней) букву шифрованного текста. После шифрования нескольких слов внутренний диск сдвигался на один шаг. Ключом данного шифра являлся порядок расположения букв на внутреннем диске и его начальное положение относительно внешнего диска.

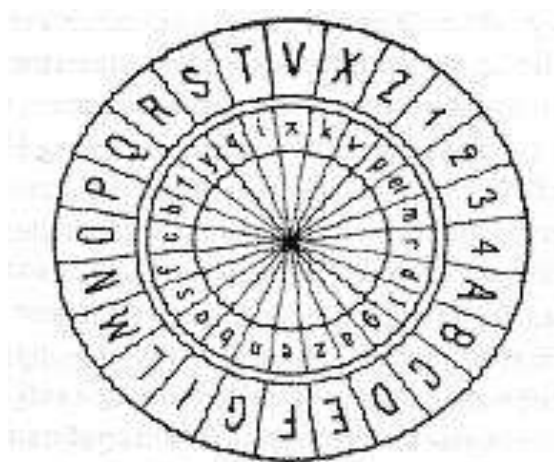


Рис. 1.5. Диск Альберт

Таблица Виженера

Неудобство рассмотренных выше шрифтов моноалфавитных подстановок очевидно, так как в случае использования стандартного алфавита таблица частот встречаемости букв алфавита позволяет определить один или несколько символов, а этого иногда достаточно для вскрытия шифра (“Пляшущие человечки” Конан Дойля или “Золотой жук” Эдгара По). Поэтому использовались различные приемы для того чтобы затруднить дешифрование, например использование “*таблицы Виженера*”, которая представляет собой квадратную таблицу с числом строк и столбцов равным количеству букв алфавита. Чтобы зашифровать какое-либо сообщение выбирают слово – лозунг (например, “монастырь”) и надписывают его над сообщением с необходимым повторением.

Чтобы получить зашифрованный текст, находят очередной знак лозунга, начиная с первого в вертикальном алфавите, а ему соответствующий знак сообщения в горизонтальном алфавите. На пересечении выделенных столбца и строки находим первую букву шифра. Очевидно, что ключом к такому шифру является используемый лозунг.

монастырьмонастырьмон

раскинулосьморешироко

эоякщайюйщовчфшьшы

Таблица Виженера³

АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯ
БВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯА
ВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБ
ГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВ
ДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГ
ЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГД
ЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕ
ЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖ
ИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗ
ЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИ
КЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙ
ЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙК
МНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛ
НОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМ
ОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМН
ПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНО
РСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОП
СТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПР
ТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРС
УФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТ
ФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУ
ХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФ
ЦЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХ
ЧШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦ
ШЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧ
ЩЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШ
ЬЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩ
ЫЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩ
ЭЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫ
ЮЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭ
ЯАБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЫЭЮ

³Блез де Виженер (1523-1596) - французский посол в Риме, написал большой труд о шифрах. Квадратный шифр Виженера на протяжении почти 400 лет не был дешифрован, считался недешифруемым шифром

Одноразовый шифровальный блокнот

Примером нераскрываемого шифра может служить так называемый *одноразовый шифровальный блокнот* – шифр, в основе которого лежит та же идея, что в шифре Цезаря. Назовем *расширенным алфавитом* множество букв алфавита и знаков препинания $\{ . , : ; ! ? () - " <пробел> \}$, число символов расширенного кириллического алфавита в данном варианте будет равно 44. Занумеруем символы расширенного алфавита числами от 0 до 43. Тогда любой передаваемый текст можно рассматривать как последовательность $\{a_n\}$ чисел множества $A = \{0, 1, 2, \dots, 43\}$.

Предположим, что имеем случайную последовательность $\{c_n\}$ из чисел множества A той же длины, что и передаваемый текст – *ключ*.

Складывая по модулю 44 число из передаваемого текста a_n с соответствующим числом из множества ключа c_n :

$$a_n + c_n \equiv b_n \pmod{44}, 0 \leq b_n \leq 43$$

получим последовательность $\{b_n\}$ знаков шифрованного текста. Чтобы получить передаваемый текст, можно воспользоваться тем же ключом:

$$a_n \equiv b_n - c_n \pmod{44}, 0 \leq a_n \leq 43$$

У двух абонентов, находящихся в секретной переписке, имеются два одинаковых блокнота. В каждом из них на нескольких листах напечатана случайная последовательность чисел множества A . Отправитель свой текст шифрует, указанным выше способом, при помощи первой страницы блокнота. Зашифровав сообщение, он уничтожает использованную страницу и отправляет текст сообщения второму абоненту, получатель шифрованного текста расшифровывает его и также уничтожает.

использованный лист блокнота. Нетрудно видеть, что одноразовый шифр не раскрываем в принципе, так как символ в тексте может быть заменен любым другим символом и этот выбор совершенно случаен.

1.3. Методы шифрования

1.3.1. Одноалфавитный метод

Данный метод, пожалуй, самый древний из всех известных методов.

В его основе лежит простой способ шифрования: отправитель и получатель зашифрованного документа заранее договариваются об определенном смещении букв относительно их обычного местоположения в алфавите. Например, для кириллицы, если смещение равно 1, то “А” соответствует букве “Б”, “Б” – “В”, и так далее, а когда алфавит подходит к концу, то начинают брать буквы из начала списка. И выходит, например, следующее: из слова “КОДИРОВАНИЕ” получается “ЛПЕЙСПГБОЙЖ”.

Частным случаем данного метода является ранее рассмотренный шифр Цезаря. Очевидно, что произвольный шифр из класса одноалфавитных методов не является шифром Цезаря (если мощность алфавита текста равна n , то число шифров Цезаря равно n , а число всех одноалфавитных шифров равно $n!$). Однако и для таких методов легко предложить способы дешифрования, основанные на статистических свойствах шифрованных текстов поскольку открытый и закрытый тексты имеют одинаковые статистические характеристики.

В лабораторной работе № 1 рассматриваются два варианта одноалфавитного метода с фиксированным смещением и с произвольным (задаваемым) смещением.

1.3.2. Шифрование методом перестановки символов

Суть этого метода заключается в том, что символы текста переставляются по определенным правилам, при этом используются только символы исходного (незашифрованного) текста.

Перестановки в классической криптографии обычно получаются в результате записи исходного текста и чтения зашифрованного текста по разным путям геометрической фигуры.

Простейшим примером перестановки является запись исходного текста по строкам некоторой матрицы и чтение его по столбцам этой матрицы.

Последовательность заполнения строк и чтения столбцов может быть любой и задается ключом. Таким образом, для матрицы размером 8×8 (длина блока 64 символа) возможно $1.6 \cdot 10^9$ ключей, что позволяет на современных компьютерах путем перебора дешифровать заданный текст. Однако для матрицы размером 16×16 (длина блока 256 символов) имеется $1.4 \cdot 10^{26}$ ключей, и перебор их с помощью современных вычислительных средств весьма затруднителен.

Примером применения метода перестановки может быть также восьмиэлементная таблица, обладающая совокупностью маршрутов, носящих название маршрутов Гамильтона. Последовательность заполнения таблицы каждый раз соответствует нумерации ее элементов. Если длина шифруемого текста не кратна числу элементов, то при последнем заполнении в свободные элементы заносится произвольный символ. Выборка из таблицы для каждого заполнения может выполняться по своему маршруту, при этом маршруты могут использоваться как последовательно, так и в порядке, задаваемом ключом.

Для методов перестановки характерны простота алгоритма, возможность программной реализации и низкий уровень защиты, так как при большой длине исходного текста в его зашифрованном варианте проявляются статистические закономерности ключа, что и позволяет его быстро раскрыть. Другой недостаток этих методов – легкое раскрытие, если уда-

ется направить в систему для шифрования несколько специально подобранных сообщений. Так, если длина блока исходном тексте равна K символам, то для раскрытия ключа достаточно пропустить через шифровальную систему $K-1$ блоков исходного текста, в которых все символы, кроме одного, одинаковы.

1.3.3. Шифрование инверсными символами (по дополнению до 255)

Данный метод шифрования, является частным случаем одноалфавитной замены в алфавите мощности 256. Суть метода заключается в замене символа ASCII-кодировки с номером i на символ с номером $255-i$. Аналогично проводится и операция расшифрования.

1.3.4. Многоалфавитные методы шифрования

Многоалфавитное шифрование (многоалфавитная замена) заключается в том, что для последовательных символов шифруемого текста используются одноалфавитные методы с различными ключами.

Например, первый символ заменяется по методу Цезаря со смещением 14, второй – со смещением 10, и так далее до конца заданного ключа. Затем процедура продолжается периодически. Более общей является ситуация, когда используется не шифр Цезаря, а последовательность произвольных подстановок, соответствующих одноалфавитным методам.

Более наглядным примером подобного шифрования является метод гаммирования. Данный способ преобразования заключается в том, что символы закрываемого текста последовательно складываются с символами некоторой специальной последовательности именуемой гаммой. Такое преобразование иногда называют наложением гаммы на открытый текст.

Собственно процедура наложения может осуществляться одним из двух способов:

1) Символы закрываемого текста и гаммы заменяются цифровыми эквивалентами а затем складываются по модулю K , где K – количество символов алфавита,

$T_{ш} = (T_o \square T_z) \bmod K$ где $T_{ш}$ – шифротекст,
 T_o – открытый текст,
 T_z – гамма.

2) Символы текста и гаммы представляются в двоичных кодах, а затем каждая пара двоичных разрядов складывается по модулю 2.

Стойкость шифрования методом гаммирования определяется, главным образом, качеством гаммы, которое определяется двумя характеристиками: длиной периода и случайностью распределения по периоду.

Длиною периода гаммы называется минимальное количество символов, после которого последовательность начинает повторяться. Случайность распределения символов по периоду означает отсутствие закономерностей между появлением различных символов в пределах периода.

В лабораторной работе № 1 рассматриваются три варианта многоалфавитного метода – с фиксированным ключом, с ключом фиксированной длины и с ключом произвольной длины.

1.3.5. Основные требования, которые предъявляются к методам шифрования информации

1) Сложность и трудоемкость процедур шифрования и расшифрования должны определяться в зависимости от степени секретности защищаемых данных.

2) Надежность закрытия должна быть такой, чтобы секретность не нарушалась даже в том случае, когда злоумышленнику известен способ закрытия.

3) Способ закрытия и набор используемых служебных данных (ключевых установок) не должны быть слишком сложными. Затраты на защитные преобразования должны быть приемлемые при заданном уровне сохранности информации.

4) Выполнение процедур прямого и обратного преобразования должно быть формальным и как можно проще.

5) Процедуры прямого и обратного преобразования не должны зависеть от длины сообщения.

6) Ошибки, возникающие в процессе преобразования, не должны распространяться по системе и вызывать потерю информации. Из-за появления ошибок передачи зашифрованного сообщения по каналам связи не должна исключаться возможность надежной расшифровки текста на приемном конце.

7) Избыточность сообщений, вносимая закрытием должна быть как можно меньшей.

8) Объем ключа не должен затруднять его запоминание и пересылку.

1.3.6. Гистограмма текста

Одним из наиболее известных методов криптоанализа является изучение статистических характеристик шифрованных текстов. Графическое отображение совокупности частот встречаемости символов в тексте называют гистограммой этого текста.

Предположим, что мы имеем дело с методом одноалфавитного шифрования. Зная частоту встречаемости букв в алфавите, можно предположить, какая буква была заменена на данную. Например, часто встречаемая буква “О” заменена на редко встречающуюся букву “Щ”.

Для наглядности, в лабораторной работе № 1 используются двойные гистограммы, отображающие частоту встречаемости символов в исходном и зашифрованном текстах.

Следует иметь в виду, что вид гистограммы для стандартного распределения зависит от вида исходного текста следующим образом: если

исходный текст содержит символы кириллицы и латинского алфавита, то выводится статистическое распределение для кириллицы и латиницы, если только кириллицы (латиницы) то выводится статистическое распределение для кириллицы (латиницы).

Лабораторная работа № 1

Использование классических криптоалгоритмов подстановки и перестановки для защиты текстовой информации

Цель работы: формирование у студентов практических умений исследования классических криптографических алгоритмов многоалфавитной подстановки, многоалфавитной подстановки и перестановки для защиты текстовой информации и формирование гистограмм, отображающих частоту встречаемости символов в тексте для криптоанализа классических шифров.

1. Описание лабораторной работы

Для выполнения лабораторной работы необходимо запустить программу *L_LUX.EXE*. На экране дисплея появляется окно, с размещенным в его центре текстовым редактором (для отображения зашифрованных и расшифрованных текстов), в верхней строке окна расположено главное меню, позволяющее пользователю выполнить требуемое действие, чуть ниже основного меню расположена панель инструментов (для управления быстрыми командными кнопками и другими “горячими” элементами управления), а в самом низу окна, под текстовым редактором, находится строка состояния, в которой указывается подсказка и выводится дополнительная информация. Клавиши панели инструментов, для удобства, снабжены всплывающими подсказками.

Для того чтобы попасть в основное меню, необходимо нажать клавишу *F10*. Передвижение по главному меню осуществляется клавишами перемещения курсора. Чтобы вызвать пункт меню, нужно нажать клавишу “ENTER”, вернуться в главное меню или вовсе выйти из него – “ESC”.

А теперь более подробно рассмотрим каждый из пунктов главного меню.

1.1. Редактор

Данный пункт основного меню содержит подпункты: создать документ, открыть файл, сохранить файл, выход из программы.

Предварительно, сразу после запуска программы, текстовый редактор недоступен, также недоступными являются почти все пункты главного меню (кроме создания документа, открытия файла, выхода из программы, информации о программе) и большая часть клавиш панели управления (за исключением создания документа, открытия файла и выхода из программы).

Создать документ (Ctrl+N) – данный подпункт делает доступным для работы тестовый редактор (пользователь получает право создать свой текстовый файл, который впоследствии можно будет использовать при работе с программой), также появляется возможность использовать все недоступные до этого пункты основного меню и клавиши панели управления.

Открыть файл (Ctrl+L) – при выборе этого пункта появляется диалоговое окно, предоставляющее возможность выбора файла для загрузки. При этом содержимое файла будет отображено в окне редактора текстов.

Аналогично пункту “Создать документ” доступным для работы становится текстовый редактор с отображаемым текстом, а также появляется возможность использовать все недоступные до этого пункты основного меню и клавиши панели управления.

Сохранить файл (Ctrl+S) - при выборе этого пункта появляется диалоговое окно, позволяющее сохранить на диск содержимое редактора текстов.

Выход из программы (Ctrl+X) - при выборе этого пункта появляется диалоговое окно, позволяющее выйти из программы.

1.2. Гистограмма

Вывод на экран двух гистограмм, отображающих частоту встречаемости символов в тексте.

Внимание ! До выполнения шифрования и дешифрования вызывать гистограмму не имеет смысла, так как еще не сформированы тексты, для которых будет просматриваться гистограмма.

Имеется возможность просмотра следующих сочетаний гистограмм:

- гистограммы исходного и зашифрованного файла,
- гистограммы зашифрованного и расшифрованного файла,
- гистограммы стандартного распределения и зашифрованного текста,
- гистограммы стандартного распределения и расшифрованного текста.

В гистограммах с целью масштабирования используются левая и правая клавиши мыши.

Например, после шифрования текста большого объема пользователь хочет посмотреть гистограммы исходного и зашифрованного файла. Поскольку размеры текста достаточно большие, то на экран будут выведены две гистограммы с большим количеством столбцов в каждой (столбец соответствует одному символу текста), однако трудно будет сказать, какой из этих столбцов соответствует тому или иному символу текста, и какова частота встречаемости данного символа. Поэтому у пользователя есть возможность увеличить масштаб любой из двух гистограмм с целью более точного определения требуемого значения частоты встречаемости конкретного символа. Для этого необходимо навести указатель мыши на левую границу того участка гистограммы, который требуется увеличить, затем нажать левую клавишу мыши, и не отпуская ее растянуть прямоугольник так, чтобы его нижний правый угол совпал с правой границей увеличиваемого участка гистограммы. После этого следует отпустить левую клавишу мыши и на экране появится увеличенное изображение нужного участка.

Причем, нажав и не отпуская правую клавишу мыши, можно перемещать гистограмму в любом направлении с целью изучения всего полученного распределения в увеличенном масштабе.

Для того, чтобы от увеличенного масштаба вернуться к исходному виду, нужно привести указатель мыши на гистограмму, затем нажать левую клавишу мыши, и, не отпуская ее, снизу вверх растянуть небольшой по размерам прямоугольник, после этого следует отпустить левую клавишу мыши и на экране появится исходное изображение гистограммы.

1.3. Шифрование

Выполнение шифрования текстового файла осуществляется одним из семи методов, рассматриваемых в лабораторной работе.

1. Одноалфавитный метод (с фиксированным смещением).
2. Одноалфавитный метод с задаваемым смещением (от 2 до 20).
3. Перестановка символов.
4. По дополнению до 255 (инверсный метод).
5. Многоалфавитный метод (с фиксированным ключом).
6. Многоалфавитный метод с ключом фиксированной длины.
7. Многоалфавитный метод с ключом произвольной длины.

Выбор метода шифрования производится как мышкой, так и клавишами перемещения курсора и клавишей “ENTER”.

Затем появляется окно в котором в зависимости от метода шифрования требуется указать те или иные параметры, и либо подтвердить процесс кодировки, либо отказаться от него. После этого в окне редактора будет выдан зашифрованный текст.

1.4. Расшифрование

Аналогично предыдущему пункту выбирается метод расшифрования (должен соответствовать методу, которым был зашифрован файл).

Снова появляется окно, в котором в зависимости от метода расшифрования требуется указать те или иные параметры, и либо подтвердить процесс расшифрования, либо отказаться от него.

После этого в окно редактора будет выдан расшифрованный текст. При правильном расшифровании, полученный текст совпадает с исходным.

1.5. Дополнительная информация

Программа предусматривает возможность посмотреть дополнительную информацию ('Помощь Ctrl+F9'), справочную информацию об используемых методах шифрования ('О методах Ctrl+F10'), сведения о программе ('О программе Ctrl+F11').

2. Формируемые компетенции

Данная лабораторная работа направлена на формирование следующих компетенций выпускника бакалавриата:

ОПК-1: способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ПК-5: способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.

3. Теоретическая часть

Для выполнения лабораторной работы необходимо запустить программу L_LUX. На экране дисплея появится окно с размещенным в его центре текстовым редактором. В верхней строке окна расположено главное меню, позволяющее пользователю выполнить требуемое действие. Чуть ниже основного меню расположена панель инструментов, а в самом низу окна, под текстовым редактором находится строка состояния, в которой указывается подсказка и выводится дополнительная информация. Кла-

виши панели инструментов для удобства снабжены всплывающими подсказками.

Для того чтобы попасть в основное меню, не обходимо нажать клавишу F10. Передвижение по главному меню осуществляется мышью.

Пример работы с программой.

Рассмотрим одноалфавитное шифрование с фиксированным ключом.

Загрузите в окно редактора исходный текст (любой) длиной от 10 до 15 предложений. Сохраните его, это необходимо для последующей работы с этим файлом. Затем вызовите пункт меню ШИФРОВАНИЕ, выберите одноалфавитный метод (с фиксированным смещением). В появившемся окне нажмите клавишу ЗАШИФРОВАТЬ. После того как шифрование выполнено, можно в редакторе просмотреть зашифрованный текст.

Перейдите к пункту меню ГИСТОГРАММА. Выберите тип гистограмм, отображающий гистораммы исходного и шифрованного текста. Проанализируйте гистограммы. Они должны иметь примерно одинаковый вид.

Чтобы узнать ключ шифра (смещение второго алфавита относительно первого), необходимо по гистограммам найти символы, имеющие одинаковую частоту встречаемости. Например, самый частый символ в первой гистограмме при шифровании должен перейти в самый частый символ во второй гистограмме. Таким образом, найдя два самых часто встречаемых символа в обеих гистограммах, можно по стандартной таблице ASCII-кодов вычислить смещение. Зная смещение и таблицу кодировки символов, текст можно легко дешифровать. Вызвав меню ДЕШИФРОВАНИЕ, можно провести те же действия в автоматическом режиме.

Примечание. При шифровании и дешифровании из таблицы кодировки не используются символы с кодами 176-223 и 240-255, то есть при ручной расшифровке эти символы следует пропускать и считать, что, например, символ Я имеет код 159, а 223, аналогично П не 175, а 239.

Иногда в гистограммах под столбцами, показывающими частоту встречаемых символов, изображены не сами символы, а их табличные коды в квадратных скобках.

4. Оборудование и материалы

К аппаратному обеспечению лабораторной работы относятся компьютерные классы с установленным сетевым оборудованием, а также

- операционная система Windows;
- программа L_LUX в среде C++.

5. Указания по технике безопасности

Требования безопасности перед началом работ:

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории;
- убедиться в целостности электрических розеток и разъемов. Запрещается работать с неисправным электроинструментом, разбивать и самостоятельно ремонтировать его;
- в лаборатории необходимо быть в сменной обуви;
- включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

Требования безопасности во время работы:

- выполняя лабораторную работу, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и дискеты, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отсоединение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса.

Требования безопасности по окончании работы:

- по команде преподавателя студенты обязаны выключить компьютер и составляющие вычислительного комплекса;
- рабочее место привести в порядок.

6. Задания

Задание №1. Для одноалфавитного метода с фиксированным смещением определить установленное в программе смещение. Для этого следует:

- просмотреть предварительно созданный с помощью редактора свой текстовый файл;
- выполнить для этого файла шифрование;
- просмотреть в редакторе зашифрованный файл;
- просмотреть гистограммы исходного и зашифрованного текстов;
- описать гистограммы (в чем похожи, в чем отличаются) и определить, с каким смещением было выполнено шифрование;
- расшифровать зашифрованный текст:
 - С помощью гистограммы, после чего проверить в редакторе правильность расшифрования;
 - Вручную с помощью гистограмм, описать и объяснить процесс дешифрования.

В отчете для каждого метода шифрования описывается последовательность выполняемых действий, имена всех использованных файлов, полученные гистограммы, указывается найденное смещение, описывается процесс дешифрования.

Преподавателю предоставляется отчет о проделанной работе и все использованные и созданные файлы.

Задание №2. Для одноалфавитного метода с заданным смещением (шифр Цезаря) следует:

- выполнить шифрование с произвольным смещением для своего исходного текста;
- просмотреть и описать гистограммы исходного и зашифрованного

текстов, определить смещение для нескольких символов;

- расшифровать текст с помощью программы;
- дешифровать зашифрованный шифром Цезаря текст с помощью программы методом перебора смещения; указать в отчете с каким смещением был зашифрован файл.

Задание №3. Для метода перестановки символов дешифровать зашифрованный файл. Для этого необходимо определить закон перестановки символов открытого текста. Создайте небольшой файл длиной в несколько символов с известным вам текстом, зашифруйте его. Посмотрите гistogramмы, опишите их, ответьте можно ли извлечь из них полезную для дешифрования информацию. Сравните с помощью редактора ваш исходный текст и зашифрованный текст и определите закон перестановки символов.

Дешифруйте файл:

- вручную и объясните ваши действия;
- с помощью программы.

Задание №4. Для инверсного кодирования (по дополнению до 255)

- выполните шифрование для своего произвольного файла;
- посмотрите гistogramмы исходного и зашифрованного текстов, опишите гistogramмы и определите смещение для нескольких символов;
- дешифруйте зашифрованный текст, проверьте в редакторе правильность дешифрования.

Задание №5. Для многоалфавитного шифрования с фиксированным ключом определите, сколько одноалфавитных методов и с каким смещением используется в программе. Для этого нужно создать свой файл, состоящий из строки одинаковых символов, выполнить для него шифрование и по гistogramме определить способ шифрования и набор смещений.

Задание №6. Для многоалфавитного шифрования с ключом фиксированной длины:

- выполните шифрование и определите по гistogramме, какое смещение получает каждый символ для файла, состоящего из строки одинаковых

ВЫХ СИМВОЛОВ;

- выполните шифрование и расшифрование для файла произвольно текста;

- просмотрите и опишите гистограммы исходного и зашифрованного текстов; ответьте, какую информацию можно получить и гистограмм.

Задание №7. Для многоалфавитного шифрования с произвольным паролем задание полностью аналогично заданию 6.

7. Содержание отчета

Отчет должен содержать:

1. наименование работы;
2. цель работы;
3. рабочие схемы;
4. основные соотношения и расчетные формулы;
5. таблицы;
6. выводы по работе.

8. Контрольные вопросы

Привести в отчете ответы на контрольные вопросы в соответствии с номером варианта, указанным преподавателем (табл. 1.1).

Таблица 1.1 – Контрольные вопросы

Номер варианта	Контрольные вопросы
1, 5, 7, 3, 9, 18, 28	Какие вы знаете методы криптографической защиты файлов?
2, 4, 6, 8, 20, 22, 24, 26, 30	В чем преимущества и недостатки одноалфавитных методов?
11, 13, 15, 10, 17, 19, 27	Если необходимо зашифровать текст, содержащий важную информацию, какой метод из рассмотренных вы выберете? Обоснуйте свой выбор
12, 14, 16, 21, 23, 25, 29	Целесообразно ли повторно применять для уже зашифрованного текста:

	а) метод многоалфавитного шифрования; б) метод Цезаря?
--	---

9. Список литературы, рекомендованный к использованию по данной теме

1. Жук, А.П., Жук, Е.П., Лепешкин, О.М., Тимошкин, А.И. Защита информации: Учебное пособие для бакалавров [Текст] / А.П. Жук и др. – М.: ИНФРА-М, 2015. -392с.

2. Шаньгин, В.Ф. Защита компьютерной информации. Эффективные методы и средства [Текст] / В.Ф. Шаньгин – М.: ДМК Пресс, 2012. – 544 с.

Лабораторная работа № 2

Исследование различных методов защиты текстовой информации и их стойкости на основе подбора ключей

Цель лабораторной работы: исследования методов шифрования/расшифрования перестановкой символов, подстановкой, гаммированием, использованием таблицы Виженера, сравнение стойкости различных методов на основе атак путем перебора всех возможных ключей.

В лабораторной работе рассматривается способ вскрытия шифра, основанный на переборе всех вариантов ключа. Критерием правильности варианта служит наличие в тексте «вероятного слова».

Перебирается множество всех возможных ключей, шифрованный текст расшифровывается на каждом ключе. В получившемся «псевдооткрытом» тексте ищется вероятное слово. Если такого слова нет, текущий текст бракуется и осуществляется переход к следующему ключу. Если такое слово найдено, на экран выводится вариант ключа. Затем перебор ключей продолжается до тех пор, пока не исчерпается все множество вариантов. Возможно обнаружение нескольких ключей, при которых в «псевдооткрытых текстах» имеется вероятное слово.

После завершения перебора необходимо расшифровать текст на найденных ключах. «Псевдооткрытый текст» выводится на экран для визуального контроля.

Если оператор признает текст открытым, то работа по вскрытию заканчивается. Иначе, данный вариант ключа бракуется и осуществляется переход к следующему ключу.

1. Формируемые компетенции

Данная лабораторная работа направлена на формирование следующих компетенций выпускника бакалавриата:

ОПК-1: способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ПК-5: способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.

2. Теоретическая часть

Для выполнения лабораторной работы необходимо запустить программу *LAB_RAB.exe*, используемую для шифрования/расшифрования, а также дешифрования (методом протяжки вероятного слова) файлов. Система реализует следующие функции:

- ввод, удаление и селекция ключей пользователя;
- поддержка списка ключей;
- шифрование и расшифрование текста;
- дешифрование текста путем подбора ключей, методом протяжки вероятного слова.

Система поддерживает следующие методы криптографического преобразования информации:

- замена;
- перестановка;
- гаммирование;
- таблица Виженера.

При запуске утилит шифрования и расшифрования у пользователя запрашивается подтверждение на правильность выбранного метода для работы и соответствия заданного ключа целям пользователя (также всегда при изменении файла в текстовом редакторе выдается запрос на сохранение изменений при каждом шаге, дальнейшее развитие которого приведет к необратимым изменениям в файле и потере изначальной информации).

Описание интерфейса.

- Окно текстового редактора с широким набором дополнительных функций.
- Таблица всех ключей, введенных в систему с быстрым доступом для ввода, удаления или выбора текущего ключа.
- Список всех методов шифрования для быстрого и удобного переключения между ними.

- Основное меню (вверху экрана).
- Дополнительное меню (вызывается нажатием правой кнопки мыши).
- Набор вспомогательных кнопок для быстрого и удобного интерфейса.
- Поля вывода текущего состояния системы:
 - текущий ключ;
 - вероятное слово;
 - сила ключа для протяжки.

Пример работы с программой

Внимание! Будьте внимательны при установке параметров работы, так как в процессе вычисления по ходу работы эти параметры изменить уже не удастся. После запуска программы абсолютно все рабочие поля пустые и необходимо провести первоначальные настройки для работоспособности системы.

1. Вводится список ключей.
2. Вводится вероятное слово (необязательно вначале, до его ввода все меню запуска протяжки все равно недоступны).
3. Выбирается необходимый метод шифрования.
4. Загружается исходный или зашифрованный файл (открываются соответствующие меню для шифрования и расшифрования).
5. Запускается необходимый процесс:
 - шифрование;
 - расшифрование;
 - протяжка вероятного слова;
 - конвертация текста.
6. Продолжение работы в любом порядке в соответствии с описанными пунктами.

7. При завершении работы не забудьте сохранить необходимые результаты (при закрытии и загрузке новых файлов система автоматически запрашивает подтверждение на запись).

Шифрование

1. Открыть файл.
2. Внести необходимые изменения.
3. Настроить соответствующие параметры:
 - тип шифрования;
 - ключ;
 - прочие.
4. Запустить процесс шифрования через пункт меню УТИЛИТЫ \ ЗАШИФРОВАТЬ F5.

Внимание! При шифровании файла все внесенные пользователем изменения до текущего момента времени будут сохранены на жестком диске.

Расшифрование

1. Открыть файл.
2. Произвести необходимые изменения.
3. Настроить соответствующие параметры:
 - тип шифрования;
 - прочие.
4. Запустить процесс расшифрования через пункт меню УТИЛИТЫ \ РАСШИФРОВАТЬ F6.

Внимание! При расшифровании файла все проведенные пользователем изменения до текущего момента времени будут сохранены на жестком диске.

Протяжка вероятного слова (дешифрование)

Внимание! Мощность ключа задается заранее в опции "сила ключа".
Длина ключа значительно влияет на время протяжки вероятного слова (в худшем случае имеем дело с логарифмическим алгоритмом).

1. Вводится вероятное слово (длинной от 1(3) до 9)
2. Для отделения вновь найденных ключей от предыдущих между ними добавляется надпись "подбор".

3. Перебор ключей.
4. Расшифровывается первая вся строка текста по текущему ключу.
5. Порциями, равными длине вероятного слова, сравнивается содержимое этой строки со значением вероятного слова.
6. Если найдено хоть одно совпадение, запоминаем ключ.
7. Переходим к новому ключу.
9. Результаты должны содержаться в списке ключей. Если совпадений не найдено, в список ключей ничего не добавляется.

Операции с ключами

С базой ключей могут осуществляться следующие действия:

- добавить новый ключ;
- удалить одну запись;
- изменить активную запись;
- очистить всю таблицу введенных ключей .

<i><u>Примечание.</u> Под словами "работа с таблицей ключей" имеются ввиду ключи, введенные для использования в двух методах (гаммирования и таблица Виженера).</i>
--

Ключи для перестановки

В каждый момент времени в системе может быть только один текущий ключ для перестановки.

Правила ввода ключа для перестановки:

1. При переключении в списке поддерживаемых системой методов шифрования на пункт "Перестановка" вызывается окно ввода ключа перестановки. Окно состоит из двух кнопок (Отмены и Выхода без изменений и кнопки Enter – подтверждение установленной длины ключа) и окна задания длины ключа для перестановки.

2. В окне задания длины ключа необходимо выбрать необходимую длину (параметры заменяются в пределах 1 ... 9), и подтвердить желание использовать ключ именно такой длины.

3. После подтверждения в окне высветятся кнопки с цифрами на лицевой стороне (в количестве, равном длине ключа), при нажатии на кнопку происходит фиксация кнопки (ее обесцвечивание) для

невозможности ее дальнейшего использования (так как все цифры в ключе перестановки должны быть неповторяющимися).

4. После перебора всех кнопок система запоминает введенный ключ, выводит его в поле ввода ключей и выходит из окна ввода ключа перестановки в окно основной программы.

4. Оборудование и материалы

К аппаратному обеспечению лабораторной работы относятся компьютерные классы с установленным сетевым оборудованием, а также

- операционная система Windows;
- программа *LAB_RAB* в среде C++.

5. Указания по технике безопасности

Требования безопасности перед началом работ:

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории;
- убедиться в целостности электрических розеток и разъемов. Запрещается работать с неисправным электроинструментом, разбивать и самостоятельно ремонтировать его;
- в лаборатории необходимо быть в сменной обуви;
- включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

Требования безопасности во время работы:

- выполняя лабораторную работу, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и дискеты, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отсоединение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса.

Требования безопасности по окончании работы:

- по команде преподавателя студенты обязаны выключить компьютер и составляющие вычислительного комплекса;
- рабочее место привести в порядок.

6. Задания

Для выполнения лабораторной работы необходимо запустить программу *LAB_RAB.exe*, используемую для шифрования/расшифрования, а также дешифрования (методом протяжки вероятного слова) файлов.

Система реализует следующие функции:

- ввод, удаление и селекция ключей пользователя;
- поддержка списка ключей;
- шифрование и расшифрование текста;
- дешифрование текста путем подбора ключей, методом протяжки вероятного слова.

Система поддерживает следующие методы криптографического преобразования информации:

- замена;
- перестановка;
- гаммирование;
- таблица Виженера.

При запуске утилит шифрования и расшифрования у пользователя запрашивается подтверждение на правильность выбранного метода для работы и соответствия заданного ключа целям пользователя (также всегда при изменении файла в текстовом редакторе выдается запрос на сохранение изменений при каждом шаге, дальнейшее развитие которого приведет к необратимым изменениям в файле и потере изначальной информации).

Описание интерфейса:

1. Окно текстового редактора с широким набором дополнительных функций.

2. Таблица всех ключей, введенных в систему с быстрым доступом для ввода, удаления или выбора текущего ключа.

3. Список всех методов шифрования для быстрого и удобного переключения между ними.

4. Основное меню (вверху экрана).

5. Дополнительное меню (вызывается нажатием правой кнопки мыши).

6. Набор вспомогательных кнопок для быстрого и удобного интерфейса.

7. Поля вывода текущего состояния системы:

- текущий ключ;
- вероятное слово;
- сила ключа для протяжки.

Пример работы с программой

Внимание! Будьте внимательны при установке параметров работы, так как в процессе вычисления по ходу работы эти параметры изменить уже не удастся.

После запуска программы абсолютно все рабочие поля пустые и необходимо провести первоначальные настройки для работоспособности системы.

1. Вводится список ключей.

2. Вводится вероятное слово (необязательно вначале, до его ввода все меню запуска протяжки все равно недоступны).

3. Выбирается необходимый метод шифрования.

4. Загружается исходный или зашифрованный файл (открываются соответствующие меню для шифрования и расшифрования).

5. Запускается необходимый процесс:

- шифрование;
- расшифрование;
- протяжка вероятного слова;

– конвертация текста.

6. Продолжение работы в любом порядке в соответствии с описанными пунктами.

7. При завершении работы не забудьте сохранить необходимые результаты (при закрытии и загрузке новых файлов система автоматически запрашивает подтверждение на запись).

Шифрование

1. Открыть файл.

2. Внести необходимые изменения.

3. Настроить соответствующие параметры:

– тип шифрования;

– ключ;

– прочие.

4. Запустить процесс шифрования через пункт меню

УТИЛИТЫ \ ЗАШИФРОВАТЬ F5.

Внимание! При шифровании файла все внесенные пользователем изменения до текущего момента времени будут сохранены на жестком диске.

Расшифрование

1. Открыть файл.

2. Произвести необходимые изменения.

3. Настроить соответствующие параметры:

– тип шифрования;

– прочие.

4. Запустить процесс расшифрования через пункт меню

УТИЛИТЫ \ РАСШИФРОВАТЬ F6.

Внимание! При расшифровании файла все проведенные пользователем изменения до текущего момента времени будут сохранены на жестком диске.

Протяжка вероятного слова (дешифрование)

Внимание! Мощность ключа задается заранее в опции "сила ключа".

Длина ключа значительно влияет на время протяжки вероятного слова (в худшем случае имеем дело с логарифмическим алгоритмом).

1. Вводится вероятное слово (длинной от 1(3) до 9)
2. Для отделения вновь найденных ключей от предыдущих между ними добавляется надпись "подбор".
3. Перебор ключей.
4. Расшифровывается первая вся строка текста по текущему ключу.
5. Порциями, равными длине вероятного слова, сравнивается содержимое этой строки со значением вероятного слова.
6. Если найдено хоть одно совпадение, запоминаем ключ.
7. Переходим к новому ключу.
8. Переходим к следующей строке.
9. Результаты должны содержаться в списке ключей. Если совпадений не найдено, в список ключей ничего не добавляется.

Операции с ключами

С базой ключей могут осуществляться следующие действия:

- добавить новый ключ;
- удалить одну запись;
- изменить активную запись;
- очистить всю таблицу введенных ключей.

Примечание. Под словами "работа с таблицей ключей" имеются ввиду ключи, введенные для использования в двух методах (гаммирования и таблица Виженера).

Ключи для перестановки

В каждый момент времени в системе может быть только один текущий ключ для перестановки.

Правила ввода ключа для перестановки:

1. При переключении в списке поддерживаемых системой методов шифрования на пункт "Перестановка" вызывается окно ввода ключа пере-

становки. Окно состоит из двух кнопок (Отмены и Выхода без изменений и кнопки Enter – подтверждение установленной длины ключа) и окна задания длины ключа для перестановки.

2. В окне задания длины ключа необходимо выбрать необходимую длину (параметры заменяются в пределах 1 ... 9), и подтвердить желание использовать ключ именно такой длины.

3. После подтверждения в окне высветятся кнопки с цифрами на лицевой стороне (в количестве, равном длине ключа), при нажатии на кнопку происходит фиксация кнопки (ее обесцвечивание) для невозможности ее дальнейшего использования (так как все цифры в ключе перестановки должны быть неповторяющимися).

4. После перебора всех кнопок система запоминает введенный ключ, выводит его в поле ввода ключей и выходит из окна ввода ключа перестановки в окно основной программы.

Задание №1.

1. Ознакомиться с описанием лабораторной работы и заданием.

2. Выполнить настройку программы: выбрать метод шифрования; ввести ключи для всех методов; ввести вероятное слово; осуществить все остальные системные настройки.

Задание №2. Для метода замены (одноалфавитного метода):

- выбрать данный алгоритм в списке доступных методов шифрования;
- установить необходимое смещение;
- открыть произвольный файл;
- просмотреть содержимое исходного файла;
- выполнить для этого файла шифрование (при необходимости можно задать имя зашифрованного файла);
- просмотреть в редакторе зашифрованный файл;
- ввести вероятное слово;
- ввести вероятную длину ключа (кроме метода замены);

- подобрать ключ;
- выполнить расшифрование со всеми найденными ключами;
- найти в каком-либо из расшифрованных файлов правильно расшифрованное ключевое слово;
- расшифровать файл исходным ключом;
- проверить результат.

Задание №3. Для метода перестановки:

- выбрать метод перестановки;
- в открывшемся окне ввода ключа перестановки символов указать сначала длину этого ключа, а затем из появившихся кнопок составить необходимую комбинацию для ключа, нажимая на кнопки в заданном порядке; при этом уже использованные кнопки становятся недоступными для предотвращения их повторного ввода;
- далее действия полностью соответствуют изложенным в предыдущем пункте задания.

Задание №4. Для метода гаммирования:

- выбрать метод гаммирования;
- ввести ключ гаммирования;
- полностью повторить из задания 3 второй пункт.

Задание №5. Для таблицы Виженера все действия повторяются из зад. 4 (метод гаммирования).

В отчете для каждого метода шифрования описывается последовательность выполняемых действий, указываются имена всех использованных файлов, исходные и найденные ключи, описывается процесс дешифрования.

Преподавателю предоставляется отчет о проделанной работе и все использованные файлы.

7. Содержание отчета

В отчете для каждого метода шифрования описывается последовательность выполняемых действий, указываются имена всех использованных

ных файлов, исходные и найденные ключи, описывается процесс дешифрования.

Преподавателю предоставляется отчет о проделанной работе и все использованные файлы.

8. Контрольные вопросы

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Чем отличается “псевдооткрытый” текст (текст, полученный при расшифровке по ложному ключу) от настоящего открытого текста?
2,4,6,8, 20,22,24, 26,30	Как зависит время вскрытия шифра описанным выше способом подбора ключей от длины “вероятного” слова?
11,13,15, 10,17,19, 27	Зависит ли время вскрытия шифра гаммирования (или таблицы Виженера) от мощности алфавита гаммы?
12,14,16 21,23,25, 29	В чем недостатки метода дешифрования с использованием протяжки вероятного слова?

9. Список литературы, рекомендованный к использованию по данной теме

1. Жук, А.П., Жук, Е.П., Лепешкин, О.М., Тимошкин, А.И. Защита информации: Учебное пособие для бакалавров [Текст] / А.П. Жук и др. – М.: ИНФРА-М, 2015. -392с.

2. Шаньгин, В.Ф. Защита компьютерной информации. Эффективные методы и средства [Текст] / В.Ф. Шаньгин – М.: ДМК Пресс, 2012. – 544 с.

Лабораторная работа № 3

Стандарт симметричного шифрования *Aes Rijndael*

Цель работы: исследования принципов шифрования, используемыми в алгоритме симметричного шифрования ***AES RIJNDAEL*** и умений делать самостоятельные выводы по результатам исследований.

2. Формируемые компетенции

Данная лабораторная работа направлена на формирование следующих компетенций выпускника бакалавриата:

ОПК-1: способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ПК-5: способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.

3. Теоретическая часть

Состояние, ключ шифрования и число циклов. Rijndael представляет собой итеративный блочный шифр, имеющий переменную длину блоков и различные длины ключей. Длина ключа и длина блока могут быть независимо друг от друга 128, 192 или 256 бит.

Разнообразные преобразования работают с промежуточным результатом, называемым Состоянием (State).

Состояние можно представить в виде прямоугольного массива байтов. Этот массив имеет 4 строки, а число столбцов обозначено как Nb и равно длине блока, деленной на 32.

Ключ шифрования также представлен в виде прямоугольного массива с четырьмя строками. Число столбцов обозначено как Nk и равно длине ключа, деленной на 32. Это показано на рис. 1.

В некоторых случаях ключ шифрования показан как линейный массив четырехбайтовых слов. Слова состоят из 4-х байтов, которые находятся в одном столбце (при представлении в виде прямоугольного массива).

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$	$a_{0,4}$	$a_{0,5}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$

$k_{0,0}$	$k_{0,1}$	$k_{0,2}$	$k_{0,3}$
$k_{1,0}$	$k_{1,1}$	$k_{1,2}$	$k_{1,3}$
$k_{2,0}$	$k_{2,1}$	$k_{2,2}$	$k_{2,3}$
$k_{3,0}$	$k_{3,1}$	$k_{3,2}$	$k_{3,3}$

Рисунок 1. Пример представления Состояния ($Nb=6$) и Ключа шифрования ($Nk=4$)

Входные данные для шифра обозначаются как байты состояния в порядке $a_{0,0}$, $a_{1,0}$, $a_{2,0}$, $a_{3,0}$, $a_{0,1}$, $a_{1,1}$, $a_{2,1}$, $a_{3,1}$... После завершения действия шифра выходные данные получаются из байтов состояния в том же порядке. Число циклов обозначено как Nr и зависит от значений Nb и Nk .

Таблица 4.1. Число циклов (Nr) как функция от длины ключа и длины блока

Nr	$Nb = 4$	$Nb = 6$	$Nb = 8$
$Nk = 4$	10	12	14
$Nk = 6$	12	12	14
$Nk = 8$	14	14	14

Цикловое преобразование. Цикловое преобразование состоит из четырех различных преобразований: замена байт, сдвиг строк, замешивание столбцов, добавление циклового ключа. Последний цикл отличается от простого цикла только отсутствием замешивания столбцов.

```

Round (State, RoundKey)
{
  ByteSub(State); // замена байт
  ShiftRow(State); // сдвиг строк
  MixColumn(State); // замешивание столбцов
  AddRoundKey(State, RoundKey); // добавление циклового ключа
}

```

Последний цикл шифра немного отличается:

```
FinalRound(State, RoundKey)
{
  ByteSub(State); // замена байт
  ShiftRow(State); // сдвиг строк
  AddRoundKey(State, RoundKey); // добавление циклового ключа
}
```

Как можно заметить, последний цикл отличается от простого цикла только отсутствием замешивания столбцов. Каждое из приведенных преобразований подробно рассмотрено далее.

Преобразование замешивания столбцов (MixColumn)

Преобразование представляет собой умножение состояния на матрицу ME при шифровании или матрицу MD при расшифровании:

$$\begin{array}{l}
 \text{ME} \quad \begin{array}{|c|c|} \hline \square & \square \\ \hline \end{array} \text{State} \\
 \text{MD} \quad \begin{array}{|c|} \hline \square \\ \hline \end{array} \text{State}
 \end{array}
 \quad
 \begin{array}{|c|c|c|c|} \hline 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \\ \hline \end{array}
 \otimes
 \begin{array}{|c|c|c|c|} \hline b1 & b5 & b9 & b13 \\ b2 & b6 & b10 & b14 \\ b3 & b7 & b11 & b15 \\ b4 & b8 & b12 & b16 \\ \hline \end{array}$$

$$b1 = (b1 * 2) \text{ XOR } (b2 * 3) \text{ XOR } (b3 * 1) \text{ XOR } (b4 * 1)$$

Умножение двух байт происходит по следующему алгоритму:

- если один из байт равен 0, результатом будет 0;
- если один из байт равен 1, результатом будет другой байт;
- в остальных случаях происходит замена каждого байта по таблице

L, заменённые байты складываются, при необходимости вычитается 255 для попадания в интервал [0, 255] и происходит замена по таблице E, что и даёт результат.

$L = \{$
, 0x00, 0x19, 0x01, 0x32, 0x02, 0x1A, 0xC6, 0x4B, 0xC7, 0x1B, 0x68, 0x33, 0xEE, 0xDF,
0x03,
0x64, 0x04, 0xE0, 0x0E, 0x34, 0x8D, 0x81, 0xEF, 0x4C, 0x71, 0x08, 0xC8, 0xF8, 0x69, 0x1C, 0xC1,
0x7D, 0xC2, 0x1D, 0xB5, 0xF9, 0xB9, 0x27, 0x6A, 0x4D, 0xE4, 0xA6, 0x72, 0x9A, 0xC9, 0x09, 0x78,
0x65, 0x2F, 0x8A, 0x05, 0x21, 0x0F, 0xE1, 0x24, 0x12, 0xF0, 0x82, 0x45, 0x35, 0x93, 0xDA, 0x8E,
0x96, 0x8F, 0xDB, 0xBD, 0x36, 0xD0, 0xCE, 0x94, 0x13, 0x5C, 0xD2, 0xF1, 0x40, 0x46, 0x83, 0x38,
0x66, 0xDD, 0xFD, 0x30, 0xBF, 0x06, 0x8B, 0x62, 0xB3, 0x25, 0xE2, 0x98, 0x22, 0x88, 0x91, 0x10,
0x7E, 0x6E, 0x48, 0xC3, 0xA3, 0xB6, 0x1E, 0x42, 0x3A, 0x6B, 0x28, 0x54, 0xFA, 0x85, 0x3D, 0xBA,
0x2B, 0x79, 0x0A, 0x15, 0x9B, 0x9F, 0x5E, 0xCA, 0x4E, 0xD4, 0xAC, 0xE5, 0xF3, 0x73, 0xA7, 0x57,
0xAF, 0x58, 0xA8, 0x50, 0xF4, 0xEA, 0xD6, 0x74, 0x4F, 0xAE, 0xE9, 0xD5, 0xE7, 0xE6, 0xAD, 0xE8,
0x2C, 0xD7, 0x75, 0x7A, 0xEB, 0x16, 0x0B, 0xF5, 0x59, 0xCB, 0x5F, 0xB0, 0x9C, 0xA9, 0x51, 0xA0,
0x7F, 0x0C, 0xF6, 0x6F, 0x17, 0xC4, 0x49, 0xEC, 0xD8, 0x43, 0x1F, 0x2D, 0xA4, 0x76, 0x7B, 0xB7,
0xCC, 0xBB, 0x3E, 0x5A, 0xFB, 0x60, 0xB1, 0x86, 0x3B, 0x52, 0xA1, 0x6C, 0xAA, 0x55, 0x29, 0x9D,
0x97, 0xB2, 0x87, 0x90, 0x61, 0xBE, 0xDC, 0xFC, 0xBC, 0x95, 0xCF, 0xCD, 0x37, 0x3F, 0x5B, 0xD1,
0x53, 0x39, 0x84, 0x3C, 0x41, 0xA2, 0x6D, 0x47, 0x14, 0x2A, 0x9E, 0x5D, 0x56, 0xF2, 0xD3, 0xAB,
0x44, 0x11, 0x92, 0xD9, 0x23, 0x20, 0x2E, 0x89, 0xB4, 0x7C, 0xB8, 0x26, 0x77, 0x99, 0xE3, 0xA5,
0x67, 0x4A, 0xED, 0xDE, 0xC5, 0x31, 0xFE, 0x18, 0x0D, 0x63, 0x8C, 0x80, 0xC0, 0xF7, 0x70, 0x07
 $\};$

$E = \{$
0x01, 0x03, 0x05, 0x0F, 0x11, 0x33, 0x55, 0xFF, 0x1A, 0x2E, 0x72, 0x96, 0xA1, 0xF8, 0x13,
0x35,
0x5F, 0xE1, 0x38, 0x48, 0xD8, 0x73, 0x95, 0xA4, 0xF7, 0x02, 0x06, 0x0A, 0x1E, 0x22, 0x66, 0xAA,
0xE5, 0x34, 0x5C, 0xE4, 0x37, 0x59, 0xEB, 0x26, 0x6A, 0xBE, 0xD9, 0x70, 0x90, 0xAB, 0xE6, 0x31,
0x53, 0xF5, 0x04, 0x0C, 0x14, 0x3C, 0x44, 0xCC, 0x4F, 0xD1, 0x68, 0xB8, 0xD3, 0x6E, 0xB2, 0xCD,
0x4C, 0xD4, 0x67, 0xA9, 0xE0, 0x3B, 0x4D, 0xD7, 0x62, 0xA6, 0xF1, 0x08, 0x18, 0x28, 0x78, 0x88,
0x83, 0x9E, 0xB9, 0xD0, 0x6B, 0xBD, 0xDC, 0x7F, 0x81, 0x98, 0xB3, 0xCE, 0x49, 0xDB, 0x76, 0x9A,
0xB5, 0xC4, 0x57, 0xF9, 0x10, 0x30, 0x50, 0xF0, 0x0B, 0x1D, 0x27, 0x69, 0xBB, 0xD6, 0x61, 0xA3,
0xFE, 0x19, 0x2B, 0x7D, 0x87, 0x92, 0xAD, 0xEC, 0x2F, 0x71, 0x93, 0xAE, 0xE9, 0x20, 0x60, 0xA0,
0xFB, 0x16, 0x3A, 0x4E, 0xD2, 0x6D, 0xB7, 0xC2, 0x5D, 0xE7, 0x32, 0x56, 0xFA, 0x15, 0x3F, 0x41,
0xC3, 0x5E, 0xE2, 0x3D, 0x47, 0xC9, 0x40, 0xC0, 0x5B, 0xED, 0x2C, 0x74, 0x9C, 0xBF, 0xDA, 0x75,
0x9F, 0xBA, 0xD5, 0x64, 0xAC, 0xEF, 0x2A, 0x7E, 0x82, 0x9D, 0xBC, 0xDF, 0x7A, 0x8E, 0x89, 0x80,
0x9B, 0xB6, 0xC1, 0x58, 0xE8, 0x23, 0x65, 0xAF, 0xEA, 0x25, 0x6F, 0xB1, 0xC8, 0x43, 0xC5, 0x54,
0xFC, 0x1F, 0x21, 0x63, 0xA5, 0xF4, 0x07, 0x09, 0x1B, 0x2D, 0x77, 0x99, 0xB0, 0xCB, 0x46, 0xCA,
0x45, 0xCF, 0x4A, 0xDE, 0x79, 0x8B, 0x86, 0x91, 0xA8, 0xE3, 0x3E, 0x42, 0xC6, 0x51, 0xF3, 0x0E,
0x12, 0x36, 0x5A, 0xEE, 0x29, 0x7B, 0x8D, 0x8C, 0x8F, 0x8A, 0x85, 0x94, 0xA7, 0xF2, 0x0D,
0x17,
0x39, 0x4B, 0xDD, 0x7C, 0x84, 0x97, 0xA2, 0xFD, 0x1C, 0x24, 0x6C, 0xB4, 0xC7, 0x52, 0xF6, 0x01
 $\};$

Добавление циклового ключа

В данной операции цикловой ключ добавляется к состоянию посредством простого EXOR. Цикловой ключ вырабатывается из ключа шифрования посредством алгоритма выработки ключей (key schedule).

Длина циклового ключа равна длине блока Nb.

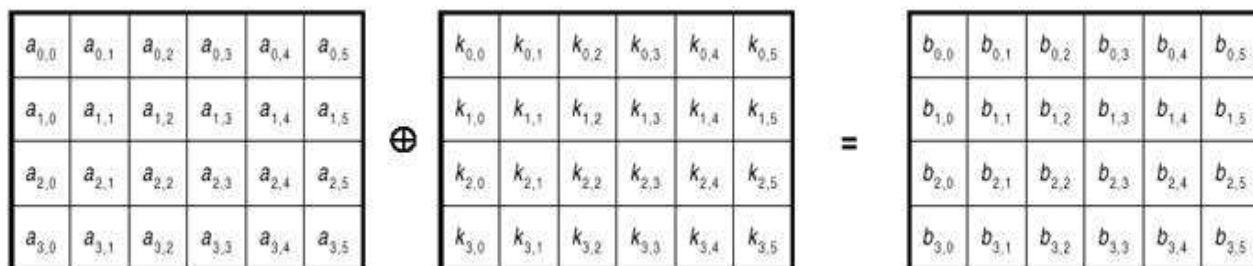


Рис. 4. 5. При добавлении ключа цикловой ключ складывается посредством EXOR с состоянием

При шифровании части расширенного ключа выбираются от начала к концу, при расшифровании – от конца к началу.

Расширение ключа (Key Expansion)

Расширенный ключ представляет собой линейный массив 4-х байтовых слов и обозначен как $W[Nb*(Nr+1)]$. Первые Nk слов содержат ключ шифрования. Все остальные слова определяются рекурсивно из слов с меньшими индексами. Алгоритм выработки ключей зависит от величины Nk : ниже приведена версия для Nk равного или меньшего 6 и версия для Nk большего 6.

Для $Nk < 6$ или $Nk = 6$ мы имеем:

```
KeyExpansion(CipherKey, W)
{
    for (i = 0; i < Nk; i++) W[i] = CipherKey[i];
    for (j = Nk; j < Nb*(Nk+1); j += Nk)
    {
        W[j] = W[j-Nk] ^ SubByte( Rotl( W[j-1] ) ) ^ Rcon[j/Nk];

        for (i = 1; i < Nk && i+j < Nb*(Nr+1); i++)
            W[i+j] = W[i+j-Nk] ^ W[i+j-1];
    }
}
```

Как можно заметить, первые Nk слов заполняются ключом шифрования. Каждое последующее слово $W[i]$ получается посредством EXOR предыдущего слова $W[i-1]$ и слова на Nk позиций ранее $W[i-Nk]$. Для слов, позиция которых кратна Nk , перед EXOR применяется преобразование к $W[i-1]$, а затем еще прибавляется цикловая константа. Преобразование содержит циклический сдвиг байтов в слове, обозначенный как $Rotl$, затем следует $SubByte$ - применение замены байт.

Для $Nk > 6$ мы имеем:

```
KeyExpansion(CipherKey, W)
{
    for (i=0; i<Nk; i++) W[i]=CipherKey[i];
    for (j=Nk; j<Nb*(Nk+1); j+=Nk)
    {
        W[j] = W[j-Nk] ^ SubByte(Rotl(W[j-1])) ^ Rcon[j/Nk];
        for (i=1; i<4; i++) W[i+j] = W[i+j-Nk] ^ W[i+j-1];
        W[j+4] = W[j+4-Nk] ^ SubByte(W[j+3]);
    }
}
```

```

for (i=5; i<Nk; i++) W[i+j] = W[i+j-Nk] ^ W[i+j-1];
}
}

```

Отличие для схемы при $Nk > 6$ состоит в применении SubByte для каждого 4-го байта из Nk .

Цикловая константа независит от Nk и определяется следующим образом:

```

Rcon[i] = ( RC[i], '00' , '00' , '00' ), где
RC[0]='01'
RC[i]=xtime(Rcon[i-1])

```

Шифрование

Шифр Rijndael включает следующие преобразования:

- начальное добавление циклового ключа;
- $Nr-1$ циклов;
- заключительный цикл.

На псевдо-Си это выглядит следующим образом:

```

Rijndael (State, CipherKey)
{
KeyExpansion(CipherKey, ExpandedKey); // Расширение ключа
AddRoundKey(State, ExpandedKey); // Добавление циклового ключа
For ( i=1 ; i<Nr ; i++) Round(State,ExpandedKey+Nb*i); // циклы
FinalRound(State, ExpandedKey+Nb*Nr); // заключительный цикл
}

```

Если предварительно выполнена процедура расширения ключа, то процедура будет выглядеть следующим образом:

```

Rijndael (State, CipherKey)
{
AddRoundKey(State, ExpandedKey);
For ( i=1 ; i<Nr ; i++) Round(State,ExpandedKey+Nb*i); FinalRound(State, ExpandedKey+Nb*Nr);
}

```

Описание демонстрационной программы Программа выполнена на языке C# и состоит из двух элементов – файла Rijndael.dll, содержащего реализацию алгоритма шифрования, и демонстрационного приложения RijndaelDemo.exe. Для работы приложения необходима ОС Windows с установленным .NET Framework v1.1. В основном окне демонстрационной программы задаются длина ключа, длина блока, можно посмотреть расширенный ключ шифрования, вычисляемый в соответствии с заданным ключом шифрования.

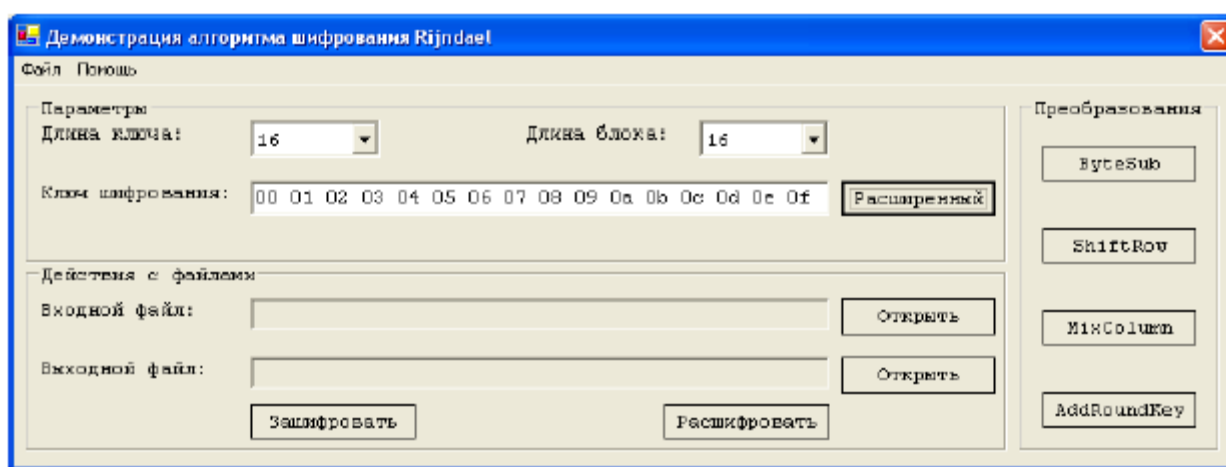


Рисунок 2. Главное окно демонстрационной программы

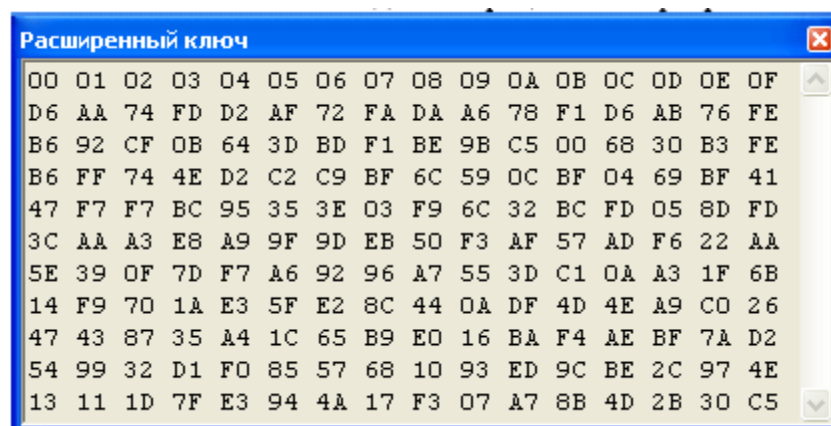


Рисунок 3. Окно расширенного ключа

Также есть возможность подробно рассмотреть действие всех цикловых преобразований (ByteSub, ShiftRow, MixColumn, AddRoundKey) как при шифровании, так и при расшифровании, рис.4, 5.

Преобразование ByteSub

Вход:

00	04	08	0c
01	05	09	0d
02	06	0a	0e
03	07	0b	0f

Выход:

63	f2	30	fe
7c	6b	01	d7
77	6f	67	ab
7b	c5	2b	76

☒ Шифрование
☐ Дешифрование

Преобразовать

Преобразование ShiftRow

Вход:

00	04	08	0c
01	05	09	0d
02	06	0a	0e
03	07	0b	0f

Выход:

00	04	08	0c
05	09	0d	01
0a	0e	02	06
0f	03	07	0b

☒ Шифрование
☐ Дешифрование

Преобразовать

Рисунок 4. Окна преобразований ByteSub и ShiftRow

Преобразование MixColumn

Вход:

00	04	08	0c
01	05	09	0d
02	06	0a	0e
03	07	0b	0f

Выход:

02	06	0a	0e
07	03	0f	0b
00	04	08	0c
05	01	0d	09

☒ Шифрование
☐ Дешифрование

Преобразовать

Преобразование AddRoundKey

Вход:

00	04	08	0c
01	05	09	0d
02	06	0a	0e
03	07	0b	0f

Ключ:

00	01	02	03
04	05	06	07
08	09	0a	0b
0c	0d	0e	0f

Выход:

00	05	0a	0f
05	00	0f	0a
0a	0f	00	05
0f	0a	05	00

Преобразовать

Рисунок 5. Окна преобразований MixColumn и AddRoundKey

При шифровании предлагается выбрать исходный файл и файл, куда будет помещен результат шифрования, при расшифровании соответственно зашифрованный файл и файл, предназначенный для помещения резуль-

тата расшифрования. В процессе используются, указанные в главном окне программы, ключ шифрования и длины ключа и блока.

4. Оборудование и материалы

Для выполнения работы потребуется:

- класс персональных ЭВМ не ниже Pentium-4, классная доска, мел;
- операционная система Microsoft Windows, программное средство *RijndaelDemo* ,

5. Указания по технике безопасности

Требования безопасности перед началом работ:

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории;
- убедиться в целостности электрических розеток и разъемов. Запрещается работать с неисправным электроинструментом, разбивать и самостоятельно ремонтировать его;
- в лаборатории необходимо быть в сменной обуви;
- включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

Требования безопасности во время работы:

- выполняя лабораторную работу, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и дискеты, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отсоединение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса.

Требования безопасности по окончании работы:

- по команде преподавателя студенты обязаны выключить компьютер и составляющие вычислительного комплекса;

– рабочее место привести в порядок.

6. Задания

Задание №1. Ознакомиться со сведениями о программе *RijndaelDemo*, изложенными в разделе 3. Запустить модуль *RijndaelDemo.exe*.

Задание №2. Изучить на примере обычных текстовых файлов способы шифрования и расшифрования с помощью алгоритма *Rijndael*. Подробно рассмотреть действие всех цикловых преобразований (ByteSub, ShiftRow, MixColumn, AddRoundKey), как при шифровании, так и расшифровании. Исходный текст для шифрования может быть подготовлен заранее и сохранен в файле *.txt.

Задание №3. Сохранить в отчете экранные формы, демонстрирующие процесс шифрования и расшифрования информации, проанализировать полученные результаты.

Задание №4. Включить в отчет о лабораторной работе ответы на контрольные вопросы, выбранные в соответствии с номером варианта, указанным преподавателем.

7. Содержание отчета

Отчет должен содержать:

1. Наименование работы, цель.
2. Последовательность действий при выполнении каждого задания.
3. Выводы по выполнению лабораторной работы.

8. Контрольные вопросы

Номер варианта	Контрольные вопросы
1,5,7,26	Сравните основные характеристики алгоритмов <i>Rijndael</i> и ГОСТ 28147-89.
2,4,6	Сравните основные характеристики алгоритмов <i>Rijndael</i> и <i>DES</i> .
11,13	Опишите структуру сети Фейстеля.
12,14,16	Приведите обобщенные схемы шифрования данных с помощью алгоритма <i>Rijndael</i> и ГОСТ 28147-89. Дайте их сравнительный анализ.
3,9,18,29	Сравните один раунд шифрования данных с помощью алгоритма <i>Rijndael</i> и ГОСТ 28147-89.
20,22,24	Сравните эквивалентность прямого и обратного преобразований в алгоритмах <i>Rijndael</i> и ГОСТ 28147-89.
10,17,19	Сравните выработку ключевой информации в алгоритмах <i>Rijndael</i> и ГОСТ 28147-89.
21,23,25	Сравните алгоритмы <i>Rijndael</i> и ГОСТ 28147-89 по показателям диффузии.
8, 28,27	Сравните алгоритмы <i>Rijndael</i> и ГОСТ 28147-89 по показателям стойкости.
12,15,30	Сравните алгоритмы <i>Rijndael</i> и ГОСТ 28147-89 по показателям производительности и удобству реализации.

9. Список литературы, рекомендованный к использованию по данной теме

1. Жук, А.П., Жук, Е.П., Лепешкин, О.М., Тимошкин, А.И. Защита информации: Учебное пособие для бакалавров [Текст] / А.П. Жук и др. – М.: ИНФРА-М, 2015. -392с.

2. Шаньгин, В.Ф. Защита компьютерной информации. Эффективные методы и средства [Текст] / В.Ф. Шаньгин – М.: ДМК Пресс, 2012. – 544 с.

ЛАБОРАТОРНАЯ РАБОТА №4

ИЗУЧЕНИЕ УСТРОЙСТВА И ПРИНЦИПА РАБОТЫ ШИФРОВАЛЬНОЙ МАШИНЫ ЭНИГМА (*Enigma*)

Цель работы: Изучение принципов шифрования/расшифрования информации, используемых в шифровальной машине Энигма. Ознакомление с общими принципами действия шифровальной машины Энигма на примере эмулятора. Предварительно необходимо установить программу эмулятор *Enigma3S*.

1. Описание лабораторной работы

Энигма, устройство и принцип действия



Рис. 3.1. Внешний вид шифровальной машины Энигма

Энигма (*Enigma*)⁴ — портативная шифровальная машина, использовавшаяся для шифрования и расшифрования секретных сообщений. Более точно, Энигма — целое семейство электромеханических роторных машин, применявшихся с 20-х годов XX века.

⁴ Эмуляторы Энигмы <http://www.attlabs.att.co.uk/andyc/enigma>

Энигма использовалась в коммерческих целях, а также в военных и государственных службах во многих странах мира, но наибольшее распространение получила в Германии во время второй мировой войны. Именно Энигма Вермахта (*Wehrmacht Enigma*) — немецкая военная модель — чаще всего является предметом изучения.

Хотя шифр Энигмы, с точки зрения криптографии, был достаточно слаб, но на практике лишь сочетание этого фактора с другими, такими, как

ераторов, процедурные изъяны и захваты экземпляров Энигмы

и шифровальных книг, позволило английским криптоаналитикам вскрывать сообщения, зашифрованные шифром Энигмы.

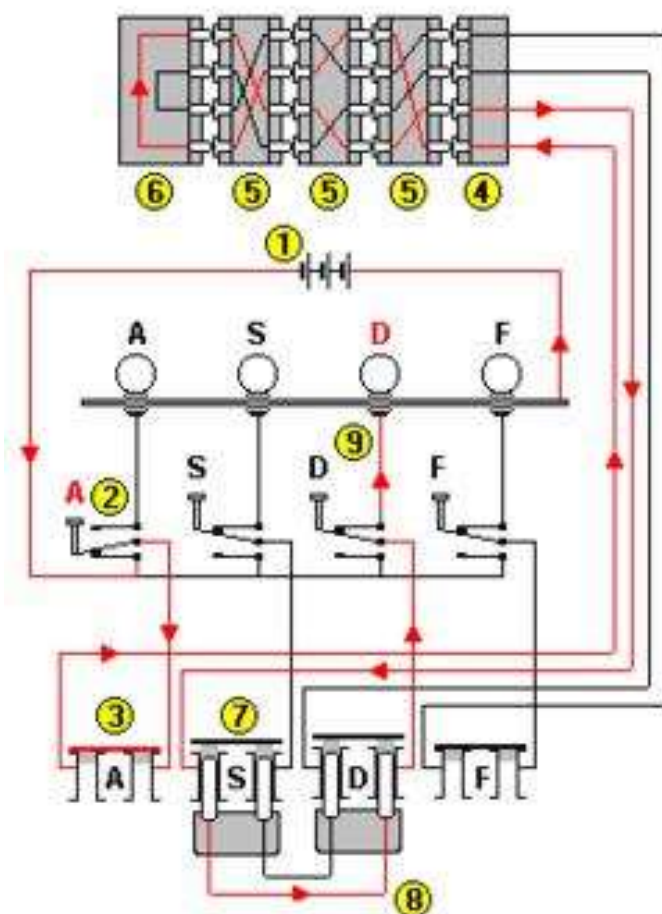


Рис. 3.2. Электрическая схема Энигмы (замена в тексте буквы 'A' буквой 'D')

Шифрующее действие Энигмы показано для двух последовательно нажатых клавиш — ток течет через роторы, "отражается" от рефлексора, затем снова возвращается через роторы. Серыми линиями на рисунке показаны другие возможные электрические цепи внутри каждого ротора. Буква "А" заменяется в шифротексте по-разному при последовательных нажатиях клавиши, сначала на "G", затем на "C". Сигнал идет по другому маршруту за счет поворота ротора.

Как и другие роторные машины, Энигма состояла из комбинации механических и электрических систем. Механическая часть включала клавиатуру, набор вращающихся дисков (роторов), расположенных вдоль вала, и ступенчатого механизма, приводящего в движение один или более роторов при каждом нажатии клавиши. Движение роторов приводит к различным вариантам подстановки символов при каждом следующем нажатии клавиши на клавиатуре.

Механические части двигались, образуя меняющийся электрический контур — то есть, фактически, шифрование осуществлялось электрически. При нажатии клавиш контур замыкался, ток проходил через различные компоненты и в итоге включал одну из множества лампочек, отображавшую выводимую букву. Например, при шифровании сообщения, начинающегося "ANX...", оператор вначале нажимал кнопку "А", и загоралась лампочка "Z", то есть "Z" становилась первой буквой криптограммы. Оператор продолжал шифрование, нажимая на клавиатуре "N" и, так далее, до конца исходного сообщения.

Постоянное изменение электрической цепи, через которую шел ток, вследствие вращения роторов, позволяло реализовать многоалфавитный шифр подстановки, что давало высокую стойкость для того времени.

Роторы



Рис. 3.3. Левая сторона ротора Энигмы, видны плоские электрические контакты



Рис. 3.4. Правая сторона ротора, видны штыревые контакты

Роторы — это сердце Энигмы. Каждый ротор представляет собой диск примерно 10 см в диаметре, сделанный из твердой резины или бакелита, с пружинными штыревыми контактами на одной стороне ротора, расположенными по окружности; на другой стороне соответствующее количество плоских электрических контактов. Штыревые и плоские контакты соответствуют буквам в алфавите; обычно это 26 букв “А”...“Z”. При соприкосновении контакты соседних роторов замыкают электрическую цепь. Внутри ротора каждый штыревой контакт соединен с некоторым плоским. Порядок соединения может быть различным.

Сам по себе ротор воспроизводит шифрование простой заменой символов. Например, контакт, отвечающий за букву “Е”, может быть соединен с контактом буквы “Т” на другой стороне ротора. Но при использовании не-

скольких роторов в связке (обычно трех или четырех), за счет их постоянного движения, получается более стойкий тип многоалфавитного шифрования.

Ротор может занимать одну из 26 позиций в машине. Он может быть повернут вручную при помощи рифленого пальцевого колесика, которое выдается наружу, как показано на Рис. 3.5. Чтобы оператор всегда мог определить положение ротора, на каждом ободе находится алфавитное кольцо; одна из букв видна через окошко. В ранних моделях Энигмы алфавитное кольцо было фиксировано; в более поздних версиях ввели усложненную конструкцию с возможностью его регулировки. Каждый ротор содержит выемку (или несколько выемок), используемых для управления движением роторов.

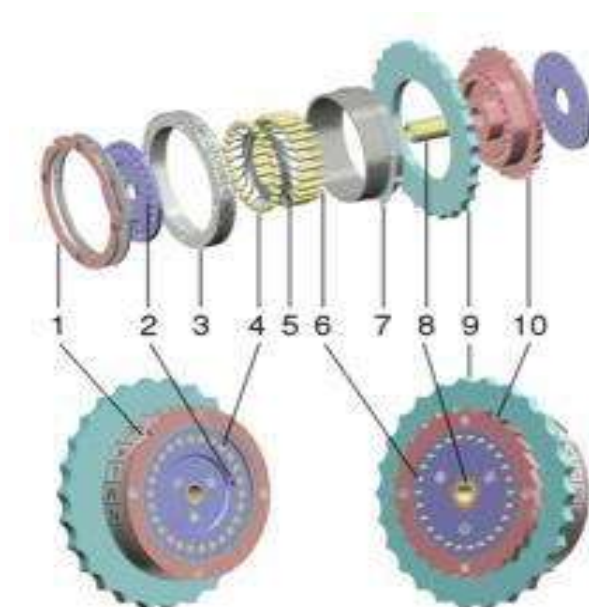


Рис. 3.5. Ротор в разобранном виде

1. кольцо с выемками
2. маркирующая точка для контакта "А"
3. алфавитное кольцо
4. залуженные контакты
5. электропроводка
6. штыревые контакты
7. пружинный рычаг для настройки кольца
8. втулка
9. пальцевое кольцо
10. храповое колесо



Рис. 3.6. Три последовательно соединенных ротора

Военные версии Энигмы выпускались с несколькими роторами; первая модель содержала только три. В 1938 г. их стало пять, но только три из них одновременно использовались в машине. Эти типы роторов были маркированы римскими цифрами I, II, III, IV, V, и все с одной выемкой, расположенной в разных местах алфавитного кольца. В военно-морских версиях *Wehrmacht Enigma* содержалось большее количество роторов, чем в других: шесть, семь или восемь.

В *Wehrmacht Enigma* каждый ротор прикреплен к регулируемому кольцу с выемками. Пять базовых роторов (I-V) имели по одной выемке,

тогда как военно-морские с дополнительными роторами (VI-VIII) — по две. В определенный момент, выемка попадает напротив собачки, позволяя ей зацепить храповик следующего ротора при последующем нажатии клавиши. Когда же собачка не попадает в выемку, она просто проскальзывает по поверхности кольца, не цепляя шестеренки. В системе с одной выемкой второй ротор продвигается вперед на одну позицию в то время, как первый продвигается на 26. Аналогично, третий ротор продвигается на один шаг в то время, как второй делает 26 шагов. Особенностью было то, что второй ротор также поворачивался, если поворачивался третий; это означает, что второй ротор мог повернуться дважды при двух последовательных нажатиях клавиш, так называемое "двухшаговое движение", что приводит к уменьшению периода при шифровании.

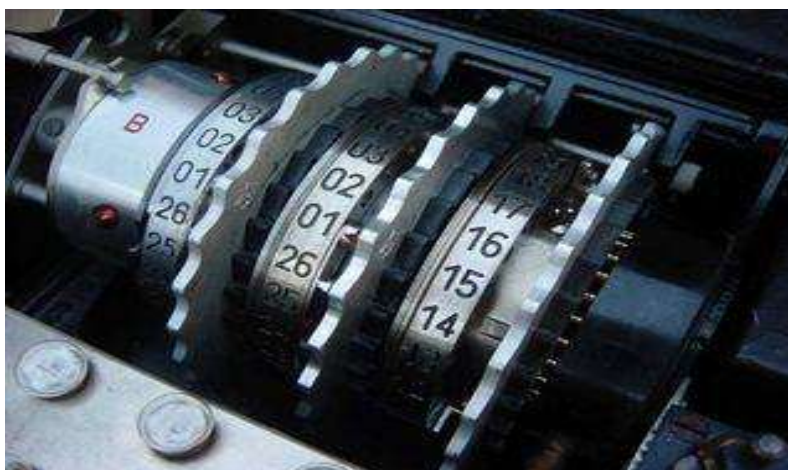


Рис. 3.7. Роторы Энигмы в собранном состоянии. Три подвижных ротора помещены между двумя неподвижными деталями: входное кольцо и рефлексор (помечен "В" слева)

Входное колесо

Входное колесо (*Eintrittswalze* по-немецки), или входной статор, соединяет коммутационную панель, или (в случае ее отсутствия) клавиатуру и ламповую панель, с роторами. Несмотря на то, что фиксированное соединение проводов имеет сравнительно небольшое значение с точки зрения безопасности, это оказалось некоторым препятствием в работе польского криптоаналитика Марьяна Реджевски, когда он пытался определить способ коммутации проводов внутри роторов. Коммерческая версия Энигмы соединяла буквы в порядке их следования на клавиатуре: QA, WB, EC и так далее. Однако, военная версия соединяла их в прямом алфавитном порядке: AA, BB, CC и т.д.

Рефлексор

За исключением ранних моделей А и В, за последним ротором следовал рефлексор (*Umkehrwalze* по-немецки), запатентованная деталь, отличавшая семейство Энигмы от других роторных машин, разработанных в то время. Рефлексор соединяет контакты последнего ротора попарно, коммутируя ток через роторы в обратном направлении, но по другому маршруту. Рефлексор гарантирует, что преобразование, реализуемое Энигмой, есть инволю-

ция, т.е. процесс расшифрования симметричен процессу шифрования. Кроме того, рефлектор придает Энигме то свойство, что никакая буква не может быть зашифрована собой же. Это

было серьезным концептуальным недостатком, впоследствии использованным дешифровальщиками.

Коммутационная панель



Рис. 3.8. Коммутационная панель в передней части машины. Могло использоваться до 13 соединений. На фотографии две пары букв переключены (“S”-“O” и “J”-“A”)

Коммутационная панель, позволяющая оператору варьировать соединения проводов, впервые появилась в немецких военных версиях в 1930 г. и вскоре успешно использовалась и в военно-морских версиях. Коммутационная панель внесла огромный вклад в усложнение шифра Энигмы, даже больший, чем введение дополнительного ротора. С Энигмой без коммутационной панели — *"unsteckered"* Энигмой — дешифровальщик может справиться практически вручную, однако эти методы оказывались беспомощными при добавлении коммутационной панели, и взломщики были вынуждены конструировать специальные дешифровальные машины. Кабель, помещенный на коммутационную панель, соединяет буквы попарно, например, “E” и “Q” могут быть соединены в пару. Эффект состоит в перестановке этих букв до и после прохождения сигнала через роторы. Например, когда оператор нажимал “E”, сигнал направлялся в “Q”, и только после этого уже во входной ротор. Одновременно могло использоваться несколько таких пар (до 13).

Аксессуары

Удобной деталью, использовавшейся на М4 Энигма, был "*Schreibmax*", маленькое печатающее устройство, которое могло печатать все 26 букв на небольшом листе бумаги.

Процедуры использования Энигмы

Во время второй мировой войны немецкие операторы использовали шифровальную книгу только для установки роторов и настроек колец. Для каждого сообщения, они выбирали случайную стартовую позицию, допустим "WZA", и случайный ключ сообщения, допустим "SXT". Далее оператор устанавливал роторы в стартовую позицию "WZA", и шифровал ключ сообщения "SXT". Предположим, что в результате шифрования ключа получится "UHL". Далее оператор ставил ключ сообщения "SXT", как начальную позицию роторов, и шифровал сообщение. Затем оператор отправлял стартовую позицию "WZA" и зашифрованный ключ "UHL" вместе с сообщением. Получатель устанавливал стартовую позицию в соответствии с первой триграммой "WZA" и, расшифровывая вторую триграмму "UHL" Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Распознавал исходный ключ "SXT". Далее, получатель использовал этот ключ как стартовую позицию для расшифровки сообщения. Обычно срок действия ключей составлял один день.

Военная версия Энигмы использовала только 26 букв. Прочие символы заменялись редкими комбинациями букв. Пробел пропускался либо заменялся "X". Символ "X" также использовался для обозначения точки либо конца сообщения. Некоторые особые символы использовались

в отдельных вооруженных частях, например, *Wehrmacht* заменяла запятую двумя символами "ZZ" и вопросительный знак - "FRAGE" либо "FRAQ".

А *Kriegsmarine* заменяла запятую - “Y” и вопросительный знак - “UD”. Два, три или четыре нуля заменялись CENTA, MILLE и MYRIA соответственно.

При выполнении лабораторной работы для исследования шифра Энигмы используется программа-эмулятор **Enigma3S**.

2. Задание

1. Запустите эмулятор Энигмы **Enigma3S** из папки, указанной преподавателем. Ознакомьтесь с файлом справки: опция меню

Help=>Help. В меню программы выберите пункт Settings => Reset, рис.3.9.

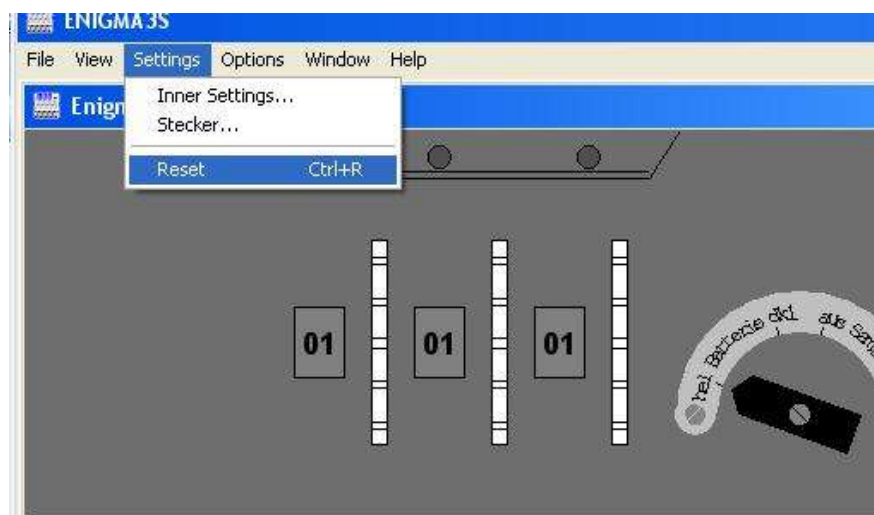


Рис. 3.9. Начальные установки эмулятора Энигмы

3. Установите значения для колец 01 01 01 путем выбора пункта меню

View=>Open Cover. В меню Settings=>Inner Settings установите следующие значения Reflector – B, Left – I, Middle – II, Right – III, Ringstellung – A-A-A (Будем считать данное положение начальным),

рис.3.10, 3.11.

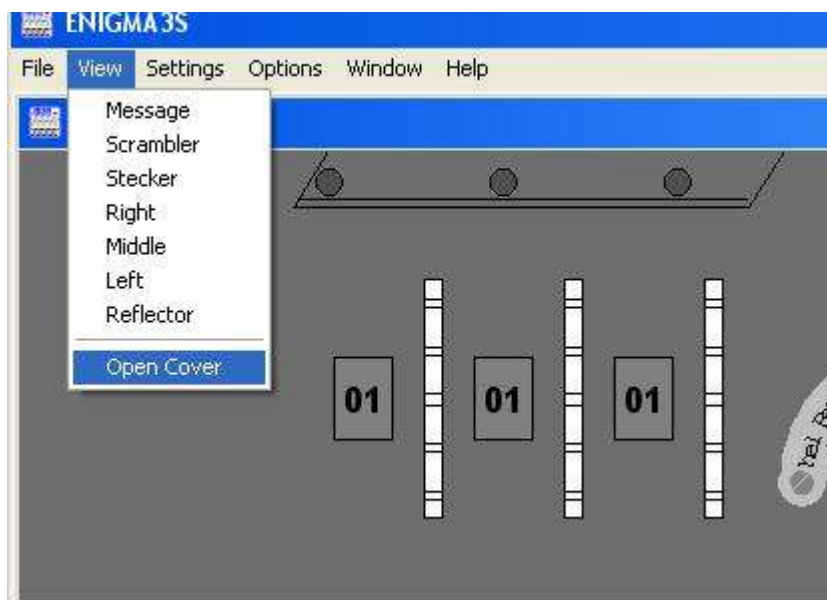


Рис. 3.10. Просмотр начальных установок

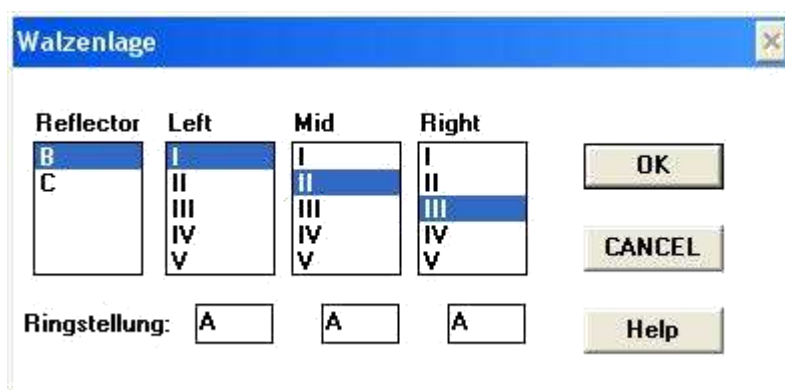


Рис. 3.11. Установки Reflector, Left, Mid, Right

4. Введите на клавиатуре Энигмы сообщение “SECRET MESSAGE”.
Какое сообщение получено на выходе?
5. Повторите пункт 3, изменив настройки Ringstellung – A-A-A на
Ringstellung – A-B-C
6. Введите на клавиатуре Энигмы сообщение “SECRET MESSAGE”.
Какое сообщение получено на выходе? Насколько оно отличается от сообщения полученного в пункте 4?
7. Сохраните полученный в пункте 6 шифротекст при помощи опции
меню File=>Save CT as, рис. 3.12.



Рис. 3.12. Сохранение шифротекста

8. Создайте в корне папки с программой эмулятором файл с расширением .pln, откройте его для редактирования в блокноте и запишите в него открытый текст для шифрования.

9. В меню программы выберите опцию File=>Open Pt и выберите файл, созданный в пункте 9. Получите шифротекст из открытого текста, выбрав опцию меню Options=>Encipher Text.

10. Установите эмулятор в начальное положение. В меню программы выберите пункт Window=> Scrambler. Введите при помощи клавиатуры произвольное сообщение из 22 символов (варианты указаны в *таблице 3.1*), обращая внимание на положение колец. Введите последовательность из 22 символов еще раз. Как изменилось положение колец? Отличается ли новая зашифрованная последовательность от начальной? Почему изменилась выходная последовательность?

11. Используя окно Scrambler проследите за тем как изменяется шифротекст в зависимости от настройки положения контактных колес. Что дает возможность настройки порядка следования контактных колес?

В окне программы выберите опцию View=>Right, рис. 3.13.

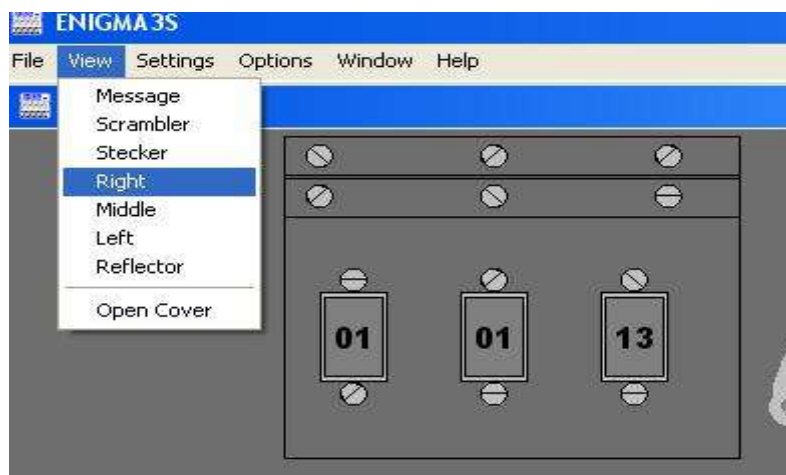


Рис. 3.13. Просмотр установок для выполнения задания

Заполните таблицу соответствия для 5-ти букв варианты указаны в таблице 3.1) для 12 первых угловых положений правого колеса.

Таблица 3.1

Буквы на входе	Угловые положения колеса											
	11	22	33	44	55	66	77	88	99	110	111	112

13. Зная, что эмулятор установлен в начальное положение расшифруйте текст в соответствии с номером варианта, указанным преподавателем (варианты контрольных заданий указаны в *таблицах 3.2,3.3,3.4*).

Таблица 3.2

Номера вариантов	Исходный алфавит
1,5,9,13,17	QRTYU FBNAK GHERL ADLKE DS
2,6,10,14,18,22	UIERT PAEVC DSNCY OPLKD BV
3,7,11,15,19,23,27	UIFGH KLBVQ FDIIT QKJLS DB
4,8,12,16,20,24,28	LLWER TYYWV BAFDP WRTOPF JK
21,25,29,26,30	OPJHG JFDPJ GFDSK LDFHU BX

Таблица 3.3

Номера вариантов	Шифротекст
1,5,9,13,17	ABCDE
2,6,10,14,18,22	FGHIJK
3,7,11,15,19,23,27	LMNOP
4,8,12,16,20,24,28	QRSTU
21,25,29,26,30	VWXYZ

Таблица 3.4

Номера вариантов	Шифротекст
1,5,9,13,17	FQGAH WABUN NL
2,6,10,14,18,22	QIKOL RCRJS EGBSS X
3,7,11,15,19,23,27	OOKWE PRFMI M
4,8,12,16,20,24,28	KIXDI ACTHJ L
21,25,29,26,30	XLXOO EABUN NL

Лабораторная работа №5

Генерация простых чисел, используемых в асимметричных системах шифрования

Цель работы: исследования генерации простых чисел, используемых в системах шифрования с открытым ключом и умений делать самостоятельные выводы по результатам исследований.

2. Формируемые компетенции

Данная лабораторная работа направлена на формирование следующих компетенций выпускника бакалавриата:

ОПК-1: способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ПК-5: способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.

3. Теоретическая часть

Асимметричные системы шифрования

Асимметричные криптосистемы (системы открытого шифрования, с открытым ключом - public key systems) – смысл данных криптосистем состоит в том, что для зашифрования и расшифрования используются разные преобразования. Одно из них – зашифрование – является абсолютно открытым для всех. Другое же – расшифрование – остается секретным за счет секретности ключа расшифрования. Таким образом, любой, кто хочет что-либо зашифровать, пользуется открытым преобразованием. Но расшифровать и прочитать это сможет лишь тот, кто владеет секретным ключом. Схема асимметричной криптосистемы представлена на рис.1.1.

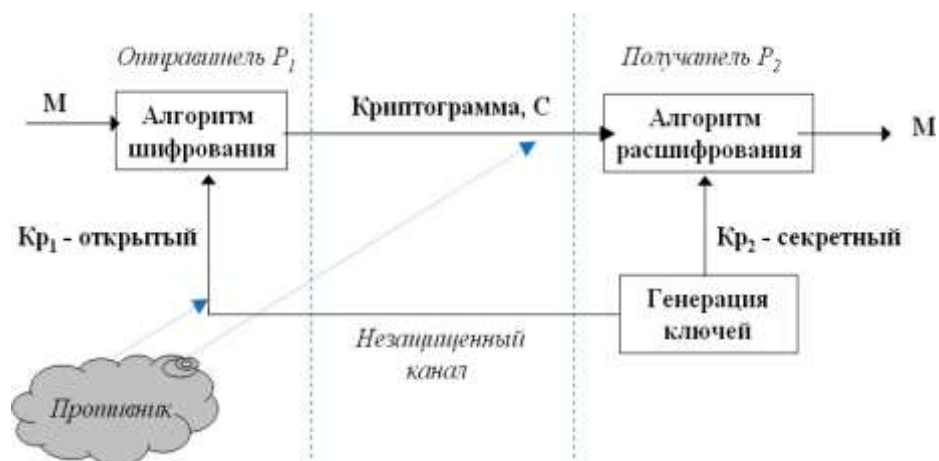


Рис. 1.1. Обобщенная схема асимметричной криптосистемы

В настоящий момент во многих асимметричных криптосистемах вид преобразования определяется ключом. У пользователя есть два ключа – секретный и открытый. Открытый ключ публикуется в общедоступном месте, и каждый, кто захочет послать сообщение этому пользователю – зашифровывает текст открытым ключом. Расшифровать сможет только упомянутый пользователь с секретным ключом. Таким образом, отпадает проблема передачи секретного ключа, как в симметричных системах. Однако, несмотря на все свои преимущества, эти криптосистемы достаточно трудоемки и медлительны. Стойкость асимметричных криптосистем базируется, в основном, на алгоритмической трудности решить за приемлемое время какую-либо задачу. Если злоумышленнику удастся построить такой алгоритм, то дискредитирована будет вся система и все сообщения, зашифрованные с помощью этой системы. В этом состоит главная опасность асимметричных криптосистем в отличие от симметричных.

Алгоритм Диффи-Хеллмана

Алгоритм Диффи-Хеллмана (*Diffie-Hellman*) использует функцию дискретного возведения в степень. Сначала генерируются два больших простых числа p и q . Эти два числа не обязательно хранить в секрете. Далее один из

партнеров P_1 генерирует случайное число x и посылает другому участнику будущих обменов P_2 значение

$$A = q^x \bmod n$$

По получении A партнер P_2 генерирует случайное число y и посылает участнику обмена P_1 вычисленное значение

$$B = q^y \bmod n$$

Партнер P_1 , получив B , вычисляет $K_x = B^x \bmod n$, а партнер P_2 вычисляет $K_y = A^y \bmod n$. Алгоритм гарантирует, что числа K_y и K_x равны и могут быть использованы в качестве секретного ключа для шифрования. Ведь даже перехватив числа A и B , трудно вычислить K_x или K_y . Схематично, работа алгоритма Диффи-Хеллмана представлена на рис. 1.2.

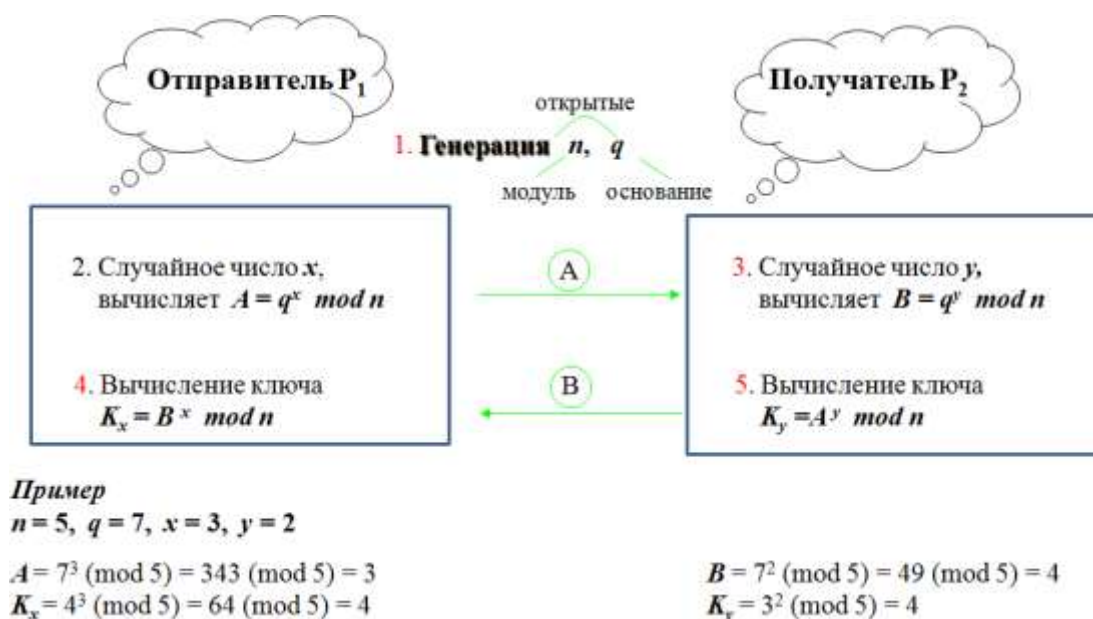


Рис. 1.2. Алгоритм Диффи-Хеллмана

Алгоритм RSA

Первое практическое воплощение принцип открытого шифрования получил в системе *RSA*, разработанной в 1977 году в Массачусетском Технологическом Институте (США) и получившей свое название от первых букв фамилий авторов: Рональд Ривест (*R.Rivest*), Эди Шамир (*A.Shamir*), Леонард Адлеман (*L.Adleman*).

Идея авторов состояла в том, что взяв целое число N в виде произведения двух больших простых чисел $N=P*Q$, легко подобрать пару чисел Y и X , таких, чтобы для любого целого числа M , меньшего N , справедливо соотношение:

$$(M^X)^Y = M \bmod N$$

В качестве открытого ключа шифрования в системе *RSA* выступают ключ Y и модуль N , а секретным ключом для расшифрования сообщений является число X .

Процедура шифрования сообщения M , рассматриваемого как целое число (такое допущение возможно вследствие того, что любой контент может быть представлен в числовой форме при обработке в средствах вычислительной техники), меньшее N (при необходимости длинное сообщение разбивается на отрезки, шифруемые независимо), состоит в вычислении значения:

$$C = M^Y \bmod N$$

Расшифрование осуществляется аналогично с использованием секретного ключа X :

$$M = C^X \bmod N$$

Математически строго можно доказать, что определение по паре чисел (N, Y) секретного ключа X , не проще разложения на простые множители числа N , то есть нахождения P и Q . Задача же разложения на множители целого числа изучается в математике с древнейших времен и известна как сложная вычислительная задача. На настоящий момент разложение числа из нескольких сотен десятичных знаков потребует от современных вычислительных машин сотен лет непрерывной работы.

Схематично, работа алгоритма *RSA* представлена на рис. 1.3.

Генерация ключей

Получатель 1. P, Q - простые, $N = P \cdot Q$
2. $\varphi(N) = (P-1) \cdot (Q-1)$, $\varphi(N)$ - функция Эйлера

Выбор открытого ключа Y :

$$1 < Y \leq \varphi(N), \text{НОД}(Y, \varphi(N)) = 1$$

Вычисление секретного ключа X :

$$X \cdot Y \equiv 1 \pmod{\varphi(N)}$$

$(N, Y) \rightarrow \text{отправителю}$

Отправитель шифрование M ($M_1 = 0, 1, 2, \dots, N-1$)
3. $C_1 = M_1^Y \pmod{N}$

Получатель расшифрование C ($C_1, C_2, \dots, C_p, \dots$)
4. $M_1 = C_1^X \pmod{N}$

Пример

Генерация ключей

1. $P = 3, Q = 11, N = P \cdot Q = 33$
2. $\varphi(N) = (P-1) \cdot (Q-1) = 2 \cdot 10 = 20$
 $Y = 7, \text{НОД}(Y, \varphi(N)) = 1$
 $X \cdot Y \equiv 1 \pmod{20}, 7 \cdot 3 \equiv 1 \pmod{20}, X = 3$

Сообщение: $M_1 M_2 \rightarrow 32; M_1 = 3 < 33, M_2 = 2 < 33$

Шифрование $C_1 = M_1^Y \pmod{N}$

3. $C_1 = 3^7 \pmod{33} = 2187 \pmod{33} = 9$
 $C_2 = 2^7 \pmod{33} = 128 \pmod{33} = 29$

Шифротекст 9,29

Расшифрование $M_1 = C_1^X \pmod{N}$

4. $M_1 = 9^3 \pmod{33} = 729 \pmod{33} = 3$
 $M_2 = 29^3 \pmod{33} = 24389 \pmod{33} = 2$

Восстановленный текст 3,2

Рис. 1.3. Алгоритм RSA

1.1. Методы проверки чисел на простоту

Одна из главных проблем асимметричного шифрования – генерация больших простых чисел. Простейшим методом проверки простоты натурального числа N является метод пробных делений: для $d=2, 3, 4 \dots$ мы проверяем выполнение условия $(d, N) > 1$ (здесь (d, N) – наибольший общий делитель чисел d и N). Число операций, требуемых для этого метода, имеет порядок корня из N .

Поэтому уже для чисел порядка $10^{30} - 10^{40}$ он не применим.

В отличие от таких “детерминированных” тестов существуют еще “вероятностные” тесты проверки простоты. Для исследуемого числа прове-

ряется выполнение некоторых, связанных со случайными числами, условий. Если какое-либо из этих условий не выполнено, то N – составное

число. Если же все условия выполнены, то с некоторой вероятностью можно утверждать, что N – простое число. Эта вероятность тем ближе к 1, чем большее количество случайных чисел мы проверим.

Обычно эти условия основаны на малой теореме Ферма, утверждающей, что для любого положительного числа b , не превосходящего некоторого простого числа p :

$$b^{(p-1)} = 1(mod\ p).$$

Например, $2^6 = 64 = 63+1 = 1(mod\ 7)$. Если требуется определить, является ли целое число r простым, то можно выбрать любое положительное целое число b , меньшее r , и проверить, выполнено ли равенство

$$b^{(r-1)} = 1(mod\ r).$$

Если равенство не выполнено, то на основании теоремы Ферма можно быть совершенно уверенным, что r – не простое число. Если же равенство выполнено, то можно лишь предполагать, что r – простое число и поэтому назвать его “псевдопростым по основанию b “. Вероятность $P(x)$ того, что составное число x окажется псевдопростым по случайному основанию, убывает с ростом x .

К сожалению, существуют так называемые числа Кармайкла – такие составные числа, которые обладают свойством:

$b^{(r-1)} = 1(mod\ r)$ для всех b из интервала $[1, r]$, которые взаимно просты с r .

Примером числа Кармайкла является число $561 = 3*11*17$.

Классический результат теории чисел – теорема Чебышева – показывает, что доля положительных целых чисел, меньших некоторого целого m и являющихся простыми, близка к $1/(\ln m)$. Например, доля целых чисел, меньших 10^{100} и являющихся простыми, близка к $1/(\ln 10^{100}) = 1/230$.

Таким образом, если мы выберем случайно большое целое положительное нечетное число x и будем последовательно проверять на простоту числа $x, x+1, x+2, \dots$, то, в среднем, мы впервые встретим простое число на шаге с номером $\ln x$.

4. Оборудование и материалы

Для выполнения работы потребуется:

- класс персональных ЭВМ не ниже Pentium-4, классная доска, мел;
- операционная система Microsoft Windows, программное средство *L_PROST.EXE*.

5. Указания по технике безопасности

Требования безопасности перед началом работ:

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории;
- убедиться в целостности электрических розеток и разъемов. Запрещается работать с неисправным электроинструментом, разбивать и самостоятельно ремонтировать его;
- в лаборатории необходимо быть в сменной обуви;
- включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

Требования безопасности во время работы:

- выполняя лабораторную работу, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и дискеты, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отсоединение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса.

Требования безопасности по окончании работы:

- по команде преподавателя студенты обязаны выключить компьютер и составляющие вычислительного комплекса;

- рабочее место привести в порядок.

Описание лабораторной работы

Для выполнения лабораторной работы необходимо запустить программу *L_PROST.EXE*. На экране дисплея появляется окно, с размещенным в его центре текстовым редактором (для отображения информации об осуществленных программой действиях), в верхней строке окна расположено главное меню, чуть ниже основного меню расположена панель инструментов (для управления быстрыми командными кнопками и другими “горячими” элементами управления), а в самом низу окна, под текстовым редактором, находится строка состояния, в которой указывается подсказка и выводится дополнительная информация. Клавиши панели инструментов, для удобства, снабжены всплывающими подсказками.

Для того чтобы попасть в основное меню, необходимо нажать клавишу F10. Передвижение по главному меню осуществляется клавишами перемещения курсора. Чтобы вызвать пункт меню, нужно нажать клавишу “ENTER”, “ESC” – выход из основного меню.

Для удобства в программе предусмотрена работа с мышью. В этом случае указатель подводится к нужному пункту главного меню или к нужной кнопке на панели инструментов и нажимается левая клавиша мыши, для отказа достаточно нажать клавишу “ESC”.

Кроме основных функций главного меню (“Генерация простого Р”, “Поиск в интервале”, “Проверка на простоту” и так далее) панель инструментов содержит клавишу, при нажатии которой выводится информация о программе.

Генерация простого Р

Возможность генерации простого числа; количество разрядов генерируемого числа задается пользователем (от 1 до 5).

Поиск в интервале

Возможность поиска простых чисел в заданном интервале. Пользователем задается: начало интервала – значение x , затем длина интервала – значение L . Поиск будет осуществляться в интервале $(x, x+L)$.

При проверке на простоту каждого числа интервала сначала выполняется тест пробных делений на первые по порядку простые числа, а затем проверка по тесту Ферма. Для задания способов проверки на простоту необходимо левой клавишей мыши отметить флажок рядом с названием нужного метода, а затем указать все необходимые данные для начала поиска.

В методе пробных делений исходные данные – количество первых простых чисел для деления, а в тесте Ферма надо указать количество оснований и их значения.

По окончании расчета на экран выведутся найденные простые числа и их количество. Полную картину результатов работы можно просмотреть в пункте меню “Вывод результатов”.

Проверка на простоту

Возможность проверки на простоту любого числа. Необходимо ввести число и параметры расчета аналогично поиску в интервале.

Вывод результатов

Возможность просмотра всех результатов последней обработки. При входе в программу, когда никаких расчетов еще не производилось, предоставляется возможность просмотра исходного файла первых простых чисел, используемых для теста пробных делений.

Дополнительные сведения о программе

1. При запуске утилит генерации простого числа, поиска в интервале и проверки на простоту у пользователя запрашивается подтверждение на правильность выбранного метода для работы.

2. Во время работы длительных по исполнению процедур запускается прогресс процесса и гасится окно текстового редактора. По полоске прогресса можно наблюдать и оценивать примерную скорость работы алгоритма и время окончания текущего процесса.

3. Будьте внимательны при установке параметров работы, так как в процессе вычисления по ходу работы эти параметры изменить уже не удастся.

4. Описание "горячих клавиш":

Ctrl+F1 - 'Генерация простого Р'

Ctrl+F2 - 'Поиск в интервале'

Ctrl+F3 - 'Проверка на простоту'

Ctrl+F4 - 'Вывод результатов'

Ctrl+X - 'Выход из программы'

5. В лабораторной работе из-за большого времени счета рекомендуется использовать числа не более пяти разрядов и длину интервала выбирать не более 500, количество оснований для теста Ферма – не более 5.

6. Для правильного функционирования программы, в рабочей директории (вместе с файлом *l_prost.exe*) обязательно должны находиться файлы *prost.txt* и *work.txt*. Не рекомендуется вносить какие-либо изменения в эти текстовые файлы иначе последствия могут быть непредсказуемые.

6. Задания

Задание №1.

1. Проверить на простоту два произвольных целых числа, разрядностью не менее 5.

2. Распределение простых чисел.

2.1. Задан интервал вида $[x, x+L]$. Вычислить количество $\Pi(x, L)$ простых чисел в интервале и сравнить с величиной $L/\ln(x)$. При каких условиях $\Pi(x, L)/L$ близко к $1/\ln(x)$?

$x=2000, L=500,$

- количество простых чисел для деления: 5-15;

- количество оснований: 1-2.

2.2. Найти в заданном интервале все простые числа. Пусть $L(i)$ – разность между двумя соседними простыми числами. Построить гистограмму для $L(i)$. Вычислить выборочное среднее $L_{\text{сред.}}$. Сравнить с величиной $\ln(x)$, где x – середина интервала.

Интервал: (1000, 1000+300):

- количество простых чисел для деления: 5-20;
- количество оснований: 1-3.

2.3. Для заданного набора чисел $\{k\}$ оценить относительную погрешность формулы для k -го простого числа:

$$p(k) = k / \ln(k), k = \{10, 15, 20, 30, 35\}.$$

3. Методы генерации простых чисел.

3.1. В заданном интервале построить график относительного количества натуральных чисел, проходящих “решето Эратосфена” (т. е. не делящихся на первые k простых).

Интервал: (500, 500+200).

Расчет производится для всех $k \leq 10$.

3.2. Для заданного интервала:

а) рассчитать точное количество P_0 простых чисел в интервале, т.е. при проверке задать только тест на делимость.

Количество первых простых чисел для деления определяется из расчета: максимальное число для деления равно квадратному корню из максимального значения интервала.

б) составить тест с небольшим количеством пробных делений и одним основанием в тесте Ферма. Вычислить количество P_1 вероятно простых чисел, удовлетворяющих этому тесту.

в) составить тест с большим, чем в предыдущем случае, количеством пробных делений и двумя или тремя основаниями в тесте Ферма. Вычислить количество P_2 вероятно простых чисел, удовлетворяющих этому тесту.

Интервал: (1500, 1500 + 300).

Проанализировать полученные данные.

3.3. Известно, что в заданном интервале имеются числа Кармайкла. Найти их. Варианты интервалов: $(1050, 1050 + 100)$ $(1700, 1700 + 100)$ $(2400, 2400 + 100)$

4. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

7. Содержание отчета

Отчет должен содержать:

1. Наименование работы, цель.
2. Последовательность действий при выполнении каждого задания.
5. Выводы по выполнению лабораторной работы.

8. Контрольные вопросы

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Почему в качестве первого основания в тестах типа теста Ферма для проверки на простоту очень больших чисел целесообразно использовать число 2?
2,4,6,8, 20,22,24, 26,30	Какова вероятность $P(x)$ того, что наугад взятое нечетное очень большое число, не превосходящее x , окажется простым?
11,13,15, 10,17,19, 27	Вычислить: $1812 \pmod{13}$, $127 \pmod{7}$.
12,14,16 21,23,25, 29	Сформулируйте суть теста на простоту с использованием пробных делений.

9. Список литературы, рекомендованный к использованию по данной теме

1. Жук, А.П., Жук, Е.П., Лепешкин, О.М., Тимошкин, А.И. Защита информации: Учебное пособие для бакалавров [Текст] / А.П. Жук и др. – М.: ИНФРА-М, 2015. -392с.

2. Шаньгин, В.Ф. Защита компьютерной информации. Эффективные методы и средства [Текст] / В.Ф. Шаньгин – М.: ДМК Пресс, 2012. – 544 с.

Лабораторная работа № 6

Электронная цифровая подпись

Цель работы: исследования принципов защищенного электронного документооборота в телекоммуникационных сетях, алгоритмов постановки электронной цифровой подписи (ЭЦП) и умений делать самостоятельные выводы по результатам исследований.

2. Формируемые компетенции

Данная лабораторная работа направлена на формирование следующих компетенций выпускника бакалавриата:

ОПК-1: способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ПК-5: способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.

3. Теоретическая часть

При обмене электронными документами по сети связи существенно снижаются затраты на обработку и хранение документов, ускоряется их поиск. Но при этом возникает проблема аутентификации автора документа и самого документа, т.е. установления подлинности автора и отсутствия изменений в полученном документе. При обработке документов в электронной форме совершенно непригодны традиционные способы установления подлинности по рукописной подписи и оттиску печати на бумажном документе. Принципиально новым решением является *электронная цифровая подпись (ЭЦП)*. Первая схема ЭЦП - RSA - была разработана еще в конце 1970-х годов. Однако проблема подтверждения авторства стала актуаль-

ной настолько, что потребовалось установление стандарта, только в 1990-х годах, во время взрывного роста глобальной сети Интернет и массового распространения электронной торговли и оказания услуг. Именно по указанной причине стандарты ЭЦП в России и США были приняты практически одновременно, в 1994 году. Из предложенных криптологами схем ЭЦП наиболее удачными оказались *RSA* и схема Эль-Гамала. Но первая из них была запатентована в США и ряде других стран (патент на *RSA* прекратил свое действие совсем недавно). Во второй же схеме существует большое количество ее возможных модификаций, и все их запатентовать весьма затруднительно. Именно по этой причине схема ЭЦП Эль-Гамала осталась по большей части свободной от патентов. Кроме того, эта схема имеет и определенные практические преимущества: размер блоков, которыми оперируют алгоритмы, и соответственно размер ЭЦП в ней оказались значительно меньше, чем в *RSA*, при той же самой стойкости. Именно поэтому стандарты ЭЦП России и США базируются на схеме Эль-Гамала. Законы об ЭЦП сегодня имеют уже более 60-ти государств. В этом списке значится и Россия. Закон “Об электронной цифровой подписи”, принятый в 2002 году должен оказать стимулирующее воздействие на развитие отечественной электронной коммерции, особенно если в соответствие с ним будут своевременно приведены иные нормативно-правовые акты.

- Правительство РФ финансово поддерживает осуществление федеральной целевой программы «Электронная Россия».
- Принят закон «О внесении изменения в статью 80 части первой Налогового кодекса Российской Федерации».
- Еще в январе 2001 г. правление Пенсионного фонда постановлением «О введении в системе Пенсионного фонда РФ криптографической защиты информации и электронной цифровой подписи» регламентировало

регистрацию и подключение юридических и физических лиц к системе своего электронного документооборота.

- В 2002 г. вышел приказ МНС России «Об утверждении порядка представления налоговой декларации в электронном виде по телекоммуникационным каналам связи», благодаря которому сегодня любое физическое или юридическое лицо может связаться с налоговой инспекцией, используя защищенную электронную почту.

- В 2004 г. были утверждены поправки к статьям 13 и 15 закона «О бухгалтерском учете», согласно которым бухгалтерская отчетность предприятия может вестись, храниться и предоставляться в контролирующие органы в электронном виде.

Принцип построения ЭЦП. Асимметрия ролей отправителя и получателя в схемах ЭЦП требует наличия двух тесно связанных ключей: *секретного*, или ключа подписи, и *открытого*, или ключа проверки подписи. Любая схема ЭЦП обязана определить три следующих алгоритма:

- алгоритм генерации ключевой пары для подписи и ее проверки;
- алгоритм постановки подписи;
- алгоритм проверки подписи.

Стандарты России и США очень похожи, они различаются лишь некоторыми числовыми параметрами и отдельными деталями выработки ключевой пары, вычисления и проверки подписи. Действительно, оба стандарта являются вариантами одной и той же схемы ЭЦП Эль-Гамала. ЭЦП используется для аутентификации текстов, передаваемых по телекоммуникационным каналам. Функционально она аналогична обычной рукописной подписи и обладает основными ее достоинствами:

- удостоверяет, что подписанный текст исходит от лица, поставившего подпись;

- не дает самому этому лицу возможность отказаться от обязательств, связанных с подписанным текстом;

- гарантирует целостность подписанного текста.

ЭЦП представляет собой относительно небольшое количество дополнительной цифровой информации, передаваемой вместе с подписываемым текстом, и включает две процедуры:

- процедуру постановки подписи, в которой используется секретный ключ отправителя сообщения;

- процедуру проверки подписи, в которой используется открытый ключ отправителя.

Процедура постановки подписи. При формировании ЭЦП, отправитель, прежде всего, вычисляет хэш-функцию $h(M)$ подписываемого текста M . Вычисленные значения хэш-функции $h(M)$ представляет собой один короткий блок информации t , характеризующий весь текст M в целом. Затем значение t шифруется секретным ключом отправителя. Получаемая при этом пара чисел представляет собой ЭЦП для данного текста M .

Процедура проверки подписи. При проверке ЭЦП, получатель сообщения снова вычисляет хэш-функцию $t = h(M)$ принятого по каналу текста M , после чего при помощи открытого ключа отправителя проверяет, соответствует ли полученная подпись вычисленному значению t хэш-функции.

Принципиальным моментом в системе ЭЦП является невозможность подделки ЭЦП пользователя без знания его секретного ключа.

Каждая подпись, как правило, содержит следующую информацию:

- дату подписи;
- срок окончания действия ключа данной подписи;
- информацию о лице, подписавшем текст;
- идентификатор подписавшего (имя открытого ключа);

- собственно цифровую подпись.

Однонаправленные хэш-функции. Хэш-функция предназначена для сжатия подписываемого документа M до нескольких десятков или сотен бит. Хэш-функция $h(.)$ использует в качестве аргумента сообщение M произвольной длины и возвращает хэш-значение $h(M)=H$ фиксированной длины. Обычно хэшированная информация является сжатым двоичным представлением основного сообщения произвольной длины. Следует отметить, что значение хэш-функции $h(M)$ сложным образом зависит от документа M и не позволяет восстановить сам документ M . Хэш-функция должна удовлетворять целому ряду условий:

- должна быть чувствительна к всевозможным изменениям в тексте M ;
- должна обладать свойством необратимости, т.е. задача подбора документа $M1$, который обладал бы требуемым значением хэш-функции, должна быть вычислительно неразрешима;
- вероятность того, что значения хэш-функции двух различных документов совпадут, должна быть ничтожно мала. Большинство хэш-функций строится на основе однонаправленной функции $f(.)$, которая образует выходное значение длиной n при задании двух входных значений длиной n . Этими входами являются блок исходного текста M_i и хэш-значение H_{i-1} предыдущего блока текста, рис.1.

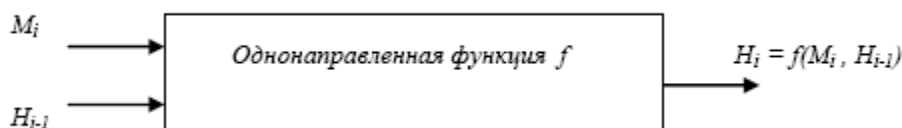


Рисунок 1. Формирование хэш-функции

Алгоритм цифровой подписи DSA. Алгоритм цифровой подписи DSA (Digital Signature Authorization) предложен в 1991 году в США и явля-

ется развитием алгоритма цифровой подписи Эль-Гамала. Отправитель и получатель электронного документа используют при вычислении большие целые числа:

G, P – простые числа по L -бит каждое ($L = 512 \dots 1024$ бит);

q – простое число длиной 160 бит делитель числа $(P - 1)$.

G, P, q являются открытыми и могут быть общими для всех пользователей сети.

Описание алгоритма

1. Отправитель выбирает случайное целое число X , $1 < X < q$. Число X является *секретным ключом* отправителя для формирования электронной подписи.

2. Отправитель вычисляет значение

$$Y = G^X \bmod P.$$

Число Y является *открытым ключом* для проверки подписи отправителя и передается всем получателям документа.

3. Для того чтобы подписать документ M , отправитель хэширует его в целое хэш-значение m :

$$m = h(M), 1 < m < q.$$

Затем генерирует случайное целое число K , $1 < K < q$ и вычисляет число r :

$$r = (G^K \bmod P) \bmod q.$$

4. При помощи секретного ключа X отправитель вычисляет число s :

$$s = ((m + r \cdot X) / K) \bmod q.$$

Пара чисел r, s образуют цифровую подпись $S = (r, s)$ под документом M .

5. Доставленное получателю сообщение вместе с подписью представляет собой тройку чисел $[M, r, s]$. Прежде всего получатель проверяет выполнение соотношений:

$$0 < r < q; 0 < s < q.$$

6. Далее получатель вычисляет значения:

$$w = 1/s \bmod q;$$

$$t = h(M) - \text{хэш-значение};$$

$$u_1 = (t \cdot w) \bmod q;$$

$$u_2 = (r \cdot w) \bmod q.$$

Затем при помощи открытого ключа Y вычисляется значение

$$v = ((G^{u_1} \cdot Y^{u_2}) \bmod P) \bmod q,$$

и проверяется выполнение равенства $v = r$. Если оно выполняется, то подпись признается подлинной, так как можно строго математически доказать, что последнее равенство будет выполняться тогда и только тогда, когда подпись $S = (r, s)$ под документом M получена при помощи именно того секретного ключа X , из которого был получен открытый ключ Y .

Новые стандарты ЭЦП. Последние достижения криптографии показали, что общая проблема логарифмирования в дискретных полях, являющаяся базой указанной схемы ЭЦП, не может считаться достаточно прочным фундаментом. Например, размеры блоков, считающиеся "безопасными", растут сравнительно быстрыми темпами. В результате это привело к тому, что стандарты ЭЦП России и США в 2001 году были обновлены: переведены на эллиптические кривые. Схемы ЭЦП при этом остались прежними, но в качестве чисел, которыми они оперируют, теперь используются не элементы конечного поля $GF(2^n)$ или $GF(p)$, а эллиптические числа - решения уравнения эллиптических кривых над указанными конечными по-

лями. Роль операции возведения числа в степень в конечном поле в обновленных стандартах выполняет операция взятия кратной точки эллиптической кривой – “умножение” точки на целое число.

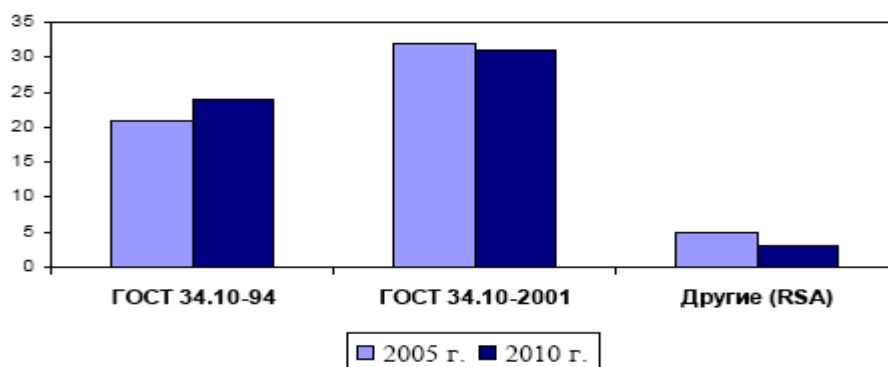


Рисунок 2. Алгоритмы ЭЦП, используемые в РФ

Надлежащий выбор типа эллиптической кривой позволяет многократно усложнить задачу взлома схемы ЭЦП и уменьшить рабочий размер блоков данных. Старый российский стандарт ЭЦП оперирует 1024-битовыми блоками, а новый, основанный на эллиптических кривых, - 256-битовыми, и при этом обладает большей стойкостью. Стойкость схемы подписи ГОСТ Р34.10-94 базируется на сложности решения задачи дискретного логарифмирования в простом поле. В настоящее время наиболее быстрым алгоритмом ее решения для общего случая является алгоритм обобщенного решета числового поля. В ГОСТ Р34.10-2001 стойкость схемы ЭЦП основана на сложности решения задачи дискретного логарифмирования в группе точек эллиптической кривой. При правильном выборе параметров кривой самыми эффективными методами ее решения являются более трудоемкие r - и l -методы Полларда. Так, по разным оценкам специалистов, трудоемкость взлома старого и нового стандартов ЭЦП России составляет величину порядка 10^{26} и 10^{38} операций умножения в базовом поле $GF(p)$ соответственно. Очевидно, что новый стандарт более стойкий.

4. Оборудование и материалы

Для выполнения работы потребуется:

- класс персональных ЭВМ не ниже Pentium-4, классная доска, мел;
- операционная система Microsoft Windows, программное средство *labWork6.exe*

5. Указания по технике безопасности

Требования безопасности перед началом работ:

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории;
- убедиться в целостности электрических розеток и разъемов. Запрещается работать с неисправным электроинструментом, разбивать и самостоятельно ремонтировать его;
- в лаборатории необходимо быть в сменной обуви;
- включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

Требования безопасности во время работы:

- выполняя лабораторную работу, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и дискеты, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отсоединение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса.

Требования безопасности по окончании работы:

- по команде преподавателя студенты обязаны выключить компьютер и составляющие вычислительного комплекса;
- рабочее место привести в порядок.

6. Задания

Задание №1. Ознакомиться с основными направлениями работ в рамках федеральной целевой программы “Электронная Россия”, а также со сведениями о порядке использования и действующих алгоритмах постановки электронной цифровой, изложенными в параграфе 3. Запустить программу *labWork6.exe*, предназначенную для демонстрации порядка постановки и проверки электронной цифровой подписи.

Задание №2. Сгенерировать и переслать участникам обмена ключи для шифрования исходного документа и ключи для подписания документа. Исходный текст для шифрования набирается непосредственно в окне программы.

Задание №3. Зашифровать исходное сообщение и подписать его на секретном ключе отправителя.

Задание №4. Переслать зашифрованное и подписанное сообщение получателю. Выполнить проверку правильности ЭЦП и восстановить исходный текст сообщения.

Задание №5. Сохранить в отчете экранные формы, демонстрирующие процесс генерации и распространения ключей; процесс шифрования исходного документа и постановки ЭЦП.

Задание №5. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

7. Содержание отчета

Отчет должен содержать:

1. Наименование работы, цель.
2. Последовательность действий при выполнении каждого задания.
3. Выводы по выполнению лабораторной работы.

8. Контрольные вопросы

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	В чем состоит назначение хэш-функций и какие требования предъявляются к хэш-функциям, используемым для постановки ЭЦП? Перечислите стандарты хэш-функций, действующие в Российской Федерации.
2,4,6,8, 20,22,24, 26,30	Опишите процедуры постановки и проверки ЭЦП. Какая информация содержится в ЭЦП?
11,13,15, 10,17,19, 27	Перечислите стандарты ЭЦП, действующие в Российской Федерации. На каких принципах основана криптостойкость современных алгоритмов ЭЦП?
12,14,16 21,23,25, 29	Приведите пример реализации алгоритма ЭЦП (<i>RSA, El-Gamal, DSA</i>)

9. Список литературы, рекомендованный к использованию по данной теме

1. Жук, А.П., Жук, Е.П., Лепешкин, О.М., Тимошкин, А.И. Защита информации: Учебное пособие для бакалавров [Текст] / А.П. Жук и др. – М.: ИНФРА-М, 2015. -392с.
2. Шаньгин, В.Ф. Защита компьютерной информации. Эффективные методы и средства [Текст] / В.Ф. Шаньгин – М.: ДМК Пресс, 2012. – 544 с.

Лабораторная работа № 7

Шифрование методом скользящей перестановки

Цель работы: исследования шифра скользящей перестановки с использованием программной реализации *XU-Move*. и умений делать самостоятельные выводы по результатам исследований.

2. Формируемые компетенции

Данная лабораторная работа направлена на формирование следующих компетенций выпускника бакалавриата:

ОПК-1: способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ПК-5: способностью сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.

3. Теоретическая часть

Устойчивые закономерности открытого текста и их использование при дешифровании шифров простой замены и перестановки

Возможность дешифрования какого либо шифра в значительной мере зависит от того, в какой степени криптографические преобразования разрушают вероятностно-статистические закономерности, присутствующие в открытом содержательном тексте. Так в осмысленных текстах любого естественного языка различные буквы встречаются с разной частотой, при этом относительные частоты букв в различных текстах одного языка близки между собой. То же самое можно сказать и о частотах пар, троек букв открытого текста. Кроме того, любой естественный язык обладает так

называемой избыточностью, что позволяет с большой вероятностью “угадывать” смысл сообщения, даже, если часть букв в сообщении не известна.

Таблица 1. Относительные частоты появления букв алфавита русского языка в тексте

1	а - 0,062	12	л - 0,035	23	ц - 0,004
2	б - 0,014	13	м - 0,026	24	ч - 0,012
3	в - 0,038	14	н - 0,053	25	ш - 0,006
4	г - 0,013	15	о - 0,090	26	щ - 0,003
5	д - 0,025	16	п - 0,023	27	ы - 0,016
6	е,ё - 0,072	17	р - 0,040	28	ъ,ь - 0,014
7	ж - 0,077	18	с - 0,045	29	э - 0,003
8	з - 0,016	19	т - 0,053	30	ю - 0,006
9	и - 0,062	20	у - 0,021	31	я - 0,018
10	й - 0,010	21	ф - 0,002	32	- 0,175
11	к - 0,028	22	х - 0,009		

Подобные таблицы приводятся в разных книгах. Они получены на основе подсчетов частот на больших объемах открытого текста. Учитывая, что для

экспериментов берется различный исходный материал, значения вероятностей несколько отличаются между собой. Если упорядочить буквы по убыванию вероятностей, то мы получим следующий вариационный ряд:

О,Е,А,И,Н,Т,С,Р,В,Л,К,М,Д,П,У,Я,З,Ы,Б,Ь,Г,Ч,Й,Х,Ж,Ю,Ш,Ц,Щ,Э,Ф

Например, в слове **СЕНОВАЛИТР** содержатся 10 наиболее часто встречающихся букв.

Частоты знаков алфавита зависят не только от языка, но и от характера текста. Так в тексте по криптографии будет повышена вероятность букв Ф, Ш (из-за часто встречающихся слов “шифр”, “криптография”). В некоторых математических текстах может быть завышена частота буквы Ф (из-за слов “функция”, “функционал” и т.п.). В стандартных текстовых файлах наиболее частым является символ “пробел”. Частотная диаграмма содержа-

тельных текстов является устойчивой характеристикой текста. Из теории вероятностей следует, что при достаточно слабых ограничениях на вероятностные свойства случайного процесса справедлив закон больших чисел, т.е. относительные частоты ϑ/N знаков сходятся по вероятности к значениям их вероятностей p_k

$$P\{|\frac{\vartheta_k}{N} - p_k| > \varepsilon\} \xrightarrow{N \rightarrow \infty} 0.$$

Шифры перестановки и простой замены не полностью разрушают вероятностно-статистические свойства, имеющиеся в открытом сообщении.

При дешифровании текста, зашифрованного шифром простой замены, используют частотные характеристики открытого текста. Именно, если подсчитать частоты встречаемости знаков в зашифрованном тексте, упорядочить их по убыванию, и сравнить с вариационным рядом вероятностей открытого текста, то эти две последовательности будут близки.

Скорее всего на первом месте окажется пробел, далее будут следовать буквы О, Е, А, И.

Конечно, если текст не очень длинный, то не обязательно полное совпадение. Может оказаться на втором месте О, а на третьем Е, но в любом случае в первых и вторых рядах одинаковые буквы будут располагаться недалеко друг от друга, и чем ближе к началу (чем больше вероятность знаков), тем меньше будет расстояние между знаками.

Аналогичная картина наблюдается и для пар соседних букв (биграмм) открытого текста (наиболее частая биграмма русского открытого текста - СТ). Однако для получения устойчивой картины длина анализируемой последовательности должна быть достаточно большой. На сравнительно небольших отрезках открытого текста эта картина как-то смазана. Более устойчивой характеристикой биграмм является отсутствие в осмысленном тексте некоторых биграмм, как говорят, наличие запретных биграмм, име-

ющих вероятность равную практически 0. Видели ли вы когда-нибудь в открытом тексте биграмму ЪЪ, или биграммы вида: “гласная” Ъ; “пробел” Ъ? Знание и использование указанных особенностей открытого текста значительно облегчает дешифрование шифра перестановки и замены.

Шифр перестановки. Положим X – множество открытых (содержательных) текстов в алфавите I . Длины всех возможных открытых текстов кратны величине T . Множеством ключей является симметрическая группа подстановок S_T степени T , для $g \in S_T$ определим функцию шифрования fg положив для $(i_1, i_2, \dots, i_T) \in X$

$$fg(i_1, i_2, \dots, i_T) = (i_{g(1)}, i_{g(2)}, \dots, i_{g(T)});$$

доопределим fg на остальных элементах из X по правилу: текст $x \in X$ делится на отрезки длины T и каждый отрезок длины T шифруется на ключе g по указанному выше закону шифрования. Последовательность, составленная из букв образов зашифрованных слов, является шифрованным текстом, соответствующим открытому тексту x и ключу g . Для шифрования текста длины не кратной T его дополняют буквами до длины кратной T .

Дешифрование шифра перестановки. Шифрованный текст записывается в таблицу с T столбцами. Для восстановления открытого текста шифра перестановки нам необходимо переставить колонки таким образом, чтобы в строках появился осмысленный текст.

Рассмотрим пример дешифрования шифра перестановки восьми столбцов (Учебное пособие «Принципы и методы защиты информации» Проскурин Г.В.). Пусть шифротекст имеет следующий вид:

1	2	3	4	5	6	7	8
п	а	я		в		и	м
о	ч	ш	г		у		е
е	б	ж	л		е		о
м		ч		о	т	о	я
	е	г	е		у	с	щ
	а	к	ь	з	а	т	т
я	р	е		е	п		ь
	ю	з	в	а	н	в	
о	й	а	в	е	ш	л	
	е	е	я	м		п	н
ь	р	р	н	з	е	е	е
з	а	м	а	н		а	к
ч	с	т	а		ь	а	н
о	я	л	м		а	л	
о	ь	ч	х	т	а	т	
в		е	о	а	л	е	п
о	е	р	м	т	ь	е	
д	с	г	ы		о	а	т
е	б	в	н		ы		
	а	у	н	н	з	н	л
	г	и	а	о	к	к	д
	а	о	б	д	г	н	
	ж	а	у	е	д	я	д
х	л	и		е	м	о	а
к	р	т	д		ь	о	е
	ь	х	в	т	о	н	
р	л	е		е	д	а	ю
р		з	е	в		е	д
ш		в	а	е	н	е	н
т	и	й	е	в			д
		в		с	д		

Сопоставим перестановке столбцов таблицу 8×8 , при этом поставим на пересечении i -той строки и j -ого столбца единицу, если j -ая колонка после обратной перестановки должна следовать за i -ой. Наша задача восстановить таблицу, отвечающую правильной перестановке столбцов.

Давайте теперь попарно пристраивать один столбец к другому. Если при этом в некоторых строках появляются запретные биграммы, то столбцы не могут в открытом тексте следовать друг за другом и соответствующая клеточка зачеркиваются. В нашем примере 6-ой столбец не может следовать за 4-ым, т.к. иначе в тексте в первой строке будет подряд два пробе-

ла. Посмотрим, например, 6-ую строку. Если бы 4-ый столбец следовал за 1-ым, то в тексте были бы слова, начинающиеся с Ь.

После просмотра всех строк мы получим следующую таблицу:

	1	2	3	4	5	6	7	8
1	х	х		х	х	х		х
2		х		х		х		
3			х					х
4	х	х		х		х	х	х
5	х				х	х	х	х
6	х	х		х		х	х	
7	х			х	х	х	х	х
8	х	х			х	х	х	х

Если бы текст был подлиннее и строк было бы побольше, то в каждой строке и в каждом столбце осталось бы ровно по одной не зачеркнутой клеточке и перестановка была бы восстановлена.

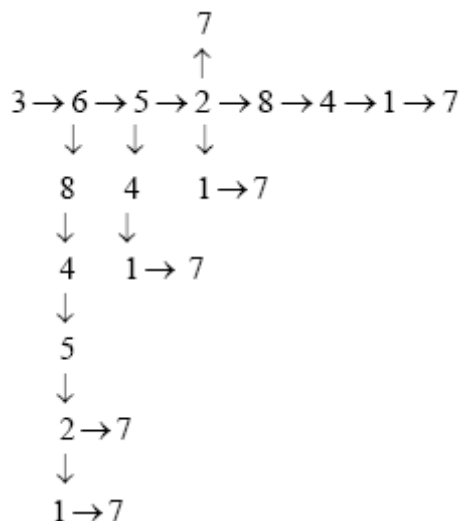
В нашей таблице мы только можем утверждать, что 6-ой столбец следует за 3-им (обозначим это событие следующим образом $3 \rightarrow 6$), если 6-ой столбец не является последним. Для 6-го столбца может быть два варианта продолжения

$$\begin{array}{c} 8 \\ \uparrow \\ 3 \rightarrow 6 \rightarrow 5 \end{array}$$

Нам надо рассмотреть оба и постараться отсеять ложный вариант. Если отсеять ложный вариант не удастся, то надо продолжать оба варианта

$$\begin{array}{cc} 8 \rightarrow 4 & 1 \\ \uparrow & \uparrow \\ 3 \rightarrow 6 \rightarrow 5 \rightarrow 2 \rightarrow 7 \end{array}$$

В итоге получаем некоторое дерево возможного следования столбцов в открытом тексте.



Каждой ветви дерева соответствует некоторая перестановка столбцов

Далее проверяем каждый вариант на осмысленность и получаем правильный вариант.

$3 \rightarrow 6 \rightarrow 5 \rightarrow 2 \rightarrow 8 \rightarrow 4 \rightarrow 1 \rightarrow 7$

Заметим, что не обязательно было строить дерево до конца. Например, ветвь $3 \rightarrow 6 \rightarrow 8 \rightarrow 4 \rightarrow 5$ можно было отсечь сразу. Разве можно признать осмысленным следующий фрагмент текста.

3	6	8	4	5
я		м		в
ш	у	е	г	
ж	е	о	л	
ч	т	я		о
г	у	щ	е	з
к	а	т	ь	е
е	а	т		а

Такая процедура отсечения ветвей была бы просто необходима, если бы строк было поменьше, и дерево было бы соответственно гораздо ветвистей.

Предложенную процедуру легко автоматизировать и сделать пригодной для реализации на ЭВМ. Алгоритм дешифрования должен состоять из следующих этапов.

1. Предварительная работа. Анализируя достаточно представительный объем открытых текстов построить множество запретных биграмм.

2. Предварительная работа. Составить словарь всех возможных v -грамм для $v=2,3,\dots,d$, которые могут встретиться в открытом тексте. Число d выбирается, исходя из возможностей вычислительной техники.

Построить таблицу 8×8 . При этом перебираются последовательно все запретные биграммы и для каждой опробуемой биграммы - последовательно все строки. Если хотя бы в одной строке первый символ биграммы встречается в i -том столбце, а второй в j -ом, то клеточка $i \times j$ таблицы зачеркивается.

3. Выбрать некоторый столбец в качестве начального столбца.

4. Начать процедуру построения дерева путем пристраивания к исходному столбцу всех вариантов столбцов.

5. Для каждого полученного варианта добавить еще один из оставшихся столбцов. Если хотя бы в одной из строк таблицы встретится 3-грамма, которая отсутствует в словаре размещенных 3-грамм, то вариант отсеивается.

6. Для каждого из не отсеянных вариантов добавляем еще один столбец и проводим отсев ложных вариантов по словарю разрешенных 4-грамм.

Если словарь был построен только для $d \leq 3$, то отсев проводится путем проверки на допустимость 3-грамм, встретившихся в последних 3-х столбцах каждой строки. Продолжаем этот процесс до получения полной перестановки.

Ниже, в таблице, приведен восстановленный для нашего примера текст.

	1	2	3	4	5	6	7	8
1	я		в	а	м		п	и
2	ш	у		ч	е	г	о	
3	ж	е		б	о	л	е	
4	ч	т	о		я		м	о
5	г	у		е	ш	е		с
6	к	а	з	а	т	ь		т
7	е	п	е	р	ь		я	
8	з	н	а	ю		в		в
9	а	ш	е	й		в	о	л
10	е		м	е	н	я		п
11	р	е	з	р	е	н	ь	е
12	м		н	а	к	а	з	а
13	т	ь		с	н	а	ч	а
14	л	а		я		м	о	л
15	ч	а	т	ь		х	о	т
16	е	л	а		п	о	в	е
17	р	ь	т	е		м	о	е
18	г	о		с	т	ы	д	а
19	в	ы		б		н	е	
20	у	з	н	а	л	и		н
21	е	к	о	г	д	а		к
22	о	г	д	а		б		н
23	а	д	е	ж	д	у		я
24	и	м	е	л	а		х	о
25	т	ь		р	е	д	к	о
26	х	о	т	ь		в		н
27	е	д	е	л	ю		р	а
28	з		в		д	е	р	е
29	в	н	е		н	а	ш	е
30	й		в	и	д	е	т	ь
31	в	а	с					

Дальнейшее развитие шифры перестановки получили при осуществлении идеи непрерывной локальной перестановки символов открытого текста под действием управляющей последовательности. Для осуществления перемешивания знаков открытого текста в памяти шифратора запоминаются отдельные знаки текста и проводится задержка их передачи в дискретном времени. Введем параметры n_1 и n_2 , так, что $n=n_1+n_2$. В этих шифрах i -й знак передаваемого сообщения переставляется в шифрованном сообщении на j -е место, где $i-n_1 \leq j \leq i+n_2$. Управляющую последовательность временем задержки стараются выбрать так, чтобы время задержки каждого символа было случайной величиной с равномерным распределением, то есть вероятность каждого фиксированного значения времени задержки должна быть близка к $1/n$.

Шифрующий автомат скользящей перестановки

Рассмотрим схему шифрующего автомата, позволяющего при подходящей управляющей последовательности реализовать произвольный шифр скользящей перестановки, рис.7.1.

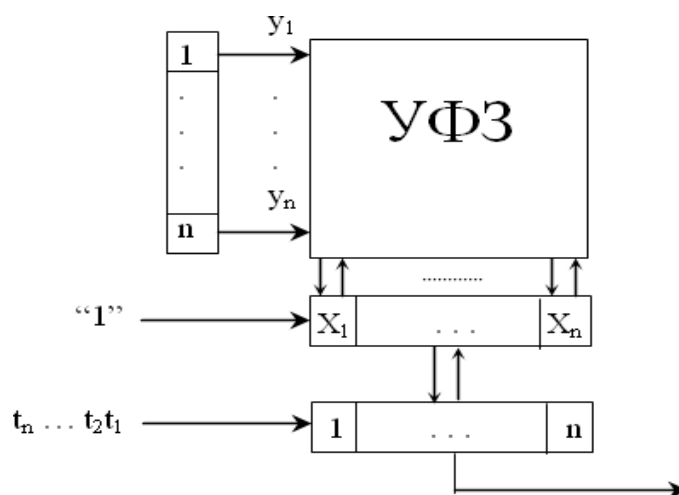


Рис. 7.1. Схема реализации шифрующего автомата скользящей перестановки

На вход узла формирования задержки (УФЗ) в каждом такте t подается вектор

Узел формирования задержки является конечным автоматом $(F^n, F^n, \{1, \dots, n\}, \delta, \lambda)$, множеством состояний которого является множество всевозможных двоичных векторов - заполнений $x(t) = (x_1^t, \dots, x_n^t)$ нижнего накопителя.

В такте t вырабатывается натуральное число – значение задержки.

$$\gamma_t = \max j : \left\{ x_i^t \mid y_i^t \in \{0,1\}, i = \overline{2, n-1}, y_1^t = y_n^t = 1. \right.$$

При этом автомат переходит в следующее состояние:

$$\bar{x}=(t+1)=(x_1^{t+1}, \dots, x_n^{t+1}),$$

где

$$x_1^{t+1}=1,$$

$$x_i^{t+1}=\begin{cases} 0, & i=\gamma_t+1 \\ x_i^t, & i\in\{2,\dots,\gamma_t,\gamma_t+2,\dots,n\} \end{cases}$$

Знаки открытого текста записываются на нижний накопитель. В линию в t -ом такте посылается знак открытого текста, записанный в ячейке с номером.

Состояния автомата $x(t)$ являются индикаторами, показывающими какие из знаков открытого текста еще не считаны с нижнего накопителя.

Для зашифрования последовательности $t_N, \dots, t_2, t_1$ поступают следующим образом. Сначала записываем в нижний накопитель первые n_1 знаков открытого текста

$$(t_{n_1}, \dots, t_1, 0, \dots, 0)$$

Одновременно автомат устанавливается в начальное состояние

$$x(1) = (0, \dots, 0).$$

После этого автомат УФЗ начинает работать по описанному выше закону до тех пор, пока в накопитель не поступит последний знак t_N открытого текста. С прекращением подачи на накопитель знаков открытого текста происходит прекращение подачи единиц на накопитель-индикатор. В оставшееся n_1 тактов производится считывание записанной в накопителе информации.

При расшифровывании УФЗ работает по той же схеме, только вместо считывания необходимо организовывать запись информации во второй накопитель.

Особенности работы УФЗ

В каждом такте t (за исключением последних n_1 тактов) в накопителе-индикаторе $\vec{x}(t)$ записано ровно n_1 единиц. Поэтому в такте t величина задержки может принимать только одно из n_1 значений в интервале $\{1, \dots, n\}$. В частном случае, когда $n=1$, либо $n_1=n$ УФЗ вырабатывает постоянно значения задержки $\gamma_i=1$ и $\gamma_i=n$ соответственно. Легко видеть, что результирующее преобразование открытого текста действительно будет шифром скользящей перестановки. Условие $y_1^t=1, t=1, 2, \dots$ обеспечивает постоянное считывание во всех тактах, а условие $y_2^t=1$ ограничивает величину задержки $\gamma_i \leq n$.

Пример, поясняющий работу шифрующего автомата скользящей перестановки
Возьмем для примера $n=7, n_1=3, n_2=4$

На вход УФЗ на каждом шаге t работы шифрующего автомата подается вектор

$\vec{y} = (y_1^t, \dots, y_n^t)$, в линейном регистре сдвига (ЛРС), рис. 7.2.

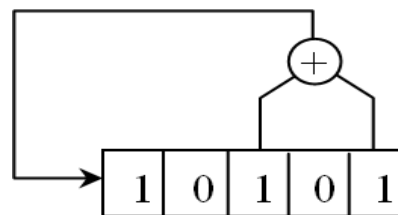


Рис.7.2. Линейный регистр сдвига

Мы надеемся, что читатель сможет написать последовательность состояний данного линейного регистра сдвига, с помощью которой образуется управляющая последовательность шифрующего автомата.

Будем обозначать на каждом шаге работы шифрующего автомата последовательность $Y_1, Y_2, \dots, Y_n$ (в нашем случае $Y_1, Y_2, \dots, Y_7$), поступающую с ЛРС, как

«Y», а последовательность единиц $X_1, X_2, \dots, X_n$ (в нашем случае $X_1, X_2, \dots, X_7$) как «1». В нижней строке будем записывать знаки открытого текста, находящиеся на данном шаге в нижнем накопителе шифрующего автомата $t_1, t_2, \dots, t_n$ (в нашем случае $t_1, t_2, \dots, t_7$). Рассмотрим пошагово работу шифратора при конкретных условиях.

1-й шаг

"Y"	1	1	0	1	0	1	1
"1"	1	1	1	0	0	0	0
	t_3	t_2	t_1	0	0	0	0

2-й шаг

"Y"	1	0	1	0	1	0	1
"1"	1	1	0	1	0	0	0
	t_4	t_3	t_2	t_1	0	0	0

3-й шаг

"Y"	1	0	0	1	0	1	1
"1"	1	0	1	0	1	0	0
	t_5	t_4	t_3	t_2	t_1	0	0

4-й шаг

"Y"	1	0	0	0	1	0	1
"1"	1	0	0	1	0	1	0
	t_6	t_5	t_4	t_3	t_2	t_1	0

5-й шаг

"Y"	1	0	0	0	0	1	1
"1"	1	0	0	0	1	0	1
	t_7	t_6	t_5	t_4	t_3	t_2	t_1

6-й шаг

"Y"	1	1	0	0	0	0	1
"1"	1	1	0	0	0	1	0
	t_8	t_7	t_6	t_5	t_4	t_3	t_2

7-й шаг

"Y"	1	0	1	0	0	0	1
"1"	0	1	0	0	0	0	1
	0	t_8	t_7	t_6	t_5	t_4	t_3

8-й шаг

"Y"	1	0	0	1	0	0	1
"1"	0	0	1	0	0	0	0
	0	0	t_8	t_7	t_6	t_5	t_4

На каждом шаге мы искали начиная с левого края и идя направо первое совпадение в строках «Y» и «1» (1 в обеих строках) и для удобства обводили этот столбец. Элемент открытого текста, который оказался в выбранном столбце уходит в линию. Таким образом, в нашем примере, последовательность ушедшая в линию имеет следующий вид: $t_2, t_4, t_5, t_6, t_1, t_7, t_3, t_8$

Описание программной реализации

Для выполнения лабораторной работы на компьютере необходимо установить программный модуль XY-Mover.

1) Строка меню В данной программе меню состоит из трех пунктов: Шифрование, Вид, Помощь. Необходимый пункт меню можно выбрать, щелкнув по нему левой кнопкой мыши, или с помощью кнопок клавиатуры «вправо», «влево», нажав перед этим функциональную клавишу «F10». После того как пользователь выбрал необходимый ему пункт меню, откроется ниспадающее подменю. Рассмотрим пункты меню подробнее.

- *Шифрование.* Пункты ниспадающего меню можно выбрать либо левой кнопкой мыши, либо кнопками клавиатуры «вверх», «вниз». Рассмотрим подробнее пункты подменю.

- Редактирование параметров – эта функция меню позволяет задать необходимые параметры в поле окна программы «Параметры шифратора»;

- Шифровать – выбор этого пункта меню запускает процесс шифрования;

- Дешифрование – выбор этого пункта меню запускает процесс дешифрования;

- Выход – завершение работы программы.

- *Вид.* Этот пункт меню позволяет выбрать внешний вид программы. Ниже приводятся пункты подменю:

- Параметры – выбор этого пункта меню позволяет использовать поле «Параметры шифратора», описанную ниже.

- Схема шифратора – выбор этого пункта меню позволяет наблюдать структурную схему шифрующего автомата.

- Строка состояния – выбор этого пункта меню позволяет наблюдать строку состояния, описанную ниже.

2) Панель инструментов

Возможно, пользователю будет удобнее воспользоваться панелью инструментов (так называемыми «кнопками»), вместо работы с меню. «Кнопки» дублируют некоторые пункты меню, но выбрать «кнопку» гораздо удобнее, чем пункт подменю. Для этого необходимо просто щелкнуть по выбранной кнопке левой кнопкой мыши. Рассмотрим «кнопки» слева направо:

- «А», выбор этой кнопки позволяет получить результат аналогичный пункту меню: Вид-Параметры;
- «Схема шифратора» – выбор этой кнопки позволяет получить результат аналогичный пункту меню Вид- Схема шифратора;
- «Строка состояния» – выбор этой кнопки позволяет получить результат аналогичный пункту меню: Вид- Строка состояния;
- «Редактирование параметров» – выбор этой кнопки позволяет получить результат аналогичный пункту меню: Шифрование-Редактирование параметров;
- «Шифровать» – выбор этой кнопки позволяет получить результат аналогичный пункту меню Шифрование – Шифровать;
- «Дешифровать» – выбор этой кнопки позволяет получить результат аналогичный пункту меню Шифрование – Дешифровать;
- «Прервать» – выбор этой кнопки позволяет прервать процесс обработки данных (шифрование или дешифрование).

3) Строка состояния

Строка состояния находится в нижней части окна программы. На ней выводится информация о состоянии программы:

- «Шифрование методом скользящей перестановки» – это сообщение указывает на то, что программа готова к работе.
- «Завершено ... %» – индикация объема выполненной работы при зашифровании или расшифровании.

4) Описание полей окна программы

- Параметры шифратора

- $n1$ – число знаков, которые записываются в нижний накопитель первыми;
- $n2$ – остальные знаки (всего их 127);
- отводы регистра – точки съема ЛРС;
- начальное заполнение – начальное заполнение ЛРС, которое пользователь может изменять.

- Входной поток

- поле, в котором отражается имя текстового файла, содержание которого нужно шифровать/расшифровать;
- кнопка «открыть» – при нажатии на эту кнопку открывается стандартное для Windows окно «Открытие файла»;
- текст – содержимое файла, открытого для шифрования/расшифрования;
- битовый поток – побитное представление символов выбранного файла.

- Выходной поток

- поле, в котором отражается имя текстового файла, содержание которого нужно расшифровать/шифровать;
- кнопка «открыть» – при нажатии на эту кнопку открывается стандартное для Windows окно «Открытие файла»;
- текст – содержимое файла, открытого для расшифрования/шифрования;
- битовый поток – побитное представление символов выбранного файла.

4. Оборудование и материалы

Для выполнения работы потребуется:

- класс персональных ЭВМ не ниже Pentium-4, классная доска, мел;
- операционная система Microsoft Windows, программное средство *labWork6.exe*

5. Указания по технике безопасности

Требования безопасности перед началом работ:

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории;
- убедиться в целостности электрических розеток и разъемов. Запрещается работать с неисправным электроинструментом, разбивать и самостоятельно ремонтировать его;
- в лаборатории необходимо быть в сменной обуви;
- включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

Требования безопасности во время работы:

- выполняя лабораторную работу, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и дискеты, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отсоединение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса.

Требования безопасности по окончании работы:

- по команде преподавателя студенты обязаны выключить компьютер и составляющие вычислительного комплекса;
- рабочее место привести в порядок.

6. Задания

1. Для выполнения лабораторной работы на компьютере необходимо установить программный модуль XY-Mover.
2. Выполнить начальные установки шифратора, согласно примеру.
3. Загрузить файл для шифрования.
4. Произвести шифрование информации с использованием шифра скользящей перестановки, сохранить шифртекст в файле.

5. Описать в отчете процесс шифрования и расшифрования данных с использованием программы-эмулятора XY-Mover. Проанализировать полученные данные.

6. Включить в отчет о лабораторной работе ответы на контрольные вопросы, выбранные в соответствии с номером варианта, указанным преподавателем.

7. Содержание отчета

Отчет должен содержать:

1. Наименование работы, цель.
2. Последовательность действий при выполнении каждого задания.
3. Выводы по выполнению лабораторной работы.

8. Контрольные вопросы

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Почему шифрование методом гаммирования является наиболее подходящим для высокоскоростных линий телекоммуникационной связи?
2,4,6,8, 20,22,24, 26,30	Какие общие требования, предъявляются к гамме шифра?
11,13,15, 10,17,19, 27	Приведите пример, поясняющий работу шифрующего автомата скользящей перестановки при $n=5$, $n_1=2$, $n_2=3$.
12,14,16 21,23,25, 29	Кратко опишите работу схемы реализации шифра скользящей перестановки.

9. Список литературы, рекомендованный к использованию по данной теме

1. Жук, А.П., Жук, Е.П., Лепешкин, О.М., Тимошкин, А.И. Защита информации: Учебное пособие для бакалавров [Текст] / А.П. Жук и др. – М.: ИНФРА-М, 2015. -392с.

2. Шаньгин, В.Ф. Защита компьютерной информации. Эффективные методы и средства [Текст] / В.Ф. Шаньгин – М.: ДМК Пресс, 2012. – 544 с.

ЛАБОРАТОРНАЯ РАБОТА №8

ИЗУЧЕНИЕ ПРОГРАММНЫХ ПРОДУКТОВ ЗИ.

ПРОГРАММА *PGP* (Pretty Good Privacy)

Цель работы: Ознакомление с общими принципами построения и использования программных средств защиты информации, в частности с программой **PGP**.

Освоение средств программной системы **PGP**, предназначенных для:

- шифрования конфиденциальных ресурсов и разграничения доступа к ним;
- обеспечения целостности информационных ресурсов с помощью механизма электронной цифровой подписи;
- надежного уничтожения остаточной конфиденциальной информации;
- скрывания присутствия в компьютерной системе конфиденциальной информации с помощью виртуального диска.
- Для выполнения лабораторной работы при отсутствии на компьютере программы PGP ее необходимо установить.
- Инсталляционный файл прилагается к описанию лабораторной работы PGPfreeware602i. Выбрать для установки только следующие компоненты:
 - PGP 6.0.2 Program Files;
 - PGP 6.0.2 User's Manual;
 - Unconfigured PGP 6.0.2 Client Install;
 - PGPdisk for Windows.

На вопрос программы установки о существовании ключей ответить «Нет», а на вопрос о необходимости перезагрузки системы – «Да».

Описание лабораторной работы

Возможности PGP 6.0

Основные задачи программы PGP – шифровать файлы и почтовые сообщения, заверять их электронными подписями, полностью уничтожать файлы на диске. Кроме этого, PGP предоставляет следующие возможности:

- хранение открытых ключей на удаленном сервере;
- использование трех симметричных алгоритмов шифрования и двух асимметричных;
- четыре способа запуска: *E-mail plugins*, *PGP tray*, *PGP tools*, контекстное меню;
- разделение ключей;
- установка уровня валидности ключа и доверия владельцу ключа.

PGPkeys

Это программа, входящая в состав PGP 6.0, и обеспечивающая работу с ключами, рис.8.1.

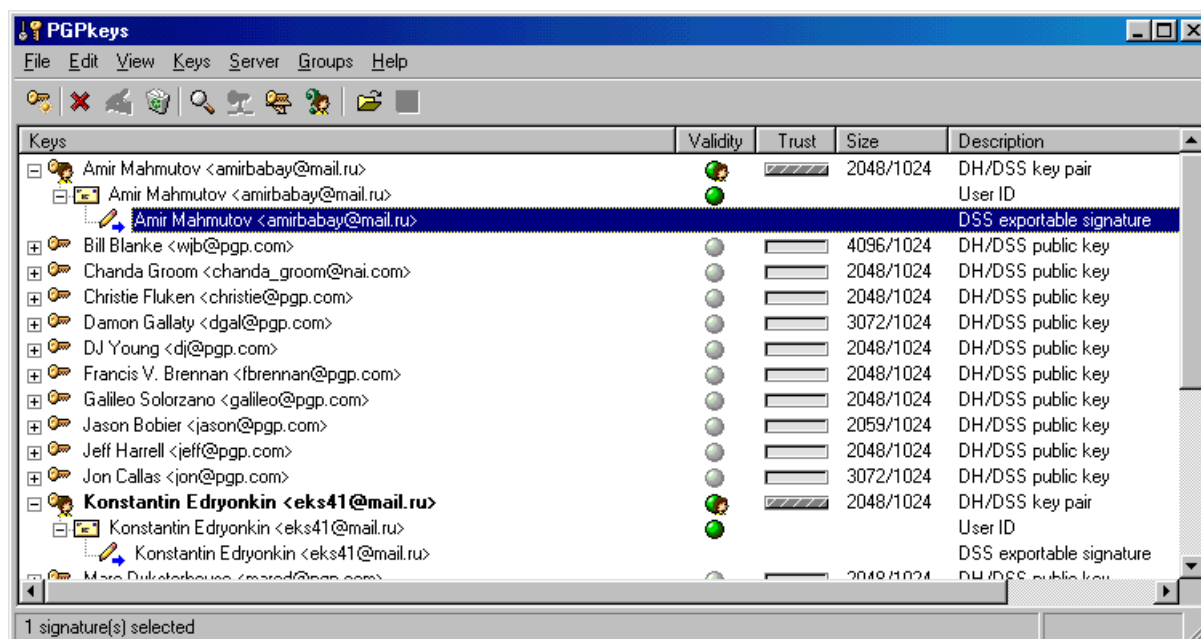


Рис. 8.1 Окно PGPkeys

В меню **File** содержатся две команды, одна из которых **Exit**, другая – **Send Key Shares** позволяет послать локальную копию разделенного ключа по сети.

Меню **Edit** кроме обычных команд содержит команду **Preferences**, которая служит для задания настроек программы, рис.8.2.

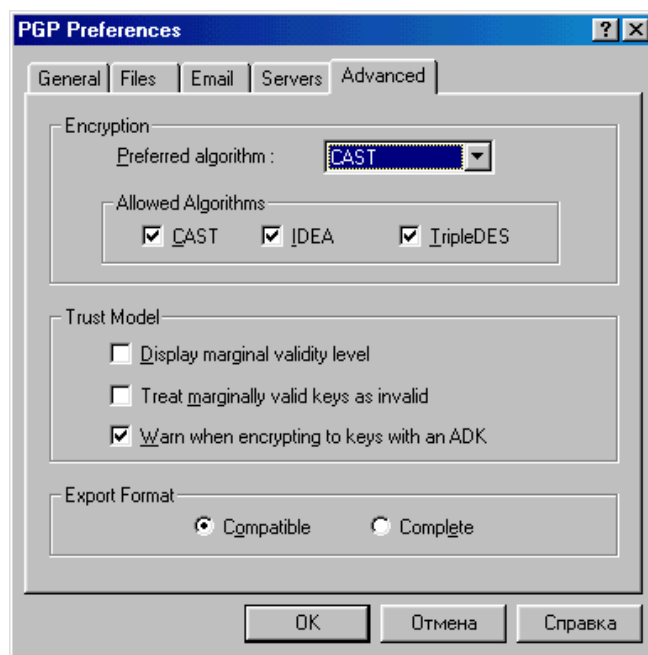


Рис. 8.2. Окно PGPPreferences

С помощью меню **View** можно задать отображаемые на экране свойства ключей.

Команды меню **Keys**

Sign – позволяет подписать своим закрытым ключом открытые ключи других пользователей. Тем самым вы можете показать, что доверяете владельцу используемого ключа, то есть ключ действительно принадлежит владельцу. При этом можно задать модификаторы: 1) *Non Exportable* – для локального набора ключей; 2) *Exportable* – заверенный ключ отсылается на сервер; 3) *Meta-introducer Non-Exportable* – доверяете не только владельцу, но и любому доверенному им ключу; 4) *Trusted Introducer Exportable* – вы доверяете владельцу подписывать ключи, рис.8.3.

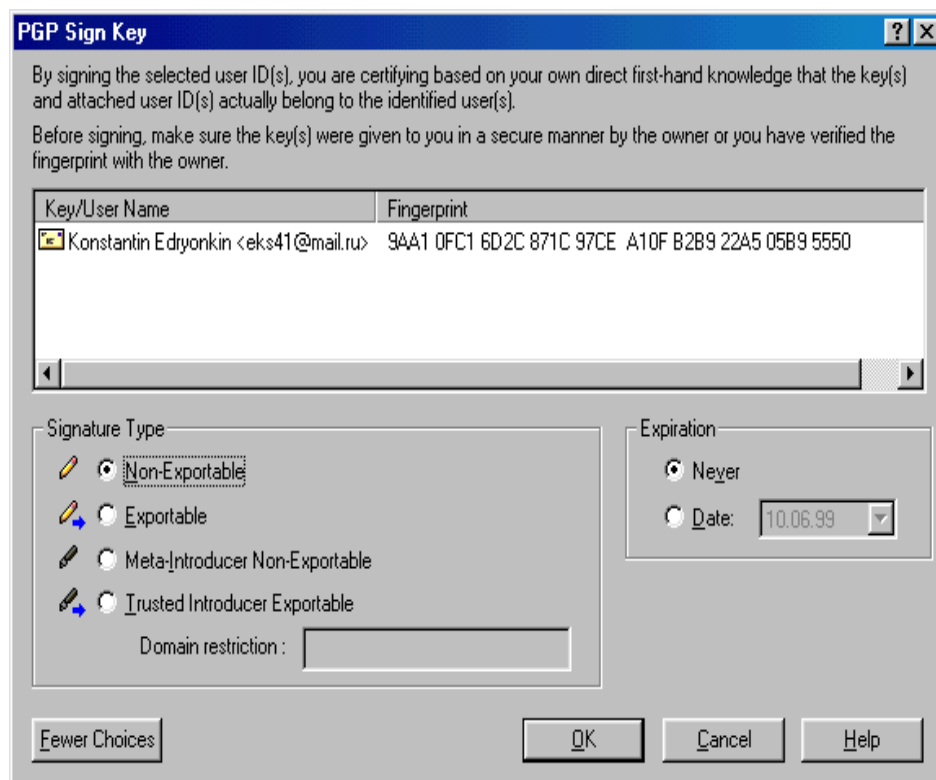


Рис.8.3. Окно PGP SignKey

Set as Default Key – назначить данный ключ ключом по умолчанию. **Add Name/Photo/Revoker** – позволяет добавить к свойствам ключа имя владельца, имеющего право объявить ваш ключ недействительным.

Revoke – объявить ключ недействительным.

Reverify Signatures – проверить сигнатуру (правильность) ключа.

New Key – добавить новый ключ.



Рис. 8.4. Окно Key Generation Wizard

Share Split – разделить ключ между несколькими владельцами.

Import/Export – импортировать/экспортировать ключ из/в текстовый файл. **Key**

Properties – свойства (при этом можно задать степень доверия, валидность ключа), рис.8.4, 8.5.

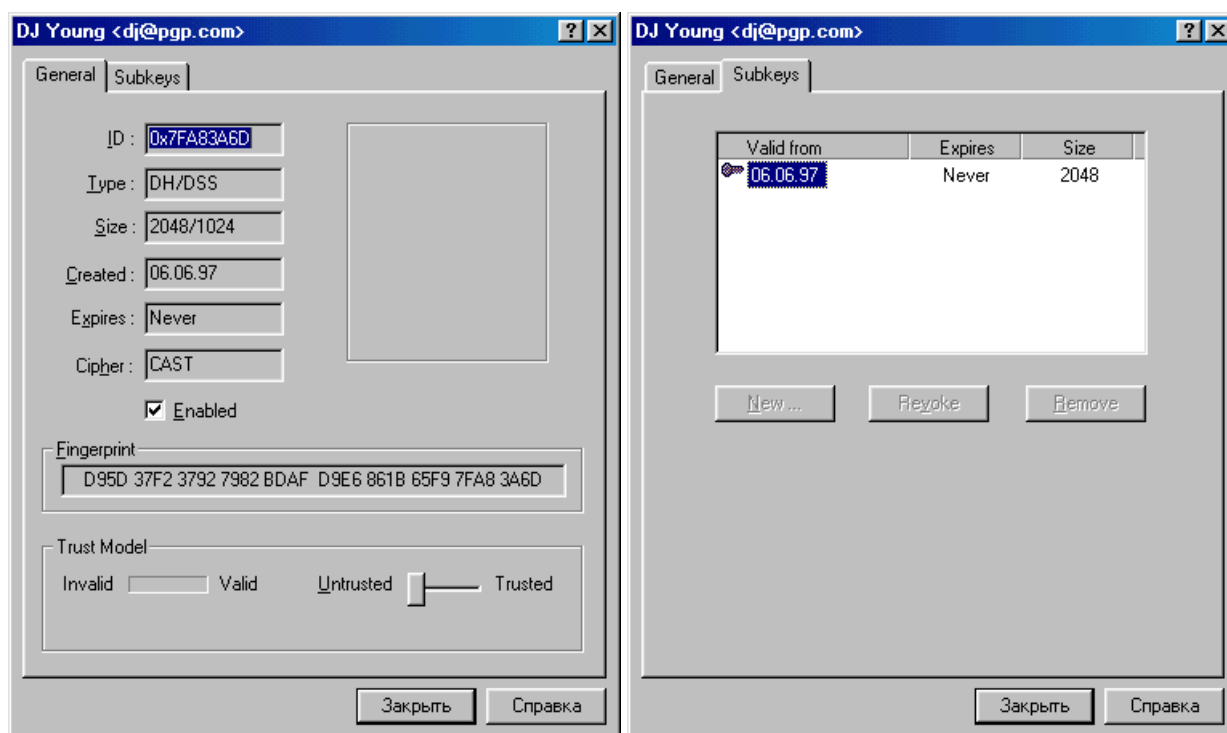


Рис. 8.5. Свойства ключа

В меню **Server** сосредоточены команды для работы с удаленным сервером ключей.

Команда Send to позволяет послать ключи на выбираемый сервер, рис.8.6.

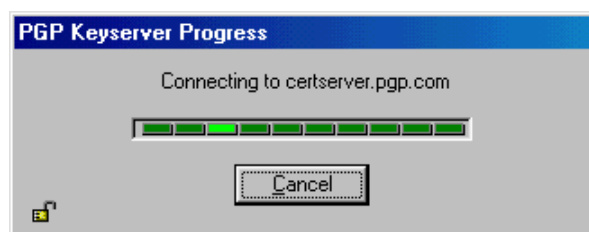


Рис.8.6. Соединение с удаленным сервером

При помощи команды Search можно попытаться найти на удаленном сервере ключ, задав соответствующие параметры, рис.8.7.

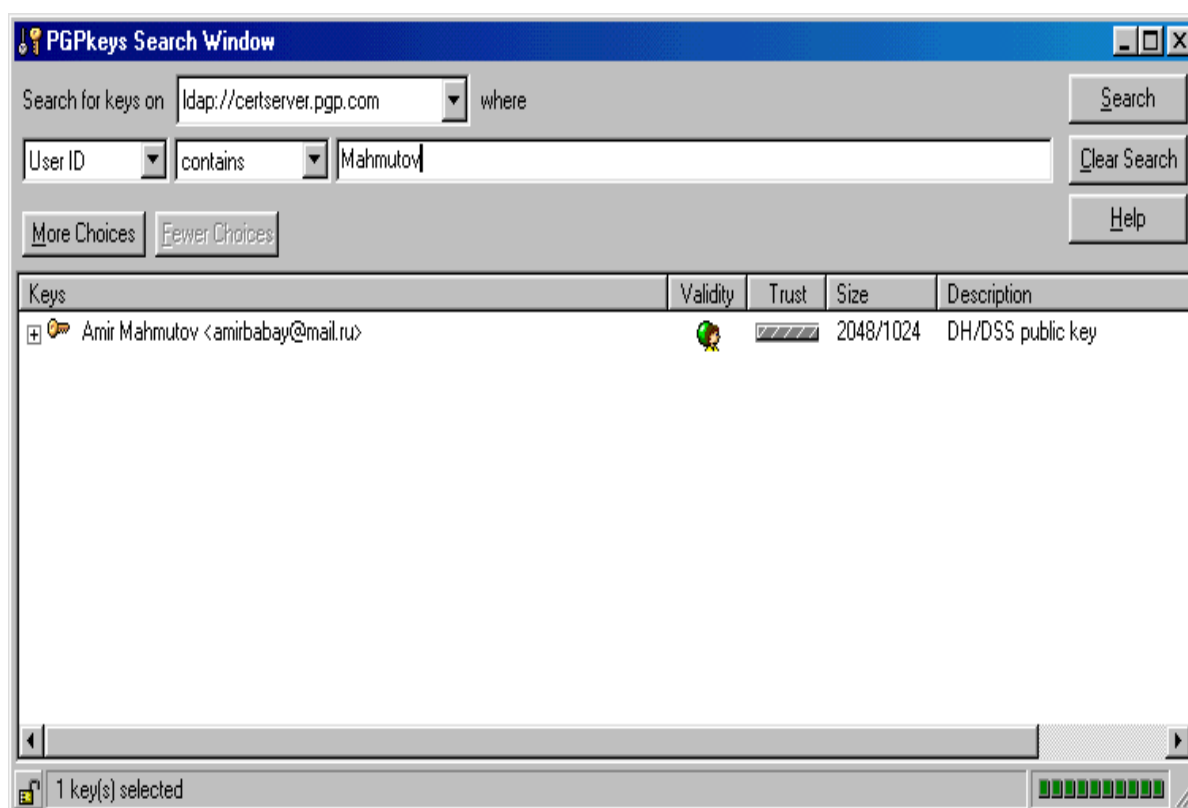


Рис. 8.7. Поиск ключа на удаленном сервере

Команда **Update** позволяет обновить выбранный ключ, получив информацию с сервера.

В меню **Groups** содержатся команды для работы с группами. Команды **New Group, Show Groups, Import Groups** позволяют,

соответственно, добавить группу, показать группы, импортировать группы из файла. Объединение владельцев ключей в группы позволяет легко и просто шифровать сообщения для отправки всем членам группы.

Наконец, меню **Help**, стандартное для *Windows*, позволяет получить справку.

PGPTray

Загружается при запуске *Windows*. Для активизации меню достаточно нажать кнопкой мыши на значок рядом с часами, рис.8.8.

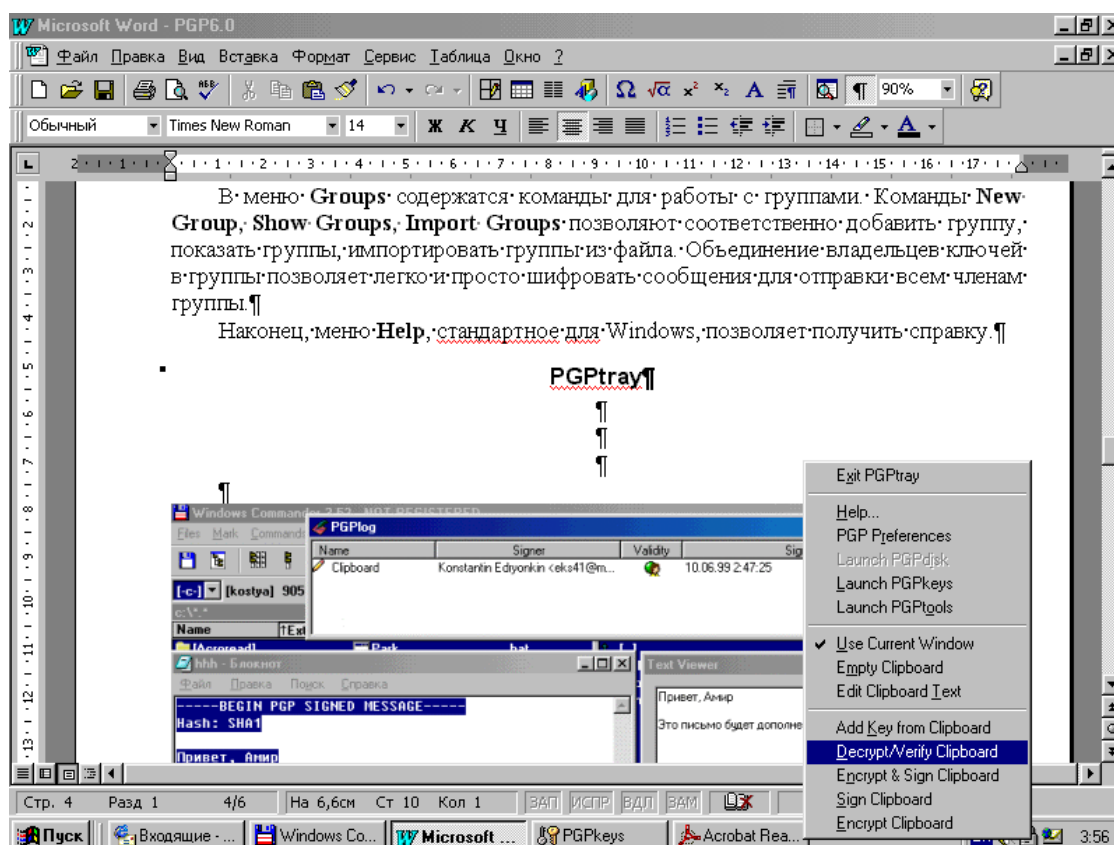


Рис. 8.8. Запуск PGP с использованием значка на панели задач

В меню содержатся команды для выхода, запуска других компонентов *PGP*, редактирования буфера обмена, работы с буфером обмена (добавление ключа, расшифрование/проверка подписи, зашифрование и подпись, просто подпись и просто зашифрование).

Если в используемом почтовом клиенте нет встроенных команд *PGP*, то можно работать с помощью *PGPtray*. Например, для подписи письма достаточно скопировать его в буфер и исполнить команду **Sign Clipboard**.

После этого достаточно вставить в тело письма уже подписанный текст, рис.8.10.

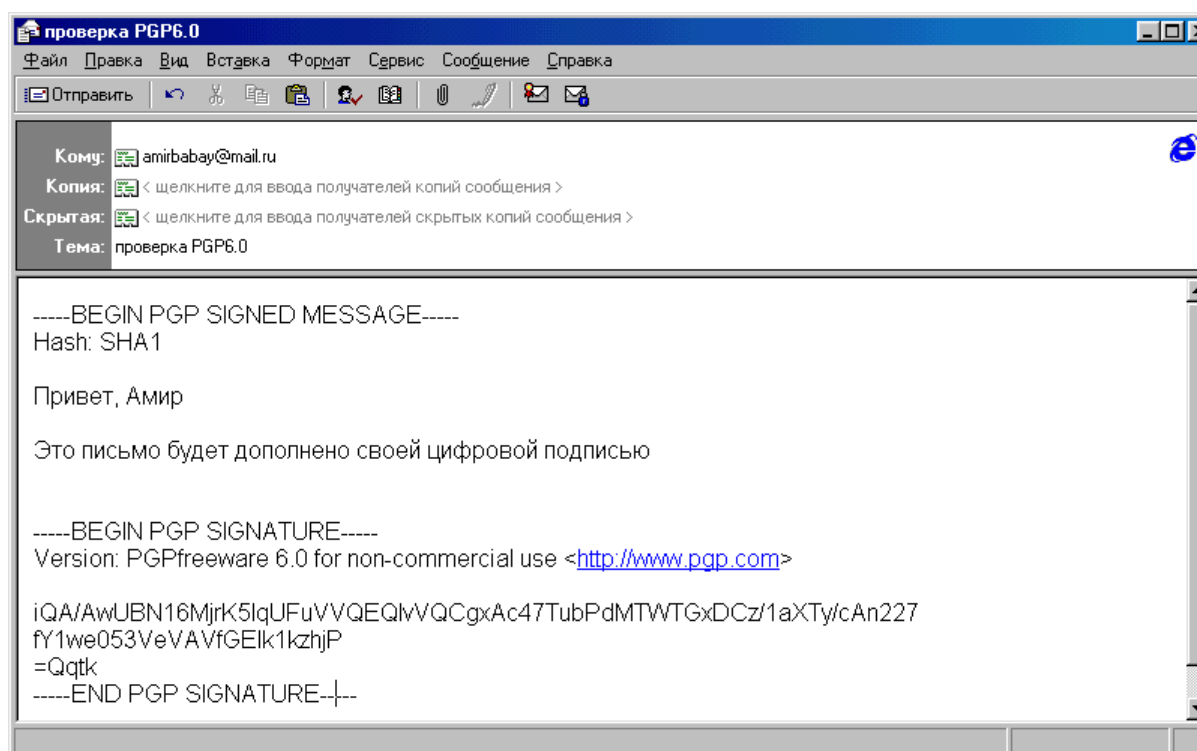


Рис. 8.10. Пример отправки письма с цифровой подписью

Аналогично получателю письма достаточно скопировать полученное письмо в буфер обмена и выбрать команду **Decrypt/Verify Clipboard**.

Естественно, в коллекции ключей должен быть открытый ключ владельца, подписавшего письмо, рис.8.11, 8.12.

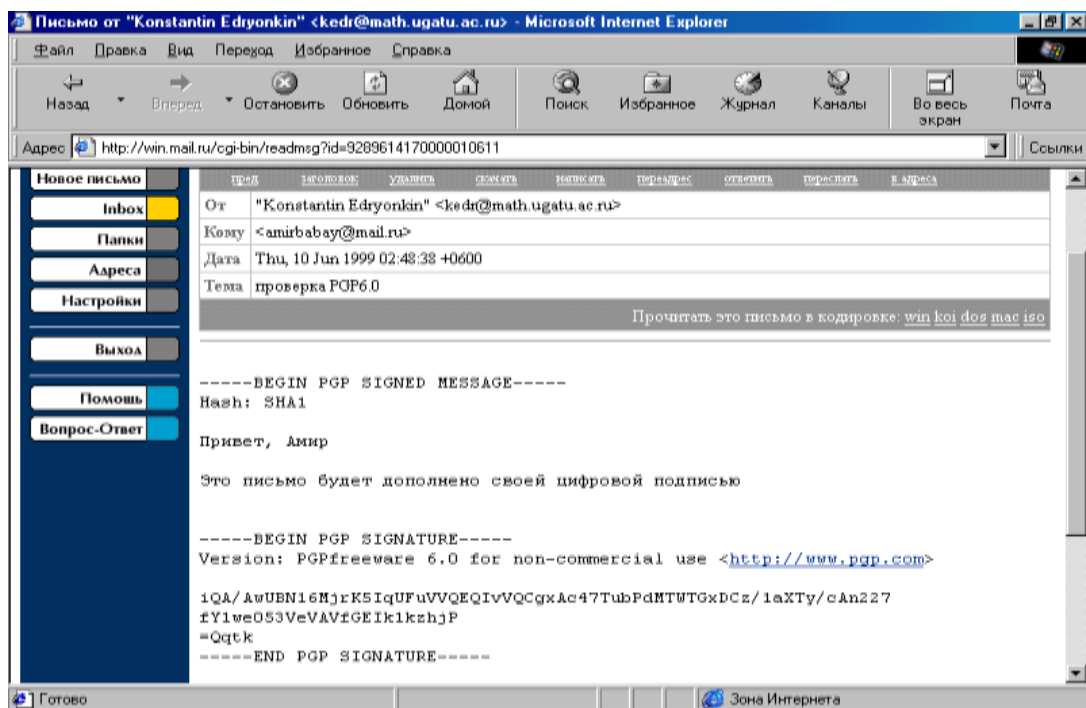


Рис. 8.11. Пример получения письма с цифровой подписью

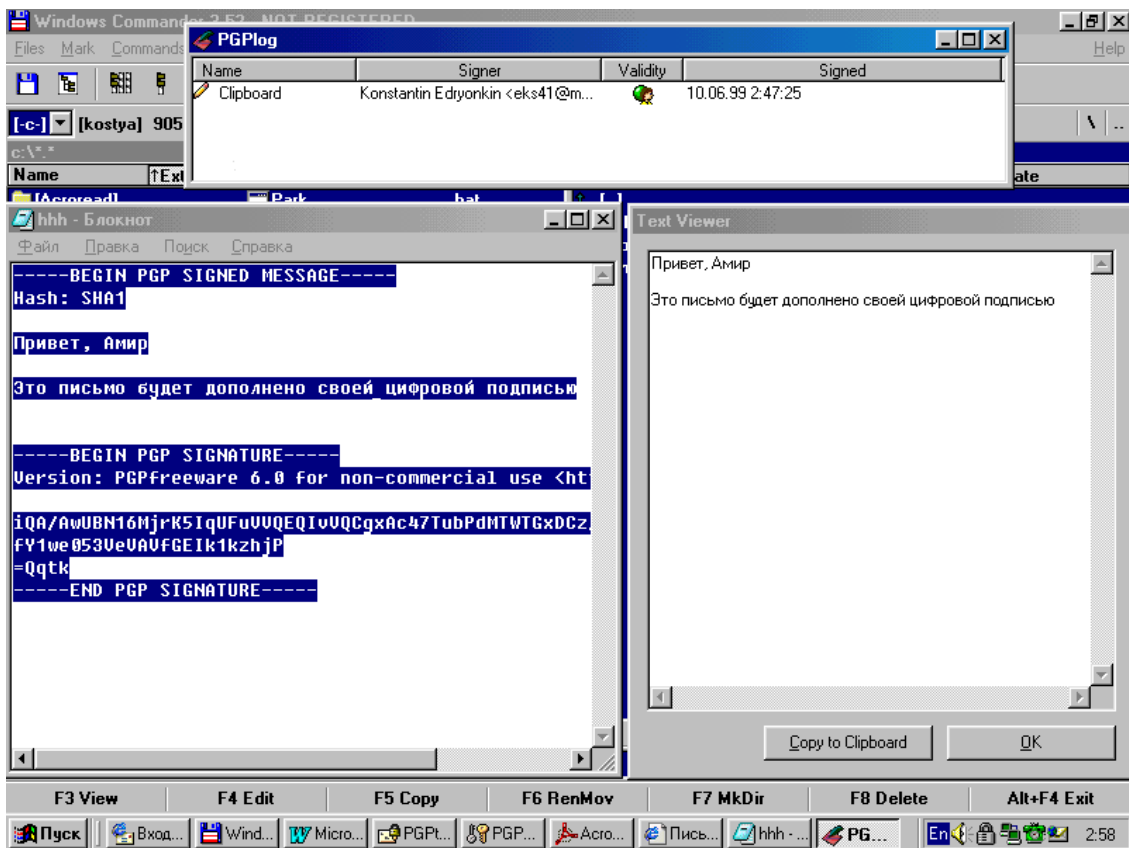


Рис. 8.12. Результаты проверки подписи

PGPtools

Этот компонент *PGP* представляет собой небольшую панель с кнопками, позволяющими выполнить нужное действие: открыть *PGPkeys*, зашифровать, подписать, зашифровать и подписать одновременно, расшифровать/проверить подпись, затереть файл, затереть неиспользуемое дисковое пространство, рис.8.13.

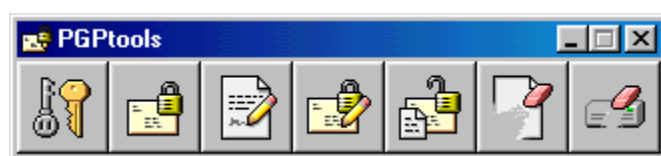


Рис. 8.13. Компоненты панели PGPtools

Альтернативой *PGPtools* могут служить команды, добавляющиеся при установке *PGP* в контекстное меню работы с файлами и меню **Файл** Проводника *Windows*.

Задание

1. Ознакомиться со сведениями о программе PGP (см. параграф 1).
2. Запустить программу PGPtools (с помощью меню «Пуск» или значка PGPtray на панели задач), ознакомиться и отразить в отчете о лабораторной работе *состав программных средств, входящих в систему PGP* (при необходимости воспользоваться справкой о системе PGP).
3. Создать криптографические ключи с помощью программы PGPkeys. Включить в отчет о лабораторной работе *сведения о порядке создания ключей шифрования в системе PGP* и включить в электронную версию отчета копии используемых при этом экранных форм, а также ответы на вопросы:
 - как обеспечивается случайность выбираемых криптографических ключей в системе PGP;
 - как и где хранится секретный ключ пользователя в системе PGP;
 - как может быть обеспечена в системе PGP возможность восстановления секретного ключа пользователя при его случайной утрате.
4. Изучить (на примере обычных текстовых файлов и файлов изображе-

ний) способы шифрования и расшифрования файлов с помощью функций Encrypt и Decrypt программы PGPtools. Включить в отчет о лабораторной работе *сведения о порядке шифрования и расшифрования файлов в системе PGP*, включить в электронную версию отчета копии используемых при этом экранных форм и ответы на вопросы:

- **какие дополнительные параметры шифрования могут быть использованы, в чем их смысл и возможное применение(обязательно**
- **проверить на примере и результаты проверки отразить в отчете);**
- **как генерируется, как и где хранится ключ симметричного шифрования файла в системе PGP;**
- **как может быть обеспечен доступ к зашифрованному файлу со стороны других пользователей;**
- **изменяется ли и как размер файла после его шифрования (привести конкретные примеры для разных типов файлов).**

5. Изучить (на примере документов из своей папки) способы получения и проверки электронной цифровой подписи под файлами с помощью функций Sign и Verify программы PGPtools. Включить в отчет *сведения о порядке обеспечения аутентичности и целостности электронных документов в системе PGP*, включить в электронную версию отчета копии используемых при этом экранных форм и ответы на вопросы:

- **какие дополнительные параметры получения электронной цифровой подписи могут быть использованы, в чем их смысл и возможное применение (обязательно проверить на примере и результаты проверки отразить в отчете);**
- **какова реакция программы на нарушение целостности подписанного документа (обязательно проверить на примере и результаты проверки отразить в отчете).**

6. Изучить способы одновременного шифрования (расшифрования) и по-

лучения (проверки) электронной цифровой подписи в системе PGP с помощью функций Encrypt Sign и Decrypt/Verify программы PGPtools. Включить в отчет *сведения о порядке одновременного обеспечения конфиденциальности, аутентичности и целостности электронных документов в этой системе*, а также включить в электронную версию отчета копии используемых при этом экранных форм.

7. Изучить способы надежного удаления файлов с конфиденциальной информацией с помощью функции Wipe программы PGPtools. Включить в отчет *сведения о порядке уничтожения конфиденциальных электронных документов в системе PGP* и включить в электронную версию отчета копии используемых при этом экранных форм.

8. Изучить способы надежного уничтожения остаточной информации, которая может содержать конфиденциальные сведения, с помощью функции Freespace Wipe программы PGPtools. Включить в отчет *сведения о назначении и порядке использования этой функции программы*, копии используемых в ней экранных форм и ответы на вопросы:

- **как достигается надежное уничтожение остаточной конфиденциальной информации в системе PGP;**
- **является ли подобный метод уничтожения абсолютно надежным и, если нет, как может быть обеспечено абсолютно надежное уничтожение остаточной информации**

9. Изучить способы быстрого выполнения функций системы PGP с помощью программы PGPtray, ярлык которой размещен в правой части панели задач. Включить в отчет *сведения о назначении и порядке использования этой программы*, а также включить в электронную версию отчета копии используемых экранных форм.

10. Изучить способы управления настройками системы PGP при ее использовании в организациях с помощью программы PGPadmin (пройти все шаги диалога с мастером вплоть до последнего, на котором вместо кнопки «Save»

нажать кнопку «Отмена»). Включить в отчет *сведения о возможностях и порядке администрирования системы PGP*, включить в электронную версию отчета копии используемых при этом экранных форм и ответы на вопросы:

- **какие функции по управлению шифрованием и обеспечением целостности информационных ресурсов предоставляет администратору программа PGPAdmin;**

- **какие функции по управлению криптографическими ключами пользователей PGP предоставляет администратору программа PGPadmin;**

- **какие возможности предоставляет программа PGPadmin по управлению доступными для пользователей функциями программы PGP и где сохраняется подобная информация.**

11. Включить в отчет о лабораторной работе ответы на контрольные вопросы, выбранные в соответствии с номером варианта, указанным преподавателем.

Номер варианта	Контрольные вопросы
1,5,7	Как выбрать длину криптографического ключа в системе PGP?
2,4,6,8	В чем достоинства и недостатки криптографических методов защиты информации?
11,13,15,17,19	Какие компьютерные системы называются безопасными?
12,14,16	В чем заключаются основные требования к защищенности компьютерных систем?
20,22,24,30	Для выполнения каких требований к защищенности компьютерных систем могут применяться криптографические методы защиты?
21,23,25	Насколько, на Ваш взгляд, надежны методы криптографической защиты информации, используемые в программе PGP?

3,9,18, 28	Какими основными функциями защиты информации обладает программа PGP?
10,26,27, 29	Какой принцип лежит в основе функции надежного уничтожения остаточной конфиденциальной информации программы PGP?

Список литературы

1. Аграновский А. В., Хади Р. А. Практическая криптография. Алгоритмы и их программирование (+ CD-ROM). СПб, СОЛОН-Пресс, 2002.
2. Вельшенбах М. Криптография на Си и С++ в действии (+CD-ROM). – М.: Триумф, 2004.
3. Башлы П.Н., Бабаш А.В., Баранова Е.К. Информационная безопасность: учебно- практическое пособие. – М.: Изд. Центр ЕАОИ, 2010.
4. Коутинхо С. Введение в теорию чисел. Алгоритм RSA. – М.: Постмаркет, 2001.
5. Мао В. Современная криптография: теория и практика. – М.: Издательский дом «Вильямс», 2005.
6. Молдовян А. А., Молдовян Н. А., Советов Б. Я. Криптография. Учебник для вузов – М.: Лань, 2005.
7. Фомичев В.М. Дискретная математика и криптология. Курс лекций. – М.: Диалог- МИФИ, 2003.

ЛАБОРАТОРНАЯ РАБОТА № 9

КОРРЕКТИРУЮЩИЕ КОДЫ. КОДЫ ХЕММИНГА

Цель работы: Ознакомление с общими принципами построения и использования корректирующих кодов для контроля целостности информации, распространяемой по телекоммуникационным каналам.

Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл

Hemming.exe, который находится в архиве *Код Хэмминга.rar*.

1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

1.1. Способы контроля правильности передачи данных

1.1.1. Код с проверкой на четность

Контроль целостности информации при передаче от источника к приемнику может осуществляться с использованием корректирующих кодов.

Простейший корректирующий код – *код с проверкой на четность*, который образуется добавлением к группе информационных разрядов одного избыточного, значение которого выбирается таким образом, чтобы сумма единиц в кодовой комбинации (то есть вес кодовой комбинации) была всегда четна.

Пример. Рассмотрим код с проверкой на четность, образованный добавлением контрольного разряда к простому двоичному коду.

	<i>Информационный</i>	<i>Контрольный</i>
0	000	0
1	001	1
2	010	1
3	011	0
4	100	1
5	101	0
6	110	0
7	111	1

Такой код обнаруживает все одиночные ошибки и групповые ошибки нечетной кратности, так как четность количества единиц в этом случае будет также нарушаться.

Следует отметить, что при кодировании целесообразно число единиц в кодовой комбинации делать нечетным, и осуществлять контроль на нечетность, в этом случае любая комбинация, в том числе и изображающая нуль, будет иметь хотя бы одну единицу, что дает возможность отличить полное отсутствие информации от передачи нуля.

1.1.2. Коды Хэмминга

N - значный код Хэмминга имеет m - информационных разрядов и k - контрольных. Число контрольных разрядов должно удовлетворять соотношению:

$$k \geq \log_2 (n + 1),$$

откуда
$$m \leq n - \log_2 (n + 1).$$

Код Хэмминга строится таким образом, что к имеющимся информационным разрядам кодовой комбинации добавляется вычисленное по вышеприведенной формуле количество контрольных разрядов, которые формируются путем подсчета четности суммы единиц для определенных групп информационных разрядов. При приеме такой кодовой комбинации из полученных информационных и контрольных разрядов путем аналогичных подсчетов четности составляют корректирующее число, которое равно нулю, при отсутствии ошибки, либо указывает номер ошибочного разряда.

Рассмотрим подробнее процесс кодирования.

Пусть первый контрольный разряд имеет нечетный порядковый номер, установим его при кодировании таким образом, чтобы сумма единиц всех разрядов с нечетными порядковыми номерами была равна 0. Такая операция может быть записана в виде соотношения:

$$E_1 = a_1 \oplus a_3 \oplus a_5 \oplus a_7 \dots = 0,$$

где a_1, a_3, a_5, a_7 - двоичные символы, размещенные в разрядах с номерами 1, 3, 5, 7,...

Появление единицы во втором разряде (справа) корректирующего числа означает ошибку в тех разрядах кодовой комбинации, порядковые номера которых (2, 3, 6, 7, ...) в двоичном изображении имеют единицу во втором справа разряде. Поэтому вторая операция кодирования, позволяющая найти второй контрольный разряд, имеет вид:

$$E_2 = a_2 \oplus a_3 \oplus a_6 \oplus a_7 \dots = 0,$$

Рассуждая аналогичным образом:

$$E_3 = a_4 \oplus a_5 \oplus a_6 \oplus a_7 \dots = 0, E_4 = a_8 \oplus a_9 \oplus a_{10} \oplus a_{11} \dots = 0,$$

После приема кодовой комбинации, совместно со сформированными контрольными разрядами, выполняются те же операции подсчета, что были описаны выше, а полученное число:

$$E_k E_{k-1} \dots E_2 E_1$$

считается корректирующим, причем при $E_k E_{k-1} \dots E_2 E_1 = 0$ ошибки отсутствуют, а при наличии ошибок неравными нулю оказываются те суммы E_i , в образовании которых участвовал ошибочный разряд.

Корректирующее число при этом будет равно порядковому номеру этого разряда.

Выбор места для контрольных разрядов в каждой из кодовых комбинаций определяется таким образом, чтобы контрольные разряды участвовали только в одной операции подсчета четности. Это упрощает процесс кодирования, такими позициями являются целые степени двойки: 1, 2, 4, 8, 16,...

Пример. Составим шестизначный код Хэмминга

$$n = 6, k \geq \log_2 7, k = 3, \quad m = n - k = 3$$

фра	Ци	Простой двоичный код	Код Хэмминга					
			6	5	4	3	2	1
0		000	0	0	0	0	0	0
1		001	0	0	0	1	1	1
2		010	0	1	1	0	0	0
3		011	0	1	1	1	1	1
4		100	1	0	1	0	1	1
5		101	1	0	1	1	0	0
6		110	1	1	0	0	1	1
7		111	1	1	0	1	0	0

Варианты исправления ошибок

Принят код: 111100 исправлено 110100 - ошибка по корректирующему числу в разряде 4;

111010 исправлено 101010 - ошибка по корректирующему числу в разряде 5;

100000 исправлено 000000 - ошибка по корректирующему числу в разряде 6

1.1.3. Циклические коды

Циклические коды являются разновидностью систематических кодов и поэтому обладают всеми их свойствами. Характерной особенностью циклического кода, определяющей его название, является то, что если n -значная кодовая комбинация $a_0 a_1 a_2 \dots a_{n-1} a_n$ принадлежит данному коду, то и комбинация $a_n a_0 a_1 a_2 \dots a_{n-1}$, полученная циклической перестановкой знаков, также принадлежит этому коду.

Идея построения циклических кодов базируется на использовании *неприводимых многочленов*. Неприводимым называется многочлен, который не может быть представлен в виде произведения многочленов низших степеней, то есть такой многочлен, который делится только на самого себя или на 1.

Неприводимые многочлены при построении циклических кодов играют роль так называемых образующих полиномов, от вида которых, собственно и зависят основные характеристики полученного кода: избыточность и корректирующая способность. В таблице 1.1 указаны неприводимые многочлены со степенями $k=1, 2, 3, 4$.

Таблица 1.1

$P(x)$		$P(1,0)$
$k=1$	$x+1$	11
$k=2$	x^2+x+1	111
$k=3$	x^3+x+1	1011
x^3+x^2+1		1101
$k=4$	x^4+x+1	10011
x^4+x^3+1		11001
$x^4+x^3+x^2+x+1$		11111

Основные принципы кодирования в циклическом коде заключаются в следующем. Двоично-кодированное n -разрядное число представляется полиномом $(n-1)$ -й степени некоторой переменной x , причем коэффициентами полинома являются двоичные знаки соответствующих разрядов. Запись, чтение и передача кодовых комбинаций в циклическом коде производятся, начиная со старшего разряда. В соответствии с этим правилом, в дальнейшем сами числа и соответствующие им полиномы будем записывать так, чтобы старший разряд оказывался справа.

Пример. Число 110101 (нумерация разрядов согласно выше приведенному правилу, ведется слева направо от 0 до 5) будет представлено полиномом пятой степени: $1 + x + x^3 + x^5$.

Следует отметить, что циклическая перестановка разрядов в двоичном представлении числа соответствует умножению полинома на x , при котором x^n заменяется 1 и переходит в начало полинома.

Пример. Произведем умножение полинома, полученного в предыдущем примере, на x . Новый полином $x + x^2 + x^4 + x^6$, преобразуем, заменив x^6 на 1. Окончательно получим $1 + x + x^2 + x^4$, что соответствует числу 111010.

Циклический код n -значного числа, как всякий систематический код, состоит из m информационных и k контрольных знаков, причем последние занимают k младших разрядов. Так как последовательная передача кодовых комбинаций производится, как мы уже указывали, начиная со старших разрядов, контрольные знаки передаются в конце кода.

Как уже указывалось, образование кода производится при помощи так называемого порождающего полинома, $P(x)$, степени k , видом которого определяются основные свойства кода – избыточность и корректирующая способность.

Кодовым полиномом, $F(x)$, является полином степени, меньшей $(m+k)$, если он делится без остатка на порождающий полином $P(x)$. После передачи сообщения, декодирование состоит в выполнении деления полинома $H(x)$, соответствующего принятому коду, на $P(x)$. При отсутствии ошибок $H(x)=F(x)$, и деление выполняется без остатка. Наличие ненулевого остатка указывает на то, что при передаче или хранении произошли искажения информации.

Для получения систематического циклического кода используется следующее соотношение:

$$F(x) = x^k G(x) \square R(x),$$

где $G(x)$ – полином, представляющий информационные символы (информационный полином);

$R(x)$ – остаток от деления $x^k G(x)$ на $P(x)$.

Пример. Рассмотрим кодирование 8-значного числа 10110111. Пусть для кодирования задан порождающий полином 3-й степени $P(x) = 1 + x + x^3$.

Делим $x^3 G(x)$ на $P(x)$: $G(x) = 1 + x^2 + x^3 + x^5 + x^6 + x^7$;
 $x^3 G(x) = x^3 + x^5 + x^6 + x^8 + x^9 + x^{10}$;

$$\begin{array}{r}
 x^{10} + x^9 + x^8 + x^6 + x^5 + x^3 \quad x^{10} + x^8 + x^7 \\
 \underline{\phantom{x^{10} + x^9 + x^8 + x^6 + x^5 + x^3} x^9 + x^7 + x^6 + x^5 + x^3} \quad x^9 + x^7 + x^6 \\
 \phantom{x^{10} + x^9 + x^8 + x^6 + x^5 + x^3} x^5 + x^3 \\
 \phantom{x^{10} + x^9 + x^8 + x^6 + x^5 + x^3} \underline{x^5 + x^3 + x^2} \quad R(x) = x^2
 \end{array}
 \quad
 \begin{array}{r}
 1 + x + x^3 \\
 \hline
 x^7 + x^6 + x^2
 \end{array}$$

Используя соотношение для получения систематического циклического кода, находим $F(x)$:

$$F(x) = (x^3 + x^5 + x^6 + x^8 + x^9 + x^{10}) \square x^2.$$

Таким образом, окончательно кодовая комбинация, соответствующая $F(x)$, имеет вид

$$\begin{array}{r}
 00010110111 \\
 001 \\
 \hline
 \text{контрольные разряды} \longrightarrow \boxed{001} 10110111
 \end{array}$$

Практически, применяемая процедура кодирования еще более проста, так как нас интересует только остаток, а частное в конечном результате не используется, то можно производить последовательное вычитание по модулю 2 делителя из делимого и полученных разностей до тех пор, пока разность не будет иметь более низкую степень, чем делитель. Эта разность и есть остаток. Такой алгоритм может быть реализован аппаратно при помощи k -разрядного сдвигающего регистра, имеющего обратные связи. Очевидно, что полученный таким способом циклический код будет являться систематическим.

Однако существует и второй вариант получения циклического кода, когда очередная кодовая комбинация получается путем умножения кодовой комбинации $C(x)$ простого n -значного кода на образующий полином $P(x)$.

При втором способе образования циклических кодов информационные и контрольные символы в комбинациях циклического кода не отделены друг от друга, что затрудняет процесс декодирования. Поэтому этот способ кодирования применяется реже, чем первый. Рассмотрим пример кодирования с использованием второго варианта, причем при выполнении операций будем использовать непосредственно записи исходных кодовых комбинаций в двоичном виде.

Пример. Дан порождающий полином вида $P(1,0)=1101$. Требуется построить циклический код из простого четырехзначного кода вторым способом. В качестве примера для построения используем исходную комбинацию $C(1,0)=0011$. Операция умножения этой комбинации на образующий полином запишется следующим образом:

$$\begin{array}{r}
 0011 \\
 1101 \\
 \hline
 0011 \\
 \oplus 0011 \\
 0011 \\
 \hline
 0010111
 \end{array}$$

— это и есть циклический код для 0011.

Аналогичным образом получены остальные кодовые комбинации несистематического циклического кода, приведенные в таблице 1.2.

Таблица 1.2

<i>Простой четырехсимвольный код $C(x)$</i>	<i>Циклический (7,4) – код $C(x)$ $P(x)$, где $P(1,0)=1101$</i>
0000	0000000
0001	0001101
0010	0011010
0011 *	0010111 *
0100	0110100
0101	0111001
0110	0101110
0111	0100011
1000	1101000
1001	1100101
1010	1110010
1011	1111111
1100	1011100
1101	1010001
1110	1000110
1111	1001011

Проблема обнаружения различного типа ошибок при помощи циклического кода, как уже указывалось, сводится к нахождению нужного порождающего полинома. В наши задачи не входит рассмотрение этого вопроса, достаточно полно основные принципы подбора порождающих полиномов изложены в учебнике.

Эффективное кодирование информации

1.1.4. Общие положения

Напомним, что *кодирование* – это представление сообщений в форме, удобной для передачи по данному каналу, а *декодирование* – восстановление информации по принятому сигналу. Одним из важнейших вопросов кодирования является повышение его эффективности, то есть путем устранения избыточ-

ности снижение среднего числа символов, требующихся на букву сообщения. Такое кодирование получило название *эффективное кодирование*. Из выше сказанного следует, что эффективное кодирование решает задачу максимального *сжатия информации*.

Учитывая статистические свойства источника информации можно минимизировать среднее число двоичных символов, требующихся для выражения одной буквы сообщения, что при отсутствии помех позволяет уменьшить время передачи сообщения и его объем.

Эффективное кодирование базируется на *основной теореме Шеннона для канала без помех*⁴, суть которой сводится к следующему.

³ Вернер М. Основы кодирования. Учебник для ВУЗов. – М.: Техносфера, 2006. – 288 с.

⁴ Шеннон К. Математическая теория связи // К.Шеннон. Работы по теории информации и кибернетике: пер.с англ.; под ред. Р.Л.Добрушина и О.Б.Лупанова. – М.: ИЛ, 1963. – 512 с.

Суть теоремы Шеннона

Сообщения, составленные из букв некоторого алфавита, можно закодировать так, что среднее число двоичных символов на букву сколь угодно близко к энтропии источника этих сообщений, но не меньше этой величины.

Наличие в сообщениях избыточности позволяет ставить вопрос о сжатии данных, то есть о передаче того же количества информации с помощью последовательностей символов меньшей длины – *методы сжатия без потерь*. Для этого используются специальные алгоритмы сжатия, уменьшающие избыточность. Эффект сжатия оценивают *коэффициентом сжатия*

$$K = n/q,$$

где n – число минимально необходимых символов для передачи сообщения;

q – число символов в сообщении, сжатом данным методом.

Так, при эффективном двоичном кодировании n равно энтропии источника информации.

Наряду с методами сжатия, не уменьшающими количество информации в сообщении, применяются методы сжатия, основанные на потере малосущественной информации – *методы сжатия с потерями*. Заметим, что применение методов сжатия с потерями неприемлемо для некоторых видов информации, например, текстовой или числовой.



Рис. 1.1. Методы сжатия информации

Простейший способ сжатия числовой информации, представленной в коде *ASCII*, заключается в использовании сокращенного кода с четырьмя битами на символ вместо восьми, так как в *ASCII*-кодировке цифры, а также символы "точка", "запятая" и "пробел" в качестве первого кодового символа имеют нуль, который может быть отброшен.

Среди широко используемых из-за своей простоты алгоритмов сжатия наиболее известен алгоритм *RLE (Run Length Encoding)*, где вместо передачи цепочки из одинаковых символов передаются символ и значение длины цепочки. Этот метод эффективен при передаче растровых изображений, особенно монохромных, где имеется большое число цепочек из 1 и 0, но малополезен при передаче текста. Алгоритм *RLE* реализован в формате *PCX*

Пример использования алгоритма RLE

Исходный текст (9 байт): 77 77 77 77 11 11 11 01 FF

В исходной последовательности:

1. Выделяется повторяющаяся последовательность байт
2. Заменяется счетчиком повторов (04) и кодирующим байтом (77)
3. 00 – нет повторений
4. 02 - сколько байт не повторяется
5. Какие байты не повторяются (01 FF)

Результат сжатия (8 байт): 04 77 03 11 00 02 01 FF

Недостатками алгоритма *RLE* являются:

- низкая степень сжатия;
- сильная зависимость от количества повторов байт в исходной информационной последовательности.

Достоинство – простота реализации.

Основная сфера использования методов сжатия с потерями – графические, аудио и видео файлы, где имеется возможность, частично пожертвовав качественными характеристиками информации, добиться значительного уменьшения ее объемов.

Наибольшее распространение среди методов сжатия информации без потерь нашли статистические алгоритмы сжатия, учитывающие вероятность появления отдельных символов в информационном потоке. В первую очередь, это алгоритмы Шеннона-Фано и Хаффмена.

1.1.5. Алгоритм Шеннона-Фано

Эффективное кодирование методом Шеннона-Фано базируется на основной теореме Шеннона для канала без помех.

Алгоритм эффективного кодирования по Шеннону-Фано

- Буквы алфавита сообщений выписываются в таблицу в порядке убывания вероятностей.

- Затем они разделяются на две группы так, чтобы суммы вероятностей в каждой группе были по возможности одинаковы.
- Всем буквам верхней половины в качестве первого символа приписывается 1, а всем нижним – 0.
- Каждая из полученных групп в свою очередь разбивается на две подгруппы с одинаковыми суммарными вероятностями и т. д., процесс повторяется до тех пор, пока в каждой подгруппе останется по одной букве.

Пример. Рассмотрим алфавит из 7 букв, приписав каждой букве вероятность ее появления в сообщении p_i (существуют специальные таблицы вероятности появления букв русского или латинского алфавитов в сообщениях, например, см. приложение 1.2.1).

Буква	Вероятность, p_i	Процесс получения кода	Код Шеннона
A	1/4	1 $\xrightarrow{\text{II}}$	11
E	1/4	1 $\xrightarrow{\text{I}}$ 0	10
F	1/8	0 1 $\xrightarrow{\text{III}}$	011
C	1/8	0 1 $\xrightarrow{\text{II}}$ 0	010
B	1/8	0 0 $\xrightarrow{\text{III}}$	001
D	1/16	0 0 0 $\xrightarrow{\text{IV}}$	0001
G	1/16	0 0 0 0	0000

$$H(\xi) = \sum_{k=1}^N p_k \log_2 1/p_k = 2/4 + 2/4 + 3/8 + 3/8 + 3/8 + 4/16 + 4/16 = 2,625.$$

Ясно, что при обычном двоичном кодировании, не учитывая статистических характеристик, для представления каждой буквы потребуется три символа. Наибольший эффект сжатия получается в случае, когда вероятности букв представляют собой целочисленные отрицательные степени двойки. Среднее число символов на букву в этом случае точно равно энтропии.

Среднее число символов на букву

$$L = \sum_{i=1}^N p_i n_i,$$

где p_i – вероятность появления i -го символа алфавита;

n_i – количество символов в кодовой комбинации i -го символа алфавита.

Для рассмотренного в таблице примера

$$L = 1/4 * 2 + 1/4 * 2 + 1/8 * 3 + 1/8 * 3 + 1/8 * 3 + 1/16 * 4 + 1/16 * 4 = 2,625.$$

Разность величин $(L - H)$ – есть *избыточность кода*, а величина

$(L - H)/L$ – *относительная избыточность*.

Избыточность может трактоваться как мера бесполезно совершаемых альтернативных выборов. Так как в практических случаях отдельные знаки никогда не встречаются одинаково часто, то кодирование с постоянной длиной кодовых слов в большинстве случаев избыточно. Несмотря на это, такое кодирование применяется достаточно часто, руководствуясь техническими соображениями.

1.1.6. Алгоритм Хаффмена

Алгоритм эффективного кодирования по Хаффмену

Буквы алфавита сообщений выписываются в основной столбец таблицы в порядке убывания вероятностей.

- Две последние буквы объединяются в одну вспомогательную букву, которой приписывается суммарная вероятность.
- Вероятности букв, не участвовавших в объединении, и полученная суммарная вероятность снова располагаются в порядке убывания вероятностей, а две последние объединяются, и т. д. до тех пор, пока не получают единственную вспомогательную букву с вероятностью 1.
- Далее для построения кода используется бинарное дерево, в корне ко-

торого располагается буква с вероятностью 1, при ветвлении ветви с большей вероятностью присваивается код 1, а с меньшей – код 0 (возможно левой – 1, а правой – 0).

Пример. Рассмотрим условный алфавит из восьми букв, каждой из которых приписана соответствующая вероятность ее появления в сообщении.

Буквы	Вероятности	Вспомогательные столбцы вероятностей	Код Хаффмена
Z1	0,22	0,22 0,22 0,26 0,32 0,42 0,58 } 1	01
Z2	0,20	0,20 0,20 0,22 0,26 0,32 } 0,42 }	00
Z3	0,16	0,16 0,16 0,20 0,22 } 0,26 }	110
Z4	0,16	0,16 0,16 0,16 } 0,20 }	111
Z5	0,10	0,10 0,16 } 0,16 }	100
Z6	0,10	0,10 } 0,10 }	1011
Z7	0,04	0,06 }	10101
Z8	0,02		10100

$$L = 0,22 \times 2 + 0,20 \times 2 + 0,16 \times 3 + 0,16 \times 3 + 0,10 \times 3 + 0,10 \times 4 + 0,04 \times 5 + 0,02 \times 5 = 2,8;$$

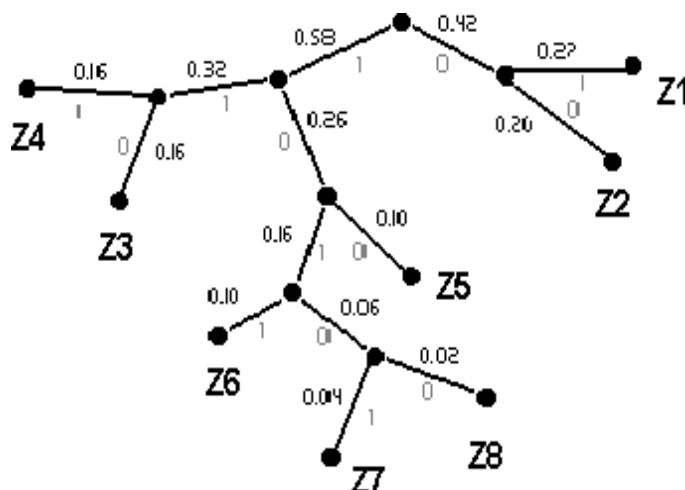
$$H = 2,76;$$

$$L - H = 0,04.$$

Для

сравнения: в коде Шеннона с таким же распределением вероятностей $L - H = 0,08$.

Построение бинарного дерева



В приложении 1.2.2 приводятся задания для самостоятельной работы по кодированию информации с использованием алгоритмов Шеннона-Фано и Хаффмена.

Приложение 1.2.1

Вероятности отдельных букв в русском языке

Буква	Вероятность	Буква	Вероятность
–	0,175	Я	0,018
О	0,090	Ы	0,016
Е	0,072	З	0,016
А	0,062	Ъ,Ь	0,014
И	0,062	Б	0,014
Т	0,053	Г	0,013
Н	0,053	Ч	0,012
С	0,045	Й	0,010
Р	0,040	Х	0,009
В	0,038	Ж	0,007
Л	0,035	Ю	0,006
К	0,028	Ш	0,006
М	0,026	Ц	0,004
Д	0,025	Щ	0,003
П	0,023	Э	0,003
У	0,021	Ф	0,002

Приложение 1.2.2

Задания для самостоятельной работы

1. Написать процедуру, обеспечивающую эффективное кодирование:
 вариант а) код Шеннона; вариант б) код Хаффмена, и подсчет среднего количества информации на знак H ; избыточности $(L - H)$; и относительной избыточности получаемого кода $(L - H) / L$.

Для тестирования программы использовать два контрольных варианта:

Вариант 1

Буквы	Вероятности	Код Шеннона
Z1	0,22	11
Z2	0,20	101
Z3	0,16	100
Z4	0,16	01
Z5	0,10	001
Z6	0,10	0001
Z7	0,04	00001
Z8	0,02	00000 $H = 2,76$ $L = 2,84$ $L - H = 0,08$

Вариант 2

Буквы	Вероятности	Код Шеннона
Z1	1/2	1
Z2	1/4	01
Z3	1/8	001
Z4	1/16	0001
Z5	1/32	00001
Z6	1/64	000001
Z7	1/128	0000001
Z8	1/128	0000000 = 1 63/64 L - H = 0

2. Источник сообщений порождает знаки A, B, C с вероятностями 0,7; 0,2; 0,1.

Написать процедуру позволяющую:

- *определить энтропию источника;*
- *указать для этих трех знаков оптимальное бинарное кодирование:*

вариант а) код Шеннона;

вариант б) код Хаффмена,

и определить среднюю длину кодовых комбинаций;

- *закодировать все пары AA, AB, ...;*
- *построить для этих девяти пар оптимальный бинарный код:*

вариант а) код Шеннона;

вариант б) код Хаффмена.

увеличить блочность кода до трехсимвольных комбинаций, и построить оптимальный бинарный код:

вариант а) код Шеннона;

вариант б) код Хаффмена.

Сделать вывод об изменении избыточности кода с увеличением блочности. Как это влияет на эффективности кода?

Буква	Вероятность	Код	Знак	Вероятность	Код пары
А	0,7	0	АА	0,49	1
В	0,2	11	АВ	0,14	011
С	0,1	10	ВА	0,14	010
			АС	0,07	0011
			СА	0,07	0010
			ВВ	0,04	0001
			ВС	0,02	00001
			СВ	0,02	000001
			СС	0,01	000000

$$H = 1,1571;$$

$$L = 1,3;$$

$$L_{\text{знак}} = 2,33;$$

$$L_{\text{символ}} = L_{\text{знак}} / 2 = 1,165.$$

Вывод: при кодировании большими блоками удастся уменьшить избыточность кода, теоретически минимум избыточности может быть достигнут при кодировании блоков, включающих бесконечное число букв:

1.2.3. LZW – сжатие

Одним из наиболее широко используемых в настоящее время методов сжатия без потерь является алгоритм Лемпела-Зива-Уэлча (*Lempel-Ziv-Welch*).

Метод сжатия был предложен в 1977 году, усовершенствованный (*Terry Welch*) вариант был опубликован в 1984 году, и в дальнейшем неоднократно модернизировался. Этот алгоритм используется, в частности, стандартной процедурой *UNIX Compress*. Алгоритмы группы *LZ* (*LZ77*, *LZ78*, *LZSS*), лежат в основе почти всех современных программных и аппаратных средств сжатия информации. *PKZIP*, *LHA*, *Stacker*, *SuperStor*, *Dblspace* и многие другие архиваторы используют ту или иную модификацию алгоритма *LZ*.

Идея сжатия, предложенная Лемпелом и Зивом заключается в следующем: если в тексте сообщения появляется последовательность из двух ранее уже встречавшихся символов, то эта последовательность объявляется новым символом, для нее назначается код, который при определенных условиях может быть значительно короче исходной последовательности. В дальнейшем, в сжатом сообщении вместо исходной последовательности записывается назначенный код.

При декодировании повторяются аналогичные действия и потому становятся известными последовательности символов для каждого кода.

Процесс кодирования

1. Каждому символу исходного алфавита присваивается определенный код (здесь код – порядковый номер, начиная с 0).

2. Выбирается первый символ сообщения и заменяется на его код.

3. Выбираются следующие два символа и заменяются своими кодами. Одновременно этой двухсимвольной комбинации присваивается свой код (обычно это порядковый номер, равный числу уже использованных кодов). Так, если алфавит включает 8 символов, имеющих коды от 000 до 111, то первая двухсимвольная комбинация получит код 1000, следующая – код 1001 и так далее.

4. Выбираются из исходного текста очередные 2, 3, ... N-символьные комбинации до тех пор, пока не образуется еще не встречавшаяся комбинация. Тогда этой комбинации присваивается очередной код, и поскольку совокупность из первых N-1 символов уже встречалась, то она имеет свой код, который и записывается вместо этих N-1 символов. Каждый акт введения нового кода назовем шагом кодирования.

5. Процесс продолжается до исчерпания исходного текста.

Процесс декодирования

При декодировании код первого символа, а затем второго и третьего заменяются на символы алфавита. При этом становится известным код комбинации второго и третьего символов. В следующей позиции могут быть только коды уже известных символов и их комбинаций. Процесс декодирования продолжается до исчерпания сжатого текста.

Пример.

Пусть исходный текст представляет собой двоичный код (первая строка таблицы 2.3), то есть исходный алфавит $A=\{1,0\}$. Коды этих символов соответственно также 0 и 1.

Образующийся по методу Лемпела-Зива *LZ*-код показан во второй строке таблицы 2.3. В третьей строке отмечены шаги кодирования, после которых происходит переход на представление кодов *A* с увеличенным числом разрядов *r*. Так, на первом шаге вводится код 10 для комбинации 00 и поэтому на следующих двух шагах $r=2$, после третьего шага $r=3$, после седьмого шага $r=4$. В общем случае $r=k$ после шага $2^{k-1}-1$.

В приведенном примере *LZ*-код оказался даже длиннее исходного кода, так как обычно короткие тексты с небольшим количеством повторяющихся символов не дают эффекта сжатия. Эффект сжатия проявляется в достаточно длинных текстах и особенно заметен, например, в графических файлах.

Таблица 1.5

Исходный текст	0.00.000. 01. 11. 111.1111. 110. 0000.00000. 1101. 1110.	
<i>LZ</i> -код	0.00.100.001.0011.1011.1101. 1010.00110.10010.10001.10110.	
<i>r</i>	2 3	4
Вводимые коды	- 10 11 100 101 110 111 1000 1001 1010 1011 1100	

Усовершенствованный алгоритм сжатия и распаковки по методу Лемпела-Зива-Уэлча приведен на рис. 1.2. На рис. 1.3 приведен пример программной реализации алгоритма *LZW*.

Алгоритм компрессии

```
СТРОКА = очередной символ из входного потока
WHILE входной поток не пуст DO
    СИМВОЛ = очередной символ из входного потока
    IF СТРОКА + СИМВОЛ в таблице строк THEN
        СТРОКА = СТРОКА + СИМВОЛ
    ELSE
        вывести в выходной поток код для СТРОКА
        добавить в таблицу строк СТРОКА + СИМВОЛ
        СТРОКА = СИМВОЛ
    END of IF
END of WHILE
вывести в выходной поток код для СТРОКА
```

Алгоритм декомпрессии

```
Читать СТАРЫЙ_КОД
вывести СТАРЫЙ_КОД
WHILE входной поток не пуст DO
    читать НОВЫЙ_КОД
    СТРОКА = перевести НОВЫЙ_КОД
    вывести СТРОКУ
    СИМВОЛ = первый символ СТРОКИ
    добавить в таблицу перевода СТАРЫЙ_КОД + СИМВОЛ
    СТАРЫЙ_КОД = НОВЫЙ_КОД
END of WHILE
```

Рис.1.2. Алгоритм LZW-компрессии и декомпрессии

Введите текст:

///wwwRRRTTET

Код

/ #256 W #258 R #260 T T E T

№	Вход новый код	Старый код	Строка выход	Символ	Новый код таблицы
1	/	/	/		
2	256	/	//	В	//
3	W	256	W	W	//W
4	258	W	WW	!	WW
5	R	258	R	R	WWR
6	260	R	RR	!	RR
7	T	260	T	T	RRT
8	T	T	T	T	TT
9	E	T	E	E	TE
10	T	E	T	T	ET

Рис.1.3. Пример программной реализации алгоритма

1. Описание лабораторной работы

Программа предназначена для кодирования символов по алгоритму Хэмминга. Главное окно программы представлено на рис.9.1.

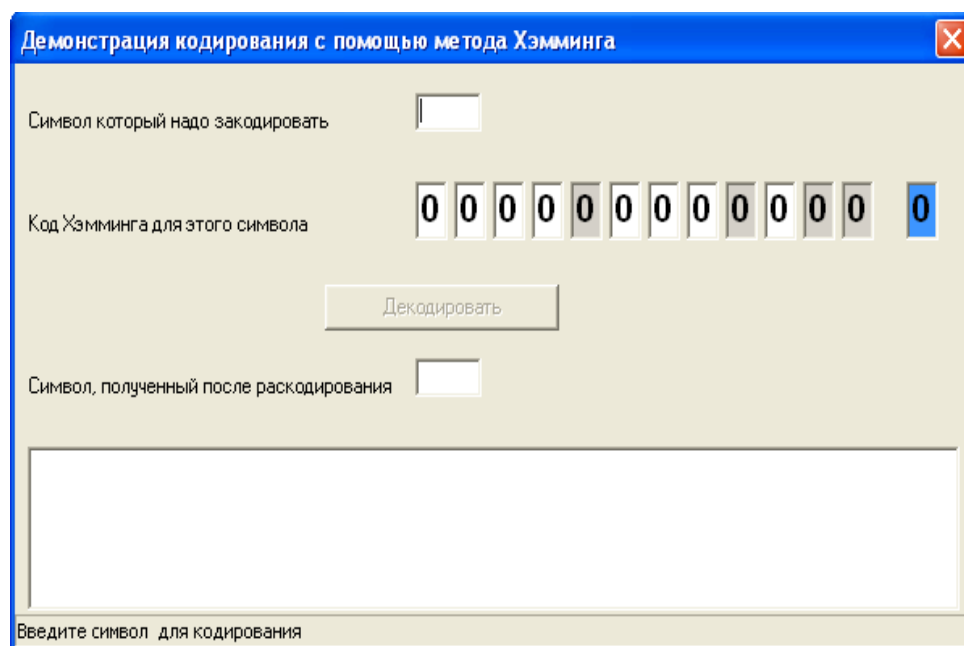


Рис. 9. 1. Главное окно программы

Кодируются любые символы *ASCII*-кодировки, с помощью тринадцатисимвольного кода Хэмминга, содержащего 8 информационных разрядов, 4 контрольных разряда и разряд общей четности.

Процессы кодирования и декодирования представлены на рис.9.2

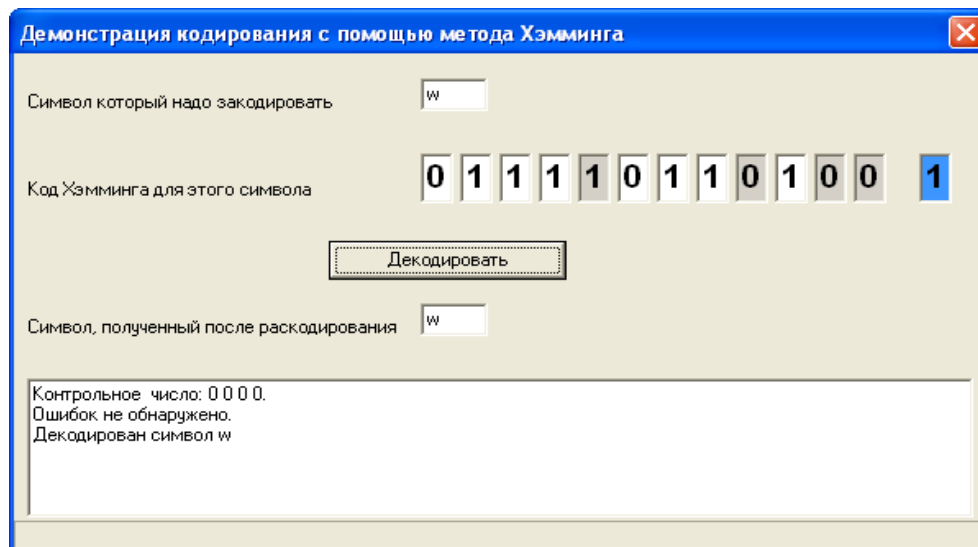


Рис. 9. 2. Процесс кодирования и декодирования

Серым цветом отмечены контрольные биты, голубым – бит общей четности. В полученный код можно вносить ошибки. Одиночные ошибки могут быть исправлены, двойные – обнаруживаются без исправления. Если ошибка была исправлена, то указывается, в каком бите она была допущена. Затем символ декодируется. Процесс исправления одиночной ошибки представлен на рис. 9.3.

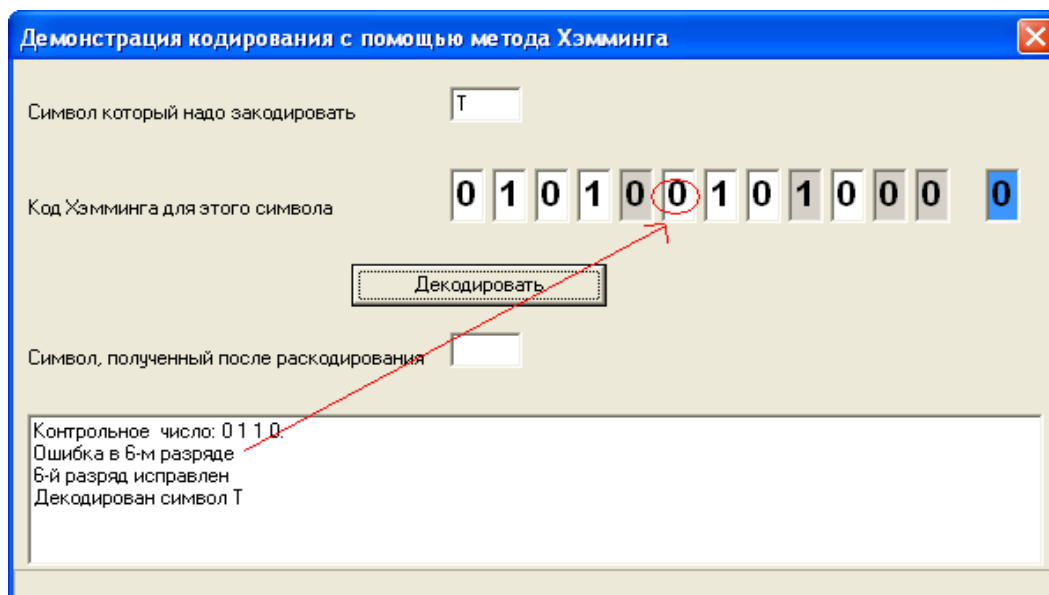


Рис. 9. 3. Процесс исправления одиночной ошибки

Если возникает ошибка двойной или более кратности – выводится сообщение о невозможности исправления кода.

2. Задание

1. Закодировать с помощью кода Хэмминга предложенный алфавит (варианты указаны в Таблице 9.1).

2. В каждую строку таблицы с закодированной информацией внести единичную ошибку, зафиксировать в кодовой таблице результат декодирования.

3. В последние две строки таблицы с закодированной информацией внести двойные ошибки, зафиксировать в кодовой таблице результат декодирования.

4. Проанализировать полученные результаты и сформулировать аргументированные выводы.

Описать полученный код Хэмминга :

- количество контрольных и информационных разрядов и их номера;
- избыточность кода;
- относительная избыточность;
- минимальное кодовое расстояние;
- оценить корректирующую способность полученного кода.

5. Составить из предложенного алфавита слово длиной не менее 5 символов и закодировать его с помощью полученного кода Хэмминга.

Подсчитать длину исходного текста (кодировка *ASCII*) и закодированного текста (код Хэмминга).

6. Представить краткий теоретический материал, в котором описать способ кодирования и декодирования информации с помощью кода Хэмминга.

7. Привести итоги проведенных экспериментов по кодированию с помощью программы *Hemming.exe*.

8. Оценить результаты обнаружения и исправления одиночных и обнаружения двойных ошибок. Сделать выводы о корректирующей способности иссле-

дуемого кода.

9. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Варианты заданий представлены в Таблице 9.1

Таблица 9.1

Номера вариантов	Исходный алфавит
1,5,9,13,17	Кириллица А...М
2,6,10,14,18,22	Кириллица О..Я
3,7,11,15,19,23,27	Латиница А..N
4,8,12,16,20,24,28	Латиница О..Z
21,25,29,26,30	Десятичные цифры 0..9

Номер варианта	Контрольные вопросы
1,5,7,3,9,18,28	Что представляет собой и как используется код Хэмминга?
2,4,6,8,20,22,24,26,30	Как составляется код Хэмминга?
11,13,15,10,17,19,27	Какие ошибки можно исправить с помощью кода Хэмминга? От каких свойств кода зависит его корректирующая способность?
12,14,16,21,23,25,29	Какие ошибки могут возникнуть при составлении кода Хэмминга?

ЛАБОРАТОРНАЯ РАБОТА № 10

КОРРЕКТИРУЮЩИЕ КОДЫ. ЦИКЛИЧЕСКИЕ КОДЫ

Цель работы: Ознакомление с общими принципами построения и использования корректирующих кодов для контроля целостности информации, распространяемой по телекоммуникационным каналам. Изучение принципов построения циклических кодов.

Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл

CyclicCode74.exe, который находится в архиве *Циклический код.rar*.

1. Описание лабораторной работы

1. Изучить сведения о программе кодирования информации с использованием циклического кода (папка *Документация*). Запустить программу кодирования (папка *Исполняемый модуль*, файл *CyclicCode74.exe*), рис. 10.1.

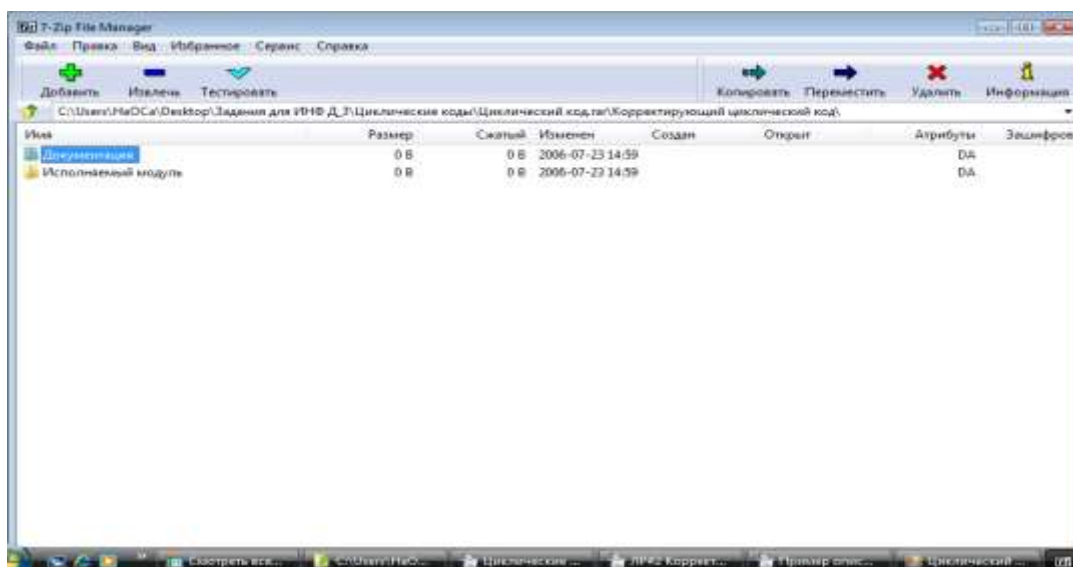


Рис. 10.1. Запуск программы *CyclicCode74*

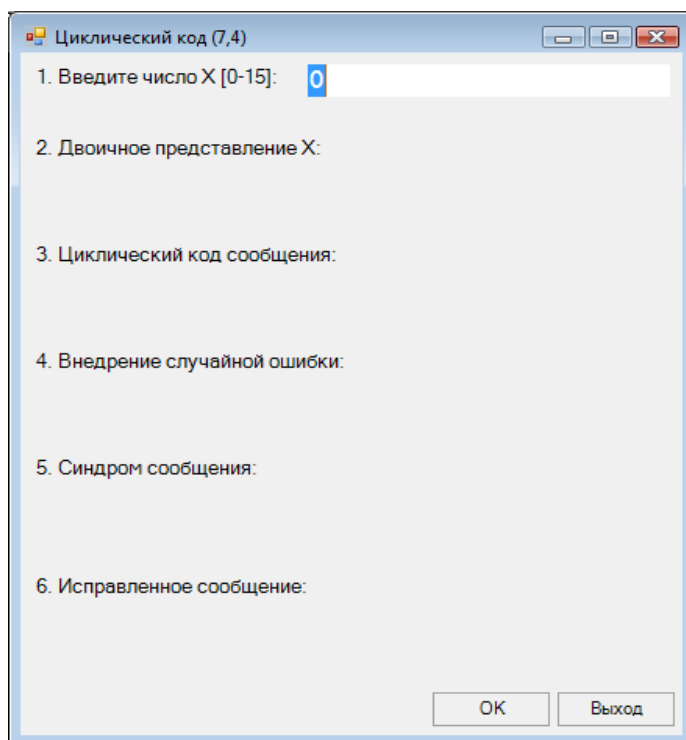


Рис.10. 2. Главное окно программы

2. Закодировать с помощью циклического кода предложенный вариант (см. варианты заданий). Для этого в строку "1. Введите число X [0-15]" вводим заданное число. Кодовую таблицу сохранить и перенести в личную папку (для этого нажимаем одновременно Ctrl+Alt+PrtScSysRq затем Ctrl+V).

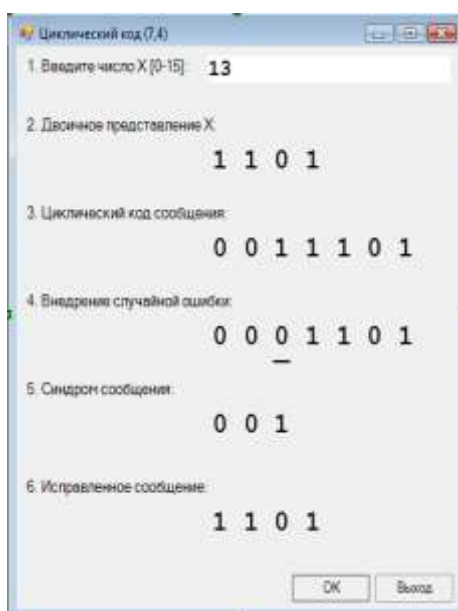


Рис. 10. 3. Пример заполнения полей главного окна программы

3. Проанализировать полученные результаты и сформулировать аргументированные выводы.

2. Задание

1. Получить систематический циклический код, используя приведенные в Таблице 10.1 порождающие полиномы, в соответствии с вариантом, указанным преподавателем:

Таблица 10.1

Вариант	Порождающий полином 3-ей степени
1,7,13,19,25	1011
2,8,14,20,26	1101
3,9,15,21,27	1011
4,10,16,22,28	1101
5,11,17,23,29	1011
6,12,18,24,30	1101

Пример: Рассмотрим кодирование восьмизначного числа 00011111. Пусть для кодирования задан порождающий полином 3-ей степени

$$P(1,0) = 1101, P(x) = 1 + x + x^3$$

Делим $x^3 G(x)$ на $P(x)$, где информационный полином $G(x) = x^3 + x^4 + x^5 + x^6 +$

$$x^7 x^3 G(x) = x^6 + x^7 + x^8 + x^9 + x^{10}$$

$$\begin{array}{r|l} & x^3 + x + 1 \\ \hline x^{10} + x^9 + x^8 + x^7 + x^6 & x^{10} + x^8 + x^7 \\ \hline & x^9 + x^6 \end{array}$$

$$\begin{array}{r} x^9 + x^6 \\ \hline x^9 + x^7 + x^6 \\ \hline \end{array}$$

$$\begin{array}{r} x^7 \\ \hline x^7 + x^5 + x^4 \\ \hline \end{array}$$

$$x^5 + x^4$$

$$\begin{array}{r} \hline x^5 + x^3 + x^2 \end{array}$$

$$x^4 + x^3 + x^2 x^4 + x^2 + x$$

$$\begin{array}{r} \hline x^3 + x \end{array}$$

$$\frac{x^3 + x + 1}{R(x) = 1}$$

Находим значение кодового полинома $F(x)$:

$$F(x) = (x^6 + x^7 + x^8 + x^9 + x^{10}) \square 1$$

Таким образом, окончательно кодовая комбинация

$$\begin{array}{cccccccccccc}
 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
 F(x) \text{ имеет вид :} & & & & & & & & & & & \\
 \text{Контрольные разряды} & \Rightarrow & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
 & & & & & & & & & & 1 & 0 & 0 \\
 & & & & & & & & & & \hline
 & & & & & & & & & & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1
 \end{array}$$

2. Представить в отчете краткий теоретический материал, в котором описать способ кодирования и декодирования информации с помощью циклического кода.

3. Привести итоги проведенных экспериментов по кодированию с помощью программы *CyclicCode74.exe*.

4. Оценить результаты обнаружения и исправления одиночных ошибок. Сделать выводы о корректирующей способности исследуемого кода.

5. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Варианты заданий приведены в Таблице 10.2.

Таблица 10.2

<i>Вариант</i>	<i>Число</i>
1,16	1
2,17	2
3,18	3
4,19	4
5,20	5
6,21	6
7,22	7
8,23	8
9,24	9
10,25	10
11,26	11
12,27	12
13,28	13
14,29	14
15,30	15

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Укажите различия понятий: <i>кодирование информации, шифрование информации.</i>
2,4,6,8, 20,22,24, 26,30	Определите понятия: <i>равномерные коды, неравномерные коды, префиксное кодирование, разделимое кодирование, систематические коды, несистематические коды.</i>
11,13,15, 10,17,19, 27	От каких характеристик кода зависит его корректирующая способность?
12,14,16 21,23,25, 29	Сравните коды Хэмминга и циклические коды по эффективности контроля целостности информации.

ЛАБОРАТОРНАЯ РАБОТА № 11

МЕТОДЫ СЖАТИЯ ПО ШЕННОНУ И ХАФФМЕНУ

Цель работы: Ознакомление со статистическими принципами сжатия информации с использованием методов Шеннона-Фано и Хаффмена.

Примечание. Для выполнения лабораторной работы на компьютере необходимо установить файл *Shannon-Huffman.exe*, который находится в архиве *Методы сжатия по Шеннону и Хаффмену.rar*.

1. Описание лабораторной работы

Работа выполняется на персональном компьютере с использованием программы *Shannon-Huffman.exe*. Программа предназначена для демонстрации методов сжатия информации с использованием алгоритмов Шеннона-Фано и Хаффмена (рис.11.1).

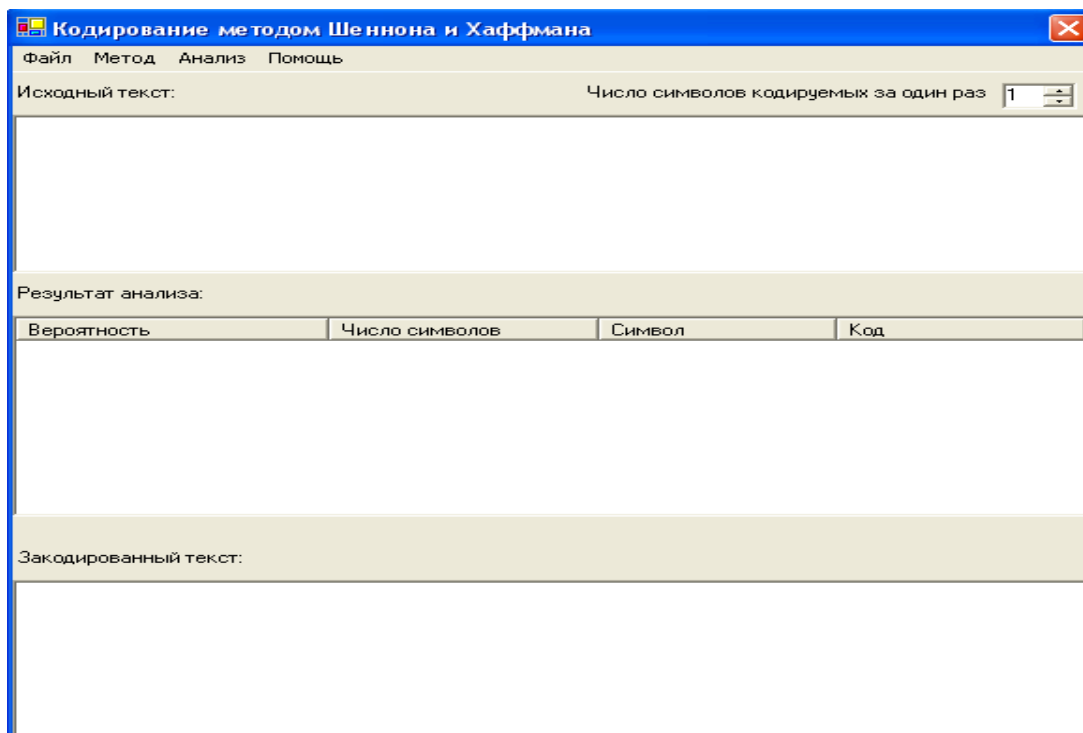


Рис. 11. 1. Главное окно программы

Для работы с программой пользователем выбирается требуемый метод сжатия, рис. 11.2.

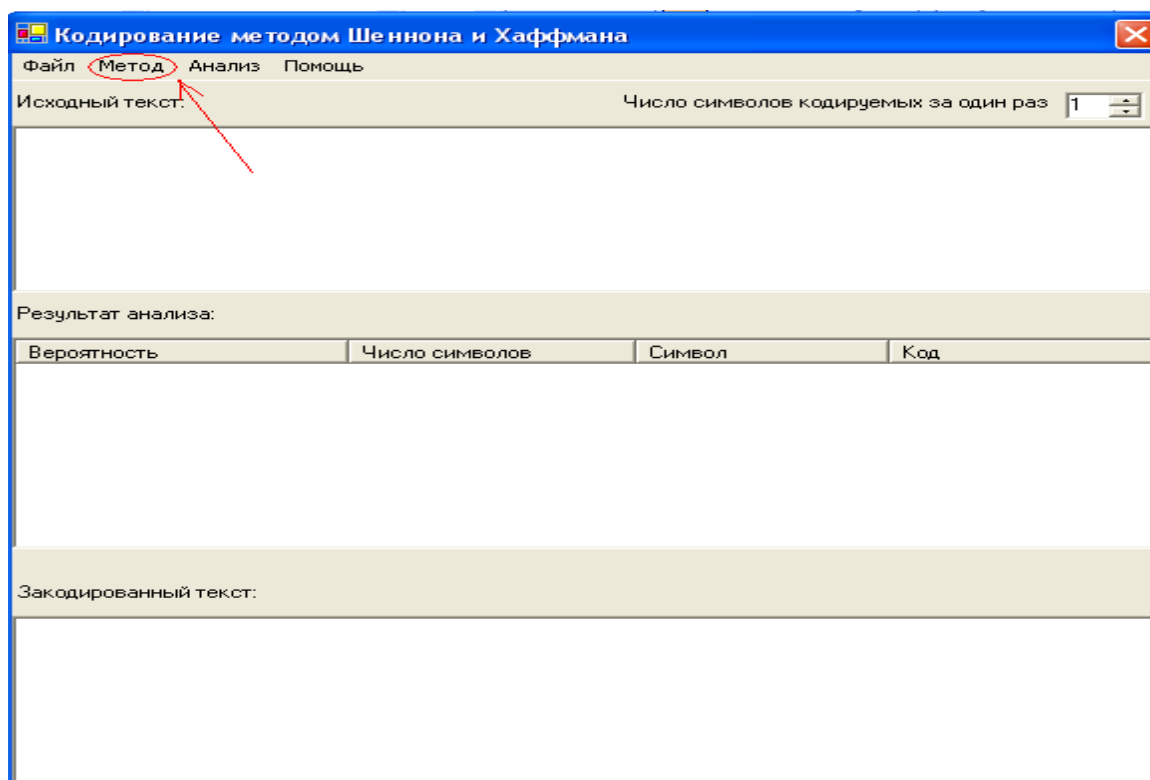
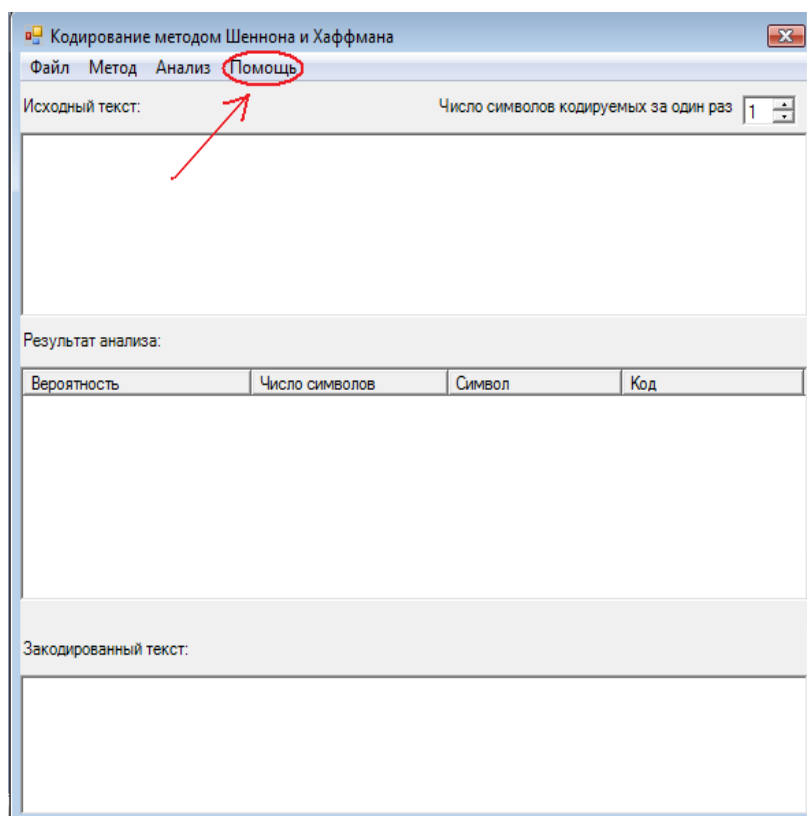


Рис. 11. 2. Выбор метода сжатия

Далее в окне программы *Исходный текст* записывается сообщение произвольной длины (или загружается сообщение из заранее подготовленного файла в формате *.txt*). После этого необходимо указать число символов, кодируемых за один раз.

Для подсчета вероятности появления букв во введенном сообщении P , определения энтропии источника сообщений H , среднего числа символов при кодировании одной буквы сообщения L , необходимо выбрать закладку *Анализ*.



Для определения эффективности кодирования рассчитывается избыточность кода ($L-H$).

Чтобы выйти из данной программы достаточно выбрать закладку *Файл* и далее *Выход*.

Пример 1.

Исходное сообщение *аббвввггггддддеееее*, состоящее из символов шестибуквенного алфавита

$A = \{a, б, в, г, д, е\}$, закодируем, используя метод Хаффмена. Определим вероятность появления символа в исходном сообщении

$$P = n/N,$$

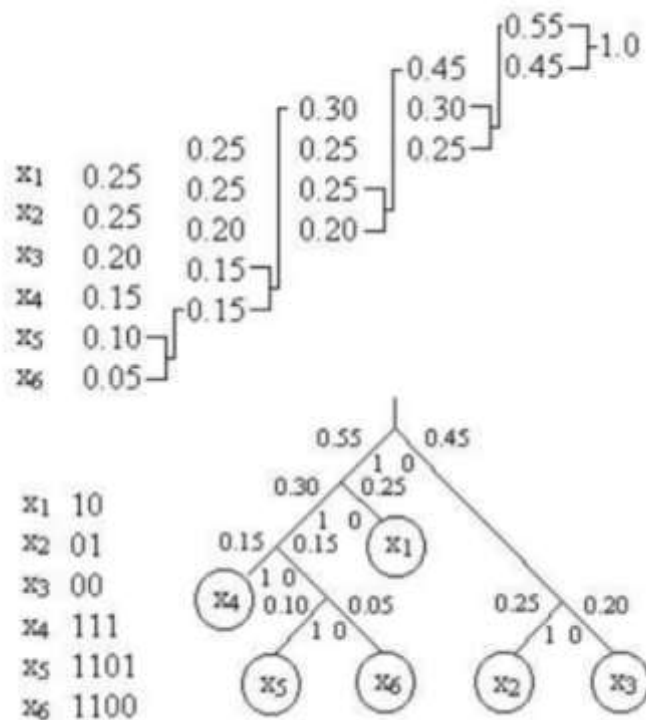
где n – число повторов символа в сообщении, N – длина сообщения.

Выписываем в столбец все символы алфавита в порядке убывания вероятности их появления в тексте (Таблица 11.1).

Таблица 11.1

СИМВОЛ	число повторений в данном сообщении	вероятность
<i>д</i>	5	0,25
<i>е</i>	5	0,25
<i>з</i>	4	0,20
<i>в</i>	3	0,15
<i>б</i>	2	0,10
<i>а</i>	1	0,05

Последовательно объединяя два символа с наименьшими вероятностями появления в новый составной символ, вероятность появления которого полагается равной сумме вероятностей составляющих его символов, в конце концов мы построим дерево, каждый узел которого имеет суммарную вероятность всех узлов, находящихся ниже него. Прослеживаем путь к каждому листу дерева пометчая направление к каждому узлу (например, направо - 0, налево -1).



Среднее количество символов на букву сообщения:

$$L = \sum_{i=1}^n P(i)n(i) = 2*2*0.25 + 2*0.20 + 3*0.15 + 4*(0.10 + 0.05) = 2.45,$$

где

$n(i)$ — количество знаков в кодовой комбинации i -го символа алфавита

$P(i)$ — вероятность появления i -го символа алфавита.

Энтропия:

$$H = \sum_{i=1}^n P(i) \log \frac{1}{P(i)} = 2.425$$

Значение избыточности кода $(L-H) = 0.025$.

Сравниваем полученные данные с результатами работы программы
(рис.11.4)

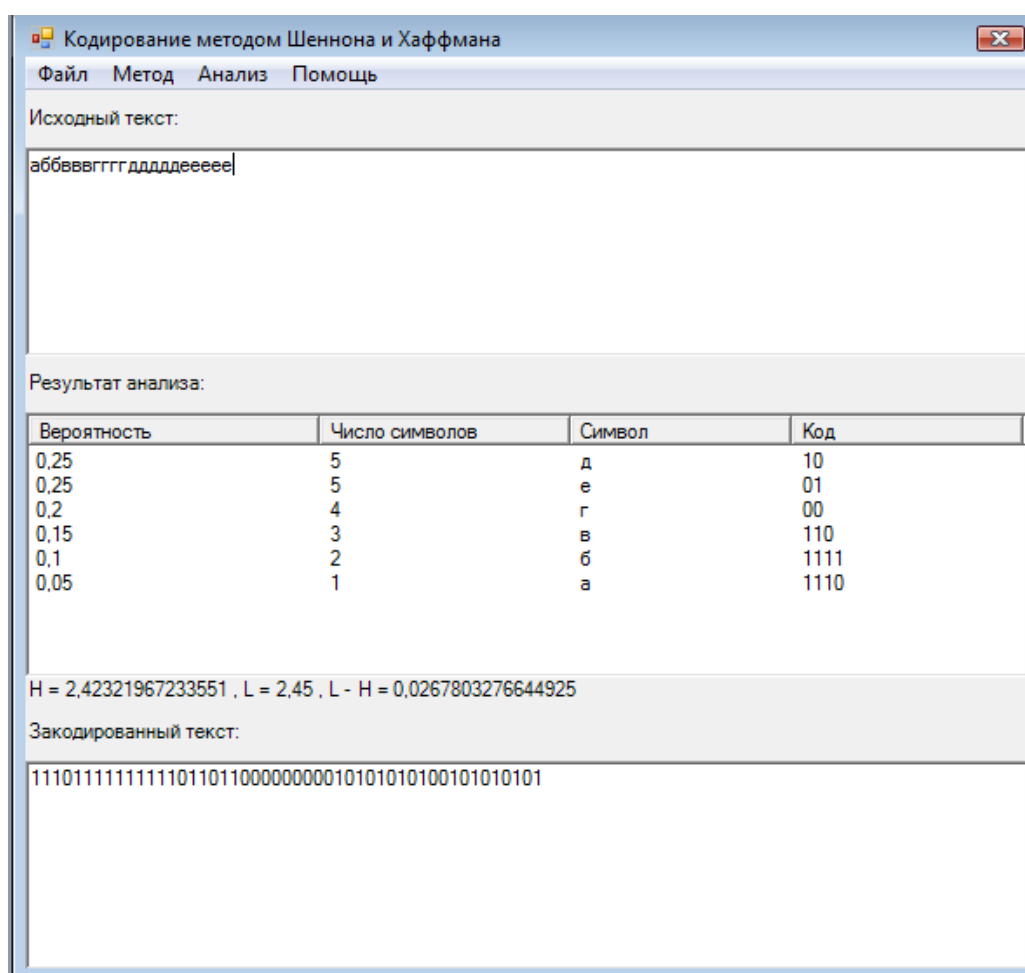


Рис. 11. 4. Пример кодирования

Пример 2.

Закодируем исходное сообщение (см. *Пример 1*), но используя метод Шеннона. Выписываем в столбец все символы в порядке убывания вероятности появления их в тексте;

Последовательно делим множество символов на два подмножества так, чтобы сумма вероятностей появления символов одного подмножества была примерно равна сумме вероятностей появления символов другого. Для нижнего подмножества каждому символу приписываем "0", для верхнего – "1". Дальнейшие разбиения повторяются до тех пор, пока все подмножества не будут состоять из одного элемента (Таблица 11.2).

Таблица 11.2

сим- вол	число повторений в данном сообщении	вероят- ность	Код Шеннона		
<i>o</i>	5	0,25	1	1	
<i>e</i>	5	0,25	1	0	
<i>z</i>	4	0,20	0	1	1
<i>v</i>	3	0,15	0	1	0
<i>b</i>	2	0,10	0	0	1
<i>a</i>	1	0,05	0	0	0

Средняя длина полученного кода будет равна

Среднее количество символов на букву сообщения:

$$L = \sum_{i=1}^n n(i) \cdot P(i) = 0.25 \cdot 2 + 0.25 \cdot 2 + 0.2 \cdot 3 + 0.15 \cdot 3 + 0.1 \cdot 3 + 0.25 \cdot 3 = 2.5$$

Энтропия:

$$H = \sum_{i=1}^n P(i) \log \frac{1}{P(i)} = 2.425$$

Сравниваем полученные данные с результатами работы программы (рис.11.5)

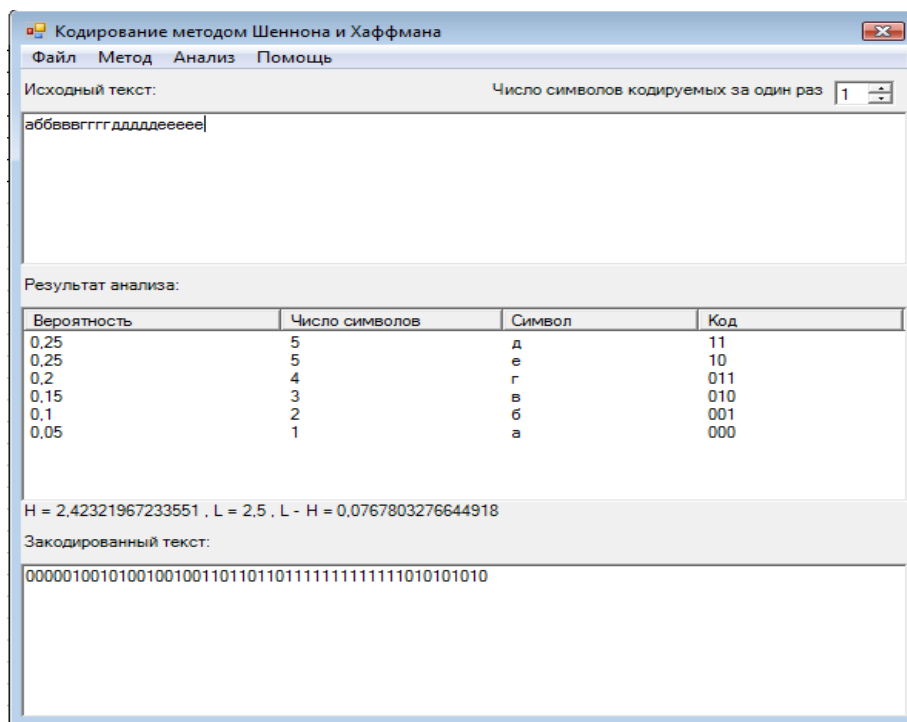


Рис. 11. 5. Пример кодирования

2. Задание

Ознакомиться со сведениями о программе *Shannon-Huffman.exe*, и рассмотреть примеры эффективного кодирования информации.

1. Запустить программу *Shannon-Huffman.exe*, предназначенную для демонстрации методов сжатия информации с использованием алгоритмов Шеннона-Фано и Хаффмана.

2. Выполнить предложенные задания:

1). Число символов алфавита $k = m$ (m – номер варианта).

Составить такое исходное сообщение, чтобы

а) символы алфавита встречались в сообщении с равными вероятностями;

б) символы алфавита встречались в сообщении с разными вероятностями.

2). Ввести произвольный связный текст на русском языке.

Это может быть пословица, стихотворение или произвольный текст. Используя результаты работы программы, следует проанализировать алфавит введенного сообщения: подсчитать количество символов алфавита, значение энтропии H , среднее количество символов на знак L при целочисленном кодировании.

3. Сохранить в отчете экранные формы, демонстрирующие процесс сжатия информации.

4. Отчет по лабораторной работе должен содержать результаты анализа программы *Shannon-Huffman.exe* при кодировании; алгоритм построения кода Хаффмена и Шеннона с применением знаний по данной тематике; результаты сравнения различных методов кодирования; выводы об эффективности используемых методов сжатия.

5. Привести в отчете ответы на контрольные вопросы, в соответствии с номером варианта, указанным преподавателем.

Включить в отчет о лабораторной работе выполненные задания 1, 2, 3.

Варианты заданий

Задание 1.

Сообщение состоит из последовательности двух букв А и В, вероятности появления каждой из которых не зависят от того, какая была передана раньше, и равны 0,8 и 0,2 соответственно. Произведите кодирование по методу Шеннона: а) отдельных букв; б) блоков, состоящих из двухбуквенных сочетаний; в) блоков, состоящих из трехбуквенных сочетаний. Сравните полученные коды по их эффективности.

Задание 2.

Составьте текст, который бы соответствовал данным, приведенным в Таблице 11.3.

Используя программу *Shannon-Huffman.exe* закодируйте текст методом Хаффмена.

Таблица 11.3

Номер Варианта	вероятность	символ	число символов
1,5,7, 3,9,12,14,18,28	0,333333333	о	2
	0,166666667	г	1
	0,166666667	р	1
	0,166666667	д	1
	0,166666667	а	1
2,4,6,8,16,21 20,22,24, 30	0,25	е	2
	0,25	т	2
	0,125	о	1
	0,125	п	1
	0,125	р	1
	0,125	а	1
11,13,15, 10,17,19,23,25	0,25	р	2
	0,25	а	2
	0,25	с	2
	0,125	т	1
	0,125	е	1

Задание 3.

Для вариантов А),Б),В) (см.рис.11.6 и Таблица 4) составьте код Хаффмена. Сделайте подсчет среднего количества символов на знак; избыточности $(L - H)$; и относительной избыточности полученного кода $(L - H) / L$. Сравните полученные значения с L , H , $(L-H)$ для кода Шеннона, сделайте выводы.

А)

Результат анализа:			
Вероятность	Число символов	Символ	Код
0,4375	7	а	11
0,1875	3	б	10
0,1875	3	г	01
0,125	2	в	001
0,0625	1	д	000

H = 2,05242128382936 , L = 2,1875 , L - H = 0,135078716170635

Б)

Результат анализа:			
Вероятность	Число символов	Символ	Код
0,357142857142857	5	к	11
0,285714285714286	4	е	10
0,142857142857143	2	ж	011
0,142857142857143	2	з	010
0,0714285714285714	1	и	00

H = 2,1209520310264 , L = 2,28571428571429 , L - H = 0,164762254687883

В)

Результат анализа:			
Вероятность	Число символов	Символ	Код
0,363636363636364	4	л	11
0,272727272727273	3	о	10
0,181818181818182	2	м	01
0,0909090909090909	1	н	001
0,0909090909090909	1	п	000

H = 2,11807820934971 , L = 2,18181818181818 , L - H = 0,0637399724684737

Рис. 11. 6. Варианты заданий

Таблица 11.4

Номер варианта	Задание
1,5,7, 3,9, 12,14,18, 28	Задание 3. Вариант А).

2,4,6,8, 20,22,23, 24,25 30	<i>Задание 3. Вариант Б).</i>
11,13,15, 16 21,10,17, 19	<i>Задание 3. Вариант В).</i>

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Какие коды позволяют производить однозначное декодирование даже без использования разделительного символа? Приведите примеры таких кодов.
2,4,6,8, 20,22,24, 26,30	Назовите условия построения оптимальных кодов.
11,13,15, 10,17,19, 27	С какой целью используются эффективные коды, и какие из них вам известны?
12,14,16 21,23,25, 29	Перечислите основные методы сжатия информации без потерь.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по организации и проведению самостоятельной работы
по дисциплине «Основы информационной безопасности
информационно-коммуникационных систем» для
студентов направления подготовки

11.03.02 Инфокоммуникационные технологии и системы связи

Направленность (профиль)

«Интеллектуальная обработка данных в инфокоммуникационных
системах и сетях»

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
1. ОБЩАЯ ХАРАКТЕРИСТИКА САМОСТОЯТЕЛЬНОЙ РАБОТЫ.....	9
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ИЗУЧЕНИЮ ТЕОРЕТИЧЕСКОГО МАТЕРИАЛА	12
3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ	13
4. ВОПРОСЫ ДЛЯ СОБЕСЕДОВАНИЯ ПО САМОСТОЯТЕЛЬНОМУ ИЗУЧЕНИЮ ЛИТЕРАТУРЫ	14
5. КОМПЬЮТЕРНЫЕ ТЕСТЫ ДЛЯ САМОТЕСТИРОВАНИЯ, ПОДГОТОВКИ К ТЕСТИРОВАНИЮ	18
6. КОМПЛЕКТ РАЗНОУРОВНЕВЫХ ЗАДАНИЙ ДЛЯ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКИХ РЕДАКТОРОВ	22
7. ВОПРОСЫ К ЭКЗАМЕНУ	28

ВВЕДЕНИЕ

1. Цель и задачи освоения дисциплины (модуля)

Целью изучения дисциплины «Основы информационной безопасности информационно-коммуникационных систем» является понимание будущим выпускником роли безопасного хранения и использования данных в информационной системе и подготовленность в разрезе информационной безопасности. Цель дисциплины овладение базовыми знаниями в области информационной защиты информационных систем и сетей на основе современных программных и операционных систем и активного оборудования. Задачами дисциплины являются изучение программно-аппаратных средств защиты информации, методов анализа и планирования информационной защиты компьютерных систем, сетей и их компонентов, средств защиты сетевых служб.

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Основы информационной безопасности информационно-коммуникационных систем» относится к части «Часть, формируемая участниками образовательных отношений», Б1.В.02.06. Ее освоение происходит в 3 семестре.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесённых с планируемыми результатами освоения образовательной программы

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ПК-7. Способен к администрированию средств обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов)	ИД-1ПК-7 понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности сетевых устройств.	Понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности информационно-коммуникационных систем.
	ИД-2ПК-7 понимает общие принципы функционирования и архитектуру аппаратных, программных и программно-аппаратных средств администрируемой сети; протоколы канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем.	Понимает общие принципы функционирования и архитектуру аппаратных, программных и программно-аппаратных средств администрируемой сети; протоколы канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем средств обеспечения информационной безопасности информационно-коммуникационных систем.
	ИД-3ПК-7 применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа; настраивает параметры современных программно-аппаратных межсетевых экранов.	Применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа.

	ИД-4пк-7 подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов); работает с контрольно-измерительными аппаратными и программными средствами.	Подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов).
	ИД-5пк-7 выполняет установку дополнительных программных продуктов для обеспечения безопасности удаленного доступа и их параметризация.	Выполняет установку дополнительных программных продуктов для обеспечения безопасности информационно-коммуникационных систем.

4. Объем учебной дисциплины (модуля) и формы контроля *

Объем занятий: всего: 4 з.е. 108 астр.ч.	ОФО, в астр. часах
Контактная работа:	48
Лекции/из них практическая подготовка	24 / -
Лабораторных работ/из них практическая подготовка	24 / 24
Практических занятий/из них практическая подготовка	- / -
Самостоятельная работа	19,5
Формы контроля	
Экзамен	40,5
Зачет	
Зачет с оценкой	
Расчетно-графические работы	
Курсовые работы	
Контрольные работы	

* Дисциплина (модуль) предусматривает применение электронного обучения, дистанционных образовательных технологий (если иное не установлено образовательным стандартом)

5. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием количества часов и видов занятий

№	Раздел (тема) дисциплины и краткое содержание	Формируемые компетенции, индикаторы	очная форма			
			Контактная работа обучающихся с преподавателем /из них в форме практической подготовки, часов			Самостоятельная работа, часов
			Лекции	Практические занятия	Лабораторные работы	

1	Основы информационной безопасности. 1. Понятие информационной безопасности. 2. Основные составляющие информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
2	Основы информационной безопасности. 1. Важность и сложность проблемы информационной безопасности. 2. Сценарии реализации угроз информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
3	Безопасность компьютерных систем. 1. Безопасность компьютерных систем. 2. Традиционный подход к анализу проблем информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
4	Безопасность компьютерных систем. 1. Объектно-ориентированный подход 2. Перспективные принципы анализа вопросов информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
5	Основные определения и критерии классификации угроз. 1. Основные понятия об угрозах. 2. Наиболее распространенные угрозы доступности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
6	Основные определения и критерии классификации угроз. 1. Основные угрозы целостности. 2. Основные угрозы конфиденциальности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
7	Уровни обеспечения информационной безопасности. 1. Законодательный уровень информационной безопасности. 2. Стандарты и спецификации в области информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
8	Уровни обеспечения информационной безопасности. 1. Административный уровень информационной безопасности. 2. Процедурный уровень информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1

9	Основные программно-технические меры обеспечения информационной безопасности. 1. Основные программно-технические меры обеспечения информационной безопасности. 2. Основные понятия программно-технического уровня информационной безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
10	Основные программно-технические меры обеспечения информационной безопасности. 1. Особенности современных информационных систем, существенные при обеспечении информационной безопасности. 2. Архитектура системы безопасности.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
11	Криптографическая защита информации. 1. Основные принципы криптографической защиты информации. 2. Аппаратно-программные криптографические средства защиты информации.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
12	Криптографическая защита информации. 1. Асимметричные криптосистемы. 2. Симметричные криптосистемы.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	4,5
13	Вредоносные программы и компьютерные вирусы. 1. Вредоносные программы и компьютерные вирусы. 2. Основные понятия и классификация вредоносных программ.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
14	Вредоносные программы и компьютерные вирусы. 1. Основы борьбы с вредоносными программами. 2. Комплексные средства антивирусной защиты.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1
15	Информационная безопасность в глобальных сетях. 1. Типовые удаленные атаки в глобальных компьютерных сетях. 2. Механизмы реализации удаленных атак в глобальной сети Internet.	ИД-1_{ПК-7} ИД-2_{ПК-7} ИД-3_{ПК-7} ИД-4_{ПК-7} ИД-5_{ПК-7}	1,5 / -		1,5/1,5	1

16	Информационная безопасность в глобальных сетях. 1. Обеспечение безопасности систем входящих в состав глобальных компьютерных сетей. 2. Обеспечение безопасного взаимодействия в глобальных компьютерных сетях.	ИД-1 _{ПК-7} ИД-2 _{ПК-7} ИД-3 _{ПК-7} ИД-4 _{ПК-7} ИД-5 _{ПК-7}	1,5 / -		1,5/1,5	1
	ИТОГО за семестр		24		24	19,5
	ИТОГО					

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Код оцениваемой компетенции, индикатора (ов)	Этап формирования компетенции (№ темы) (в соответствии с рабочей программой дисциплины)	Средства и технологии оценки	Вид контроля, аттестация (текущий/промежуточный)	Тип контроля (устный, письменный или с использованием технических средств)	Наименование оценочного средства
3 семестр					
ИД-1ПК-7 ИД-2ПК-7 ИД-3ПК-7 ИД-4ПК-7 ИД-5ПК-7	Темы № 1–8	Собеседование, оценка ответов по вопросам заданных тем	текущий	устный	вопросы для собеседования
ИД-1ПК-7 ИД-2ПК-7 ИД-3ПК-7 ИД-4ПК-7 ИД-5ПК-7	Темы № 6, 8	Зачетное задание (контрольная точка), выполнение индивидуальных (групповых) заданий из лабораторных работ	текущий	письменный	вопросы для защиты лабораторных работ
ИД-1ПК-7 ИД-2ПК-7 ИД-3ПК-7 ИД-4ПК-7 ИД-5ПК-7	Темы № 1–8	Оценка знаний экзаменационных вопросов	промежуточный	устный	Вопросы к экзамену

2. Описание показателей и критериев оценивания на различных этапах их формирования, описание шкал оценивания

Уровни сформированности компетенций, индикаторов	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
Компетенция: ПК-7				
ИД-1ПК-7 понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности сетевых устройств.	Не может выполнить задачи, относящиеся к данному индикатору достижения компетенции	Выполняет на репродуктивном уровне задачи, относящиеся к данному индикатору достижения компетенции	Выполняет в стандартных ситуациях задачи, относящиеся к данному индикатору достижения компетенции, допуская незначительные ошибки	Выполняет в полном объеме задачи, относящиеся к данному индикатору достижения компетенции, в том числе в нестандартных ситуациях

ИД-2пк-7 понимает общие принципы функционирования и архитектуру аппаратных, программных и программно-аппаратных средств администрируемой сети; протоколы канального, сетевого, транспортного и прикладного уровней модели взаимодействия открытых систем.	Не может выполнить задачи, относящиеся к данному индикатору достижения компетенции	Выполняет на репродуктивном уровне задачи, относящиеся к данному индикатору достижения компетенции	Выполняет в стандартных ситуациях задачи, относящиеся к данному индикатору достижения компетенции, допуская незначительные ошибки	Выполняет в полном объеме задачи, относящиеся к данному индикатору достижения компетенции, в том числе в нестандартных ситуациях
ИД-3пк-7 применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа; настраивает параметры современных программно-аппаратных межсетевых экранов.	Не может выполнить задачи, относящиеся к данному индикатору достижения компетенции	Выполняет на репродуктивном уровне задачи, относящиеся к данному индикатору достижения компетенции	Выполняет в стандартных ситуациях задачи, относящиеся к данному индикатору достижения компетенции, допуская незначительные ошибки	Выполняет в полном объеме задачи, относящиеся к данному индикатору достижения компетенции, в том числе в нестандартных ситуациях
ИД-4пк-7 подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов); работает с контрольно-измерительными аппаратными и программными средствами.	Не может выполнить задачи, относящиеся к данному индикатору достижения компетенции	Выполняет на репродуктивном уровне задачи, относящиеся к данному индикатору достижения компетенции	Выполняет в стандартных ситуациях задачи, относящиеся к данному индикатору достижения компетенции, допуская незначительные ошибки	Выполняет в полном объеме задачи, относящиеся к данному индикатору достижения компетенции, в том числе в нестандартных ситуациях
ИД-5пк-7 выполняет установку дополнительных программных продуктов для обеспечения безопасности	Не может выполнить задачи, относящиеся к данному индикатору достижения компетенции	Выполняет на репродуктивном уровне задачи, относящиеся к данному индикатору достижения компетенции	Выполняет в стандартных ситуациях задачи, относящиеся к данному индикатору достижения компетенции,	Выполняет в полном объеме задачи, относящиеся к данному индикатору достижения компетенции, в том числе в

удаленного доступа и их параметризация.			допуская незначительные ошибки	нестандартных ситуациях
---	--	--	--------------------------------	-------------------------

Описание шкалы оценивания

В рамках рейтинговой системы успеваемость студентов по каждой дисциплине оценивается в ходе текущего контроля и промежуточной аттестации.

Текущий контроль

Рейтинговая оценка знаний студента (в случаях, предусмотренных нормативными актами СКФУ).

№ п/п	Вид деятельности студентов	Сроки выполнения	Количество баллов
3 семестр			
1	Зачетное занятие по лаб. работам № 1–4 с проверкой выполнения разноуровневых индивидуальных заданий.	7	12
2	Зачетное занятие по лаб. работам № 5–8 с проверкой выполнения разноуровневых индивидуальных заданий.	13	18
3	Зачетное занятие по лаб. работам № 9–12 с проверкой выполнения разноуровневых индивидуальных заданий. Собеседование по результатам самостоятельного изучения литературы.	17	25

Максимально возможный балл за весь текущий контроль устанавливается равным 55. Текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

<i>Уровень выполнения контрольного задания</i>	<i>Рейтинговый балл (в % от максимального балла за контрольное задание)</i>
<i>Отличный</i>	<i>100</i>
<i>Хороший</i>	<i>80</i>
<i>Удовлетворительный</i>	<i>60</i>
<i>Неудовлетворительный</i>	<i>0</i>

3. Типовые контрольные задания и иные материалы, характеризующие этапы формирования компетенций

1. Критерии оценивания компетенций

Оценка «отлично» выставляется студенту, если он выполняет в полном объеме прикладные практические задачи профессиональной деятельности в области информационной безопасности, понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности информационно-коммуникационных систем, применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа, подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов), выполняет

установку дополнительных программных продуктов для обеспечения безопасности информационно-коммуникационных систем.

Оценка **«хорошо»** выставляется студенту, если он выполняет прикладные практические задачи профессиональной деятельности в области информационной безопасности, понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности информационно-коммуникационных систем, применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа, подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов), выполняет установку дополнительных программных продуктов для обеспечения безопасности информационно-коммуникационных систем.

Оценка **«удовлетворительно»** выставляется студенту, если он с трудом выполняет прикладные практические задачи профессиональной деятельности в области информационной безопасности, понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности информационно-коммуникационных систем, применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа, подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов), выполняет установку дополнительных программных продуктов для обеспечения безопасности информационно-коммуникационных систем.

Оценка **«неудовлетворительно»** выставляется студенту, если он не выполняет прикладные практические задачи профессиональной деятельности в области информационной безопасности, понимает и использует основные принципы, протоколы и программные криптографические средства обеспечения информационной безопасности информационно-коммуникационных систем, применяет программные, аппаратные и программно-аппаратные средства защиты сетевых устройств от несанкционированного доступа, подключает и настраивает современные средства обеспечения безопасности удаленного доступа (операционных систем и специализированных протоколов), выполняет установку дополнительных программных продуктов для обеспечения безопасности информационно-коммуникационных систем.

2. Описание шкалы оценивания

Максимально возможный балл за весь текущий контроль устанавливается равным 55. Текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

Уровень выполнения контрольного задания	Рейтинговый балл (в % от максимального балла за контрольное задание)
Отличный	100
Хороший	80
Удовлетворительный	60
Неудовлетворительный	0

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Текущая аттестация студентов проводится преподавателями, ведущими лабораторные работы по дисциплине, в следующих формах:

— собеседование по результатам самостоятельного изучения и конспектирования литературы;

— выполнение разноуровневых заданий с использованием специализированного программного обеспечения.

При подготовке к контрольной точке в ходе лабораторной работе студент должен:

— оформить отчет по лабораторной работе с выполненными разноуровневыми индивидуальными заданиями;

— подготовиться к ответам на вопросы собеседования по результатам самостоятельного изучения литературы.

Защита контрольной точки осуществляется путем собеседования студента с преподавателем. При собеседовании студент представляет на проверку отчет по лабораторной работе. Собеседование со студентом, претендующим на оценку «отлично», проводится по вопросам и заданиям повышенного уровня обучения.

Критерии оценивания результатов самостоятельной работы по курсу «Обеспечение информационной безопасности в инфокоммуникациях» приведены в Фонде оценочных средств по дисциплине.

ВОПРОСЫ ДЛЯ СОБЕСЕДОВАНИЯ ПО САМОСТОЯТЕЛЬНОМУ ИЗУЧЕНИЮ ЛИТЕРАТУРЫ

Базовый уровень:

Тема 1. Основы информационной безопасности.

3. Понятие информационной безопасности.
4. Основные составляющие информационной безопасности.
5. Важность и сложность проблемы информационной безопасности.
6. Сценарии реализации угроз информационной безопасности.

Тема 2. Безопасность компьютерных систем.

1. Традиционный подход к анализу проблем информационной безопасности.
2. Объектно-ориентированный подход – перспективный принцип анализа вопросов информационной безопасности.

Тема 3. Основные определения и критерии классификации угроз.

1. Основные понятия об угрозах.
2. Наиболее распространенные угрозы доступности.
3. Основные угрозы целостности.
4. Основные угрозы конфиденциальности.

Тема 4. Уровни обеспечения информационной безопасности.

1. Законодательный уровень информационной безопасности.
2. Стандарты и спецификации в области информационной безопасности.
3. Административный уровень информационной безопасности.
4. Процедурный уровень информационной безопасности.

Тема 5. Основные программно-технические меры обеспечения информационной безопасности.

1. Основные понятия программно-технического уровня информационной безопасности.
2. Особенности современных информационных систем, существенные при обеспечении информационной безопасности.
3. Архитектура системы безопасности.

Тема 6. Криптографическая защита информации.

1. Основные принципы криптографической защиты информации.
2. Аппаратно-программные криптографические средства защиты информации.

Тема 7. Криптографическая защита информации.

1. Асимметричные криптосистемы.
2. Симметричные криптосистемы.

Тема 8. Вредоносные программы и компьютерные вирусы.

1. Основные понятия и классификация вредоносных программ.
2. Основы борьбы с вредоносными программами.
3. Комплексные средства антивирусной защиты.

Тема 9. Информационная безопасность в глобальных сетях.

1. Типовые удаленные атаки в глобальных компьютерных сетях.
2. Механизмы реализации удаленных атак в глобальной сети Internet.
3. Обеспечение безопасности систем входящих в состав глобальных компьютерных сетей.
4. Обеспечение безопасного взаимодействия в глобальных компьютерных сетях.

Повышенный уровень:

Тема 1. Основы информационной безопасности.

1. Какая информация является конфиденциальной?
2. Что относится к защищаемой информации?
3. Что понимается под политикой безопасности?

4. Что понимается под несанкционированным воздействием на защищаемую информацию?

5. Дайте понятие конфиденциальности, целостности и доступности информации.

6. Дайте определение информационной безопасности.

7. Какие цели и задачи включает в себя концепция национальной безопасности РФ?

8. Перечислите основные виды угроз информационной безопасности РФ.

9. Дайте определение комплексного обеспечения информационной безопасности.

10. Перечислите основные элементы организационной основы государственной системы обеспечения информационной безопасности РФ.

11. Перечислите основные технологические мероприятия комплексной системы обеспечения информационной безопасности РФ.

12. Перечислите основные методы и средства обеспечения защиты информации.

13. Что включают в себя организационные средства обеспечения защиты информации?.

Тема 2. Безопасность компьютерных систем.

1. Что включают в себя правовые средства обеспечения защиты информации?

2. Что включают в себя инженерно-технические средства защиты информации?

3. Что включают в себя программно-аппаратные средства защиты информации?

4. Перечислите требования, предъявляемые к системам защиты информации.

5. Какие существуют модели защиты информации?

6. Перечислите основные мероприятия организации работ по обеспечению защиты информации.

7. Приведите структуру и содержание цикла работ по защите информации..

Тема 3. Основные определения и критерии классификации угроз.

1. Дайте определение организационного обеспечения защиты информации.

2. Приведите классификацию всех возможных угроз информационной безопасности.

3. Что относится к угрозам нарушения целостности информации?

4. Что относится к угрозам нарушения конфиденциальности информации?

5. Что относится к угрозам нарушения доступности информации?

6. Дайте определения разглашения, утечки и несанкционированного доступа к информации.

7. Перечислите основные способы несанкционированного доступа к информации.

8. Что относится к числу основных задач службы безопасности?

9. Перечислите основные принципы построения системы безопасности предприятия.

10. Перечислите основные функции службы защиты информации.

11. Приведите типовую структуру службы безопасности.

12. Перечислите и охарактеризуйте основные этапы процедуры отбора персонала на работу.

13. Приведите особенности увольнения сотрудников, владеющих конфиденциальной информацией.

14. Что понимается под режимом секретности?

15. Что включает в себя режим секретности?

Тема 4. Уровни обеспечения информационной безопасности.

1. Перечислите основные уровни правового обеспечения защиты информации.

2. Приведите деление информации с точки зрения права.

3. Какие сведения относятся к государственной тайне?

4. Какие сведения не относятся к государственной тайне?

5. Перечислите и охарактеризуйте формы допуска к секретным сведениям.

6. На кого возлагается организация доступа гражданина к сведениям, составляющим государственную тайну?

7. Какие сведения относятся к коммерческой тайне?

8. Какие сведения относятся к банковской тайне?

9. Какие сведения относятся к профессиональной тайне?

10. Какие сведения относятся к служебной тайне?
11. Что относится к числу основных объектов интеллектуальной собственности?
12. Перечислите законодательные акты, осуществляющие правовое регулирование объектов авторского права.
13. Что относится к объектам авторского права?
14. Что относится к основным конституционным гарантиям в области авторского права?

Тема 5. Основные программно-технические меры обеспечения информационной безопасности.

1. Перечислите основные сферы интересов разведки государства.
2. Перечислите основные задачи органов коммерческой разведки, их состав и возможности.
3. Приведите структуру канала утечки информации и дайте его общую характеристику.
4. Представьте классификацию каналов утечки информации.
5. Приведите структурную схему средства наблюдения в оптическом диапазоне и дайте его общую характеристику.
6. Охарактеризуйте способы и средства наблюдения в радиодиапазоне.
7. Приведите структуру комплекса средств перехвата и дайте его общую характеристику.
8. Охарактеризуйте способы и средства подслушивания информации.
9. Приведите структуру акустического приемника и дайте его общую характеристику.
10. Приведите структуру акустической закладки и дайте ее общую характеристику.
11. Дайте определение инженерно-технической защиты информации и перечислите ее задачи.
12. Приведите классификацию методов инженерно-технической защиты информации.
13. Дайте определение дезинформирования как метода защиты информации.
14. Перечислите виды и способы охраны объекта.
15. Перечислите виды и состав систем видеонаблюдения.
16. Дайте определение системы охранно-пожарной сигнализации и перечислите ее составные элементы.

17. Перечислите методы и средства инженерной защиты объектов информатизации.

Тема 6. Криптографическая защита информации.

3. Основные принципы криптографической защиты информации.

4. Аппаратно-программные криптографические средства защиты информации.

Тема 7. Криптографическая защита информации.

1. Перечислите основные понятия и определения, используемые в криптографии.
2. Охарактеризуйте основные этапы истории развития криптографии.
3. Приведите классификацию криптографических алгоритмов шифрования.
4. Дайте характеристику шифров замены.
5. Дайте характеристику шифра гаммирования.
6. Дайте характеристику шифра блочной замены.
7. Дайте характеристику шифра многоалфавитной замены.
8. Поясните принцип аналогового скремблирования.
9. Приведите структуру блок-схемы комбинированного скремблера и поясните назначение ее элементов.
10. Поясните принцип цифрового шифрования.
11. Дайте понятие распределения ключей и приведите их основные способы.
12. Дайте общую характеристику шифра RSA.
13. Дайте общую характеристику криптографического стандарта шифрования данных

DES.

14. Дайте общую характеристику криптографического стандарта шифрования ГОСТ 28147–89.

15. Приведите понятие функции хеширования.

16. Поясните основные свойства и процедуры электронной цифровой подписи.

17. Приведите схему формирования электронной цифровой подписи и поясните ее основные этапы.

18. Поясните процедуру проверки электронной цифровой подписи.

19. Дайте понятие идентификации и аутентификации.

20. Приведите классификацию протоколов аутентификации.

21. Приведите классификацию протоколов строгой аутентификации.

22. Дайте общую характеристику биометрической аутентификации.

Тема 8. Вредоносные программы и компьютерные вирусы.

1. Приведите общие сведения о компьютерных вирусах.

2. Охарактеризуйте основные этапы жизненного цикла компьютерных вирусов.

3. Приведите схему заражения файлов компьютерными вирусами.

4. Приведите схему заражения загрузчиков компьютерными вирусами.

5. Дайте общую характеристику способам маскировки, используемым компьютерными вирусами.

6. Приведите классификацию компьютерных вирусов.

7. Охарактеризуйте основные методы и технологии борьбы с компьютерными вирусами.

8. Поясните принцип защиты от деструктивных действий и размножения компьютерных вирусов.

9. Поясните стратегию заблаговременной подготовки к эффективной ликвидации последствий вирусной эпидемии в компьютерной системе.

10. Поясните технологию гарантированного восстановления вычислительной системы после заражения компьютерными вирусами.

Тема 9. Информационная безопасность в глобальных сетях.

1. Типовые удаленные атаки в глобальных компьютерных сетях.

2. Механизмы реализации удаленных атак в глобальной сети Internet.

3. Обеспечение безопасности систем входящих в состав глобальных компьютерных сетей.

4. Обеспечение безопасного взаимодействия в глобальных компьютерных сетях.

5. Приведите методы и средства защиты информации в типовых операционных системах.

6. Приведите методы и средства защиты информации в системах управления базами данных

Критерии оценивания компетенций:

Оценка «отлично» выставляется студенту, если студент дал полный, развернутый ответ на поставленный вопрос, при этом показана совокупность осознанных знаний по теме дисциплины, доказательно раскрыты основные положения вопроса; в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий. Ответ изложен литературным языком с использованием технической терминологии. Могут быть допущены недочеты в определении понятий, исправленные студентом самостоятельно в процессе ответа. Ситуационная задача по теме решена правильно.

Оценка «хорошо» выставляется студенту, если студент дал полный, развернутый ответ на поставленный вопрос, при этом показано умение выделить существенные и несущественные признаки, причинно-следственные связи. Ответ четко структурирован, логичен, изложен литературным языком с использованием технической терминологии. Могут быть допущены 2-3 неточности или незначительные ошибки, исправленные студентом с

помощью преподавателя. Ситуационная задача по теме решена правильно, но при ее решении допущено не более одной ошибки.

Оценка «удовлетворительно» выставляется студенту, если студент дал недостаточно полный и недостаточно развернутый ответ. Логика и последовательность изложения имеют нарушения. Допущены ошибки в раскрытии понятий, употреблении терминов. Студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи. В ответе отсутствуют выводы. Умение раскрыть значение обобщенных знаний не показано. Речевое оформление требует поправок, коррекции. Ситуационная задача по теме решена правильно, но при ее решении допущено не более двух ошибок.

Оценка «неудовлетворительно» выставляется студенту, если ответ представляет собой разрозненные знания с существенными ошибками по вопросу. Присутствуют фрагментарность, нелогичность изложения. Студент не осознает связь обсуждаемого вопроса с другими вопросами темы. Отсутствуют выводы, конкретизация и доказательность изложения. Речь неграмотная, терминология не используется. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента, или ответ на вопрос полностью отсутствует или студент отказался от ответа. Ситуационная задача по теме решена неправильно.

Описание шкалы оценивания

Оценка за ответы на вопросы в ходе собеседования определяется в соответствии с таблицей.

Оценка	Оценка за теоретический вопрос базового уровня	Оценка за теоретический вопрос повышенного уровня	Примечание
отлично	Отлично (хорошо)	Отлично	Оценка хорошо допускается только по вопросу базового уровня
хорошо	Хорошо (удовлетворительно)	Хорошо	Оценка удовлетворительно допускается только по вопросу базового уровня
удовлетворительно	Удовлетворительно	Удовлетворительно	
неудовлетворительно	По одному или двум вопросам получена оценка неудовлетворительно		

Согласно рейтинговой системы текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

Уровень выполнения контрольного задания	Рейтинговый балл (в % от максимального балла за контрольное задание)
Отличный	100
Хороший	80
Удовлетворительный	60
Неудовлетворительный	0

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедура проведения данного оценочного мероприятия включает в себя собеседование по двум теоретическим вопросам из числа перечисленных. Один вопрос

относится к базовому уровню, а второй – к повышенному. В ходе собеседования студент имеет право пользоваться собственным конспектом.

Предлагаемые студенту вопросы позволяют проверить профессиональные компетенции ПК-7. Принципиальные отличия заданий базового уровня от повышенного заключаются в сложности вопросов по уровню обученности «понимание».

Для подготовки к данному оценочному мероприятию необходимо 34,5 часов в 6-м семестре. Студент должен подготовить письменные ответы на перечисленные теоретические вопросы в виде конспекта учебного материала и на их основании подготовиться к указанным в методических материалах вопросам для обсуждения.

При проверке задания, оцениваются активное участие в работе; уверенное владение материалом; умение четко и логично излагать свои мысли; умение творчески подходить к решению основных вопросов темы; самостоятельность мышления, полное соответствие конспекта установленным требованиям; амостоятельная устная речь, полностью раскрывающая суть суть проблематики собеседования.

Оценочный лист

№ п/п	Фамилия, имя студента.	Вид работы						Итог
		Работа с дополните льными источника ми информаци и	Ответы на промежу точные вопросы	Логика и научнос ть изложе ния	Формулир ование выводов и доказатель ств	Выступл ение с результат ами работы	Участие в дискуссии , умение вести диалог	
1								
2								
...								

КОМПЛЕКТ РАЗНОУРОВНЕВЫХ ЗАДАНИЙ ДЛЯ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКИХ РЕДАКТОРОВ

Лабораторная работа № 1. Использование классических криптоалгоритмов подстановки и перестановки для защиты текстовой информации

Базовый уровень:

1. Какие вы знаете методы криптографической защиты файлов?
2. В чем преимущества и недостатки одноалфавитных методов?
3. Если вам необходимо зашифровать текст, содержащий важную информацию, какой метод, из рассмотренных, вы выберете? Обоснуйте свой выбор?
4. Целесообразно ли повторно применять для уже зашифрованного текста: а) метод многоалфавитного шифрования? б) метод Цезаря?

Повышенный уровень:

1. Где могут использоваться одноалфавитные методы шифрования?

Лабораторная работа 2. Исследование различных методов защиты текстовой информации и их стойкости на основе подбора ключей

Базовый уровень:

1. Чем отличается “псевдооткрытый” текст (текст, полученный при расшифровке по ложному ключу) от настоящего открытого текста ?
2. Как зависит время вскрытия шифра описанным выше способом подбора ключей от длины “вероятного” слова?
3. Зависит ли время вскрытия шифра гаммирования (или таблицы Виженера) от мощности алфавита гаммы ?
4. В чем недостатки метода дешифрования с использованием протяжки вероятного слова?

Повышенный уровень:

1. Какова стойкость изученных методов?

Лабораторная работа 3. Исследование шифрования методом Виженера

Базовый уровень:

1. В чем преимущество шифров многоалфавитной замены?
2. Что из себя представляет множество алфавитов в шифре Виженера?
3. Какой формулой описывается алгоритм Виженера?
4. Можно ли шифр Цезаря назвать частным случаем шифра Виженера?
5. Что из себя представляет шифр Вернама?

Повышенный уровень:

1. Какой формулой можно описать процесс шифрования методом Виженера?

Лабораторная работа 4. Исследование стандарта симметричного шифрования AES RIJNDAEL

Базовый уровень:

1. Сравните основные характеристики алгоритмов Rijndael и ГОСТ 28147-89.
2. Сравните основные характеристики алгоритмов Rijndael и DES.
3. Опишите структуру сети Фейстеля.
4. Приведите обобщенные схемы шифрования данных с помощью алгоритма Rijndael и ГОСТ 28147-89. Дайте их сравнительный анализ.
5. Сравните один раунд шифрования данных с помощью алгоритма Rijndael и ГОСТ 28147-89.

6. Сравните эквивалентность прямого и обратного преобразований в алгоритмах Rijndael и ГОСТ 28147-89.
7. Сравните выработку ключевой информации в алгоритмах Rijndael и ГОСТ 28147-89.
8. Сравните алгоритмы Rijndael и ГОСТ 28147-89 по показателям производительности и удобству реализации.

Повышенный уровень:

1. Сравните алгоритмы Rijndael и ГОСТ 28147-89 по показателям диффузии.
2. Сравните алгоритмы Rijndael и ГОСТ 28147-89 по показателям стойкости

Лабораторная работа 5. Генерация простых чисел, используемых в асимметричных системах шифрования

Базовый уровень:

1. Почему в качестве первого основания в тестах типа теста Ферма для проверки на простоту очень больших чисел целесообразно использовать число 2?
2. Какова вероятность $P(x)$ того, что наугад взятое нечетное очень большое число, не превосходящее x , окажется простым.
3. Вычислить: $1812 \pmod{13}$.
4. Вычислить $127 \pmod{7}$.
5. Вычислить $1345 \pmod{11}$?

Повышенный уровень:

1. Сформулируйте суть теста на простоту с использованием пробных делений?

Лабораторная работа 6. Исследование электронной подписи

Базовый уровень:

1. В чем состоит назначение хэш-функций и какие требования предъявляются к хэш-функциям, используемым для постановки ЭЦП?
2. Перечислите стандарты хэш-функций, действующие в Российской Федерации?
3. Опишите процедуры постановки и проверки ЭЦП?
4. Какая информация содержится в ЭЦП?
5. Перечислите стандарты ЭЦП, действующие в Российской Федерации
6. На каких принципах основана криптостойкость современных алгоритмов ЭЦП?

Повышенный уровень:

1. Приведите пример реализации алгоритма ЭЦП (RSA, El-Gamal, DSA)

Лабораторная работа 7. Шифрование методом скользящей перестановки

Базовый уровень:

1. Почему шифрование методом гаммирования является наиболее подходящим для высокоскоростных линий телекоммуникационной связи.
2. Какие общие требования, предъявляются к гамме шифра.
3. Приведите пример, поясняющий работу шифрующего автомата скользящей перестановки при $n=5$, $n_1=2$, $n_2=3$.
4. Кратко опишите работу схемы реализации шифра скользящей перестановки.

Повышенный уровень:

1. Где может быть использован шифр скользящей перестановки

Лабораторная работа 8. Исследование возможностей программных продуктов ЗИ. Программа PGP

Базовый уровень:

1. Как выбрать длину криптографического ключа в системе PGP?
2. В чем достоинства и недостатки криптографических методов защиты информации?

3. Какие компьютерные системы называются безопасными?
4. В чем заключаются основные требования к защищенности компьютерных систем?
5. Для выполнения каких требований к защищенности компьютерных систем могут применяться криптографические методы защиты?
6. Насколько, на Ваш взгляд, надежны методы криптографической защиты информации, используемые в программе PGP?

Повышенный уровень:

1. Какими основными функциями защиты информации обладает программа PGP.
2. Какой принцип лежит в основе функции надежного уничтожения остаточной конфиденциальной информации программы PGP

Лабораторная работа 9. Корректирующие коды. Коды Хемминга

Базовый уровень:

1. Что представляет собой и как используется код Хэмминга.
2. Как составляется код Хэмминга?
3. Какие ошибки можно исправить с помощью кода Хэмминга.
4. От каких свойств кода зависит его корректирующая способность.
5. Объясните работу протоколов состояния связи.

Повышенный уровень:

1. Какие ошибки могут возникнуть при составлении кода Хэмминга?

Лабораторная работа 10. Корректирующие коды. Циклические коды

Базовый уровень:

1. Укажите различия понятий: кодирование информации, шифрование информации.
2. Определите понятия: равномерные коды, неравномерные коды, префиксное кодирование, разделимое кодирование.
3. От каких характеристик кода зависит его корректирующая способность.
4. Определите понятия: систематические коды, несистематические коды.

Повышенный уровень:

1. Сравните коды Хэмминга и циклические коды по эффективности контроля целостности информации.

Лабораторная работа 11. Методы сжатия по Шеннону и Хаффмену

Базовый уровень:

1. Какие коды позволяют производить однозначное декодирование даже без использования разделительного символа? Приведите примеры таких кодов.
2. Назовите условия построения оптимальных кодов.
3. С какой целью используются эффективные коды, и какие из них вам известны.

Повышенный уровень:

1. Перечислите основные методы сжатия информации без потерь

Критерии оценивания:

Критерии оценивания компетенций:

Оценка «отлично» выставляется, если студент продемонстрировал высокое умение применять полученные знания на практике через выполнение конкретного задания повышенного уровня, свободно без затруднений справился с поставленной задачей, показав владение разносторонними приемами и навыками ее выполнения, не допустил ошибок и неточностей.

Оценка «хорошо» выставляется, если студент продемонстрировал умение применять полученные знания на практике через решение конкретного задания повышенного уровня, справился с поставленной задачей, показав владение необходимыми приемами и навыками ее выполнения, при этом допустил не более одной ошибки, исправленной после указания преподавателя.

Оценка «удовлетворительно» выставляется, если студент продемонстрировал посредственное умение применять полученные знания на практике через решение конкретного задания повышенного уровня, с трудом справился с поставленной задачей, при этом допустил не более двух ошибок, исправленных после указания преподавателя, либо успешно выполнил задание базового уровня.

Оценка «неудовлетворительно» выставляется, если студент не продемонстрировал умение применять полученные знания на практике через решение конкретного задания базового уровня, не справился с поставленной задачей или допустил при ее выполнении серьезные ошибки.

Описание шкалы оценивания

Текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

Уровень выполнения контрольного задания	Рейтинговый балл (в % от максимального балла за контрольное задание)
Отличный	100
Хороший	80
Удовлетворительный	60
Неудовлетворительный	0

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедура проведения данного оценочного мероприятия включает в себя:

— представление оформленного отчета по лабораторной работе с выполненными разноуровневыми заданиями;

— защиту путем выполнения указанного преподавателем разноуровневого задания, причем уровень выполняемого задания определяется студентом самостоятельно.

Предлагаемые студенту задания позволяют проверить профессиональные компетенции ПК-7.

Для подготовки к данному оценочному мероприятию студенту отводится в 6 семестре 34.5 часов в рамках его самостоятельной работы.

При подготовке к ответу студенту предоставляется право пользования отчетом по лабораторной работе.

Оценочный лист

№ п/п	Фамилия, имя студента.	Вид работы						Итог
		Владение методикой выполнения задания	Правильность схематического моделирования	рабoтoспoсoбнoсть устрoйтвa	Точность измерений и правильность расчетов параметров и характеристик	Правильность оформления схем, графиков характеристик	Формулирование выводов, их соответствие теории	

1								
2								
...								

ВОПРОСЫ К ЭКЗАМЕНУ

Вопросы для проверки уровня обученности

Базовый уровень:

Тема 1. Основы информационной безопасности.

7. Понятие информационной безопасности.

8. Основные составляющие информационной безопасности.

9. Важность и сложность проблемы информационной безопасности.

10. Сценарии реализации угроз информационной безопасности.

Тема 2. Безопасность компьютерных систем.

3. Традиционный подход к анализу проблем информационной безопасности.

4. Объектно-ориентированный подход – перспективный принцип анализа вопросов информационной безопасности.

Тема 3. Основные определения и критерии классификации угроз.

5. Основные понятия об угрозах.

6. Наиболее распространенные угрозы доступности.

7. Основные угрозы целостности.

8. Основные угрозы конфиденциальности.

Тема 4. Уровни обеспечения информационной безопасности.

5. Законодательный уровень информационной безопасности.

6. Стандарты и спецификации в области информационной безопасности.

7. Административный уровень информационной безопасности.

8. Процедурный уровень информационной безопасности.

Тема 5. Основные программно-технические меры обеспечения информационной безопасности.

4. Основные понятия программно-технического уровня информационной безопасности.

5. Особенности современных информационных систем, существенные при обеспечении информационной безопасности.

6. Архитектура системы безопасности.

Тема 6. Криптографическая защита информации.

5. Основные принципы криптографической защиты информации.

6. Аппаратно-программные криптографические средства защиты информации.

Тема 7. Криптографическая защита информации.

3. Асимметричные криптосистемы.

4. Симметричные криптосистемы.

Тема 8. Вредоносные программы и компьютерные вирусы.

4. Основные понятия и классификация вредоносных программ.

5. Основы борьбы с вредоносными программами.

6. Комплексные средства антивирусной защиты.

Тема 9. Информационная безопасность в глобальных сетях.

5. Типовые удаленные атаки в глобальных компьютерных сетях.

6. Механизмы реализации удаленных атак в глобальной сети Internet.

7. Обеспечение безопасности систем входящих в состав глобальных компьютерных сетей.

8. Обеспечение безопасного взаимодействия в глобальных компьютерных сетях.

Повышенный уровень:

Тема 1. Основы информационной безопасности.

14. Какая информация является конфиденциальной?

15. Что относится к защищаемой информации?

16. Что понимается под политикой безопасности?
17. Что понимается под несанкционированным воздействием на защищаемую информацию?
18. Дайте понятие конфиденциальности, целостности и доступности информации.
19. Дайте определение информационной безопасности.
20. Какие цели и задачи включает в себя концепция национальной безопасности РФ?
21. Перечислите основные виды угроз информационной безопасности РФ.
22. Дайте определение комплексного обеспечения информационной безопасности.
23. Перечислите основные элементы организационной основы государственной системы обеспечения информационной безопасности РФ.
24. Перечислите основные технологические мероприятия комплексной системы обеспечения информационной безопасности РФ.
25. Перечислите основные методы и средства обеспечения защиты информации.
26. Что включают в себя организационные средства обеспечения защиты информации?.

Тема 2. Безопасность компьютерных систем.

8. Что включают в себя правовые средства обеспечения защиты информации?
9. Что включают в себя инженерно-технические средства защиты информации?
10. Что включают в себя программно-аппаратные средства защиты информации?
11. Перечислите требования, предъявляемые к системам защиты информации.
12. Какие существуют модели защиты информации?
13. Перечислите основные мероприятия организации работ по обеспечению защиты информации.
14. Приведите структуру и содержание цикла работ по защите информации..

Тема 3. Основные определения и критерии классификации угроз.

16. Дайте определение организационного обеспечения защиты информации.
17. Приведите классификацию всех возможных угроз информационной безопасности.
18. Что относится к угрозам нарушения целостности информации?
19. Что относится к угрозам нарушения конфиденциальности информации?
20. Что относится к угрозам нарушения доступности информации?
21. Дайте определения разглашения, утечки и несанкционированного доступа к информации.
22. Перечислите основные способы несанкционированного доступа к информации.
23. Что относится к числу основных задач службы безопасности?
24. Перечислите основные принципы построения системы безопасности предприятия.
25. Перечислите основные функции службы защиты информации.
26. Приведите типовую структуру службы безопасности.
27. Перечислите и охарактеризуйте основные этапы процедуры отбора персонала на работу.
28. Приведите особенности увольнения сотрудников, владеющих конфиденциальной информацией.

29. Что понимается под режимом секретности?

30. Что включает в себя режим секретности?

Тема 4. Уровни обеспечения информационной безопасности.

15. Перечислите основные уровни правового обеспечения защиты информации.
16. Приведите деление информации с точки зрения права.
17. Какие сведения относятся к государственной тайне?
18. Какие сведения не относятся к государственной тайне?
19. Перечислите и охарактеризуйте формы допуска к секретным сведениям.

20. На кого возлагается организация доступа гражданина к сведениям, составляющим государственную тайну?

21. Какие сведения относятся к коммерческой тайне?

22. Какие сведения относятся к банковской тайне?

23. Какие сведения относятся к профессиональной тайне?

24. Какие сведения относятся к служебной тайне?

25. Что относится к числу основных объектов интеллектуальной собственности?

26. Перечислите законодательные акты, осуществляющие правовое регулирование объектов авторского права.

27. Что относится к объектам авторского права?

28. Что относится к основным конституционным гарантиям в области авторского права?

Тема 5. Основные программно-технические меры обеспечения информационной безопасности.

18. Перечислите основные сферы интересов разведки государства.

19. Перечислите основные задачи органов коммерческой разведки, их состав и возможности.

20. Приведите структуру канала утечки информации и дайте его общую характеристику.

21. Представьте классификацию каналов утечки информации.

22. Приведите структурную схему средства наблюдения в оптическом диапазоне и дайте его общую характеристику.

23. Охарактеризуйте способы и средства наблюдения в радиодиапазоне.

24. Приведите структуру комплекса средств перехвата и дайте его общую характеристику.

25. Охарактеризуйте способы и средства подслушивания информации.

26. Приведите структуру акустического приемника и дайте его общую характеристику.

27. Приведите структуру акустической закладки и дайте ее общую характеристику.

28. Дайте определение инженерно-технической защиты информации и перечислите ее задачи.

29. Приведите классификацию методов инженерно-технической защиты информации.

30. Дайте определение дезинформирования как метода защиты информации.

31. Перечислите виды и способы охраны объекта.

32. Перечислите виды и состав систем видеонаблюдения.

33. Дайте определение системы охранно-пожарной сигнализации и перечислите ее составные элементы.

34. Перечислите методы и средства инженерной защиты объектов информатизации.

Тема 6. Криптографическая защита информации.

7. Основные принципы криптографической защиты информации.

8. Аппаратно-программные криптографические средства защиты информации.

Тема 7. Криптографическая защита информации.

23. Перечислите основные понятия и определения, используемые в криптографии.

24. Охарактеризуйте основные этапы истории развития криптографии.

25. Приведите классификацию криптографических алгоритмов шифрования.

26. Дайте характеристику шифров замены.

27. Дайте характеристику шифра гаммирования.

28. Дайте характеристику шифра блочной замены.

29. Дайте характеристику шифра многоалфавитной замены.

30. Поясните принцип аналогового скремблирования.

31. Приведите структуру блок-схемы комбинированного скремблера и поясните назначение ее элементов.

32. Поясните принцип цифрового шифрования.

33. Дайте понятие распределения ключей и приведите их основные способы.
34. Дайте общую характеристику шифра RSA.
35. Дайте общую характеристику криптографического стандарта шифрования данных DES.
36. Дайте общую характеристику криптографического стандарта шифрования ГОСТ 28147–89.
37. Приведите понятие функции хеширования.
38. Поясните основные свойства и процедуры электронной цифровой подписи.
39. Приведите схему формирования электронной цифровой подписи и поясните ее основные этапы.
40. Поясните процедуру проверки электронной цифровой подписи.
41. Дайте понятие идентификации и аутентификации.
42. Приведите классификацию протоколов аутентификации.
43. Приведите классификацию протоколов строгой аутентификации.
44. Дайте общую характеристику биометрической аутентификации.
- Тема 8. Вредоносные программы и компьютерные вирусы.
11. Приведите общие сведения о компьютерных вирусах.
12. Охарактеризуйте основные этапы жизненного цикла компьютерных вирусов.
13. Приведите схему заражения файлов компьютерными вирусами.
14. Приведите схему заражения загрузчиков компьютерными вирусами.
15. Дайте общую характеристику способам маскировки, используемым компьютерными вирусами.
16. Приведите классификацию компьютерных вирусов.
17. Охарактеризуйте основные методы и технологии борьбы с компьютерными вирусами.
18. Поясните принцип защиты от деструктивных действий и размножения компьютерных вирусов.
19. Поясните стратегию заблаговременной подготовки к эффективной ликвидации последствий вирусной эпидемии в компьютерной системе.
20. Поясните технологию гарантированного восстановления вычислительной системы после заражения компьютерными вирусами.
- Тема 9. Информационная безопасность в глобальных сетях.
7. Типовые удаленные атаки в глобальных компьютерных сетях.
8. Механизмы реализации удаленных атак в глобальной сети Internet.
9. Обеспечение безопасности систем входящих в состав глобальных компьютерных сетей.
10. Обеспечение безопасного взаимодействия в глобальных компьютерных сетях.
11. Приведите методы и средства защиты информации в типовых операционных системах.
12. Приведите методы и средства защиты информации в системах управления базами данных

Критерии оценивания компетенций

(в соответствии с результатами освоения дисциплины)

Критерии оценки для проверки уровня теоретической обученности:

Оценка «отлично» выставляется студенту, если он показал глубокое, прочное и аргументированное освоение программного учебного материала, при этом поставленный вопрос раскрыт последовательно, четко и логически стройно, в полном исчерпывающем объеме, основные категории, понятия и термины учебного курса формулировались правильно, не допущено при ответе ошибок

Оценка «хорошо» выставляется студенту, если он показал твердое знание программного учебного материала, при этом поставленный вопрос раскрыт грамотно и по существу, в достаточно полном объеме, основные категории, понятия и термины учебного курса формулировались правильно, допущены при ответе отдельные неточности или одна ошибка

Оценка «удовлетворительно» выставляется студенту, если он показал знание только основной части учебного материала без его частных деталей, при этом поставленный вопрос раскрыт с нарушением логической последовательности, не в полном объеме; были допущены неточные формулировки основных категорий, понятий и терминов учебного курса, а также ошибки (не более двух) или ряд незначительных неточностей, не исказивших существенно суть ответа.

Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки (более двух), существенно исказившие его суть. Оценка неудовлетворительно выставляется также, если отсутствует ответ на вопрос, либо студент отказался его сдавать.

Критерии оценки для проверки практической обученности:

оценка «отлично» выставляется, если он продемонстрировал высокое умение применять полученные знания на практике через решение конкретной задачи, свободно без затруднений справился с поставленной задачей, показав владение разносторонними приемами и навыками ее выполнения, не допустил ошибок и неточностей;

оценка «хорошо» выставляется, если он продемонстрировал умение применять полученные знания на практике через решение конкретной задачи, справился с поставленной задачей, показав владение необходимыми приемами и навыками ее выполнения, при этом допустил не более одной ошибки;

оценка «удовлетворительно» выставляется, если он продемонстрировал посредственное умение применять полученные знания на практике через решение конкретной задачи, с трудом справился с поставленной задачей, при этом допустил не более двух ошибок;

оценка «неудовлетворительно» выставляется, если он не продемонстрировал умение применять полученные знания на практике через решение конкретной задачи, не справился с поставленной задачей или допустил при ее решении три и более серьезные ошибки.

Описание шкалы оценивания

Промежуточная аттестация в форме экзамена предусматривает проведение обязательной экзаменационной процедуры и оценивается 40 баллами из 100. В случае если рейтинговый балл студента по дисциплине по итогам семестра равен 60, то программой автоматически добавляется 32 премиальных балла и выставляется оценка «отлично». Положительный ответ студента на экзамене оценивается рейтинговыми баллами в диапазоне от 20 до 40 ($20 \leq S_{\text{экз}} \leq 40$), оценка меньше 20 баллов считается неудовлетворительной.

Шкала соответствия рейтингового балла экзамена 5-балльной системе

Рейтинговый балл по дисциплине	Оценка по 5-балльной системе
35 – 40	Отлично
28 – 34	Хорошо
20 – 27	Удовлетворительно

Положительный ответ студента на экзамене оценивается рейтинговыми баллами в диапазоне от 5 до 10 за контрольный вопрос базового уровня и практическое задание, от 10 до 20 за контрольный вопрос повышенного уровня. Шкала соответствия оценок рейтинговым баллам представлена в таблице.

Оценка	Рейтинговые баллы		
	Контрольный вопрос №1 (базовый уровень)	Контрольный вопрос №2 (повышенный уровень)	Практическое задание

Отлично	10	20	10
Хорошо	7	15	7
Удовлетворительно	5	10	5
Неудовлетворительно	0	0	0

Если студент на экзамене допустил случай нарушения учебной дисциплины (списывание, использование средств мобильной связи, персональных или планшетных компьютеров, аудио- или видеоплееров, других технических устройств), то данный студент удаляется с экзамена с оценкой неудовлетворительно.

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Предлагаемые студенту задания позволяют проверить профессиональные компетенции ПК-7. Для подготовки к экзамену студенту отводится 40.50 часов в рамках его самостоятельной работы.

Процедура проведения экзамена осуществляется в соответствии с Положением о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по образовательным программам высшего образования в СКФУ.

Экзамен по дисциплине предусмотрен в устной форме по билетам. В экзаменационный билет включаются два теоретических вопроса и одно практическое задание

Для подготовки по билету отводится 30 минут - на теоретические вопросы, 10 минут на выполнение практического задания.

При подготовке к ответу студенту предоставляется право пользования рабочей программы дисциплины, собственным конспектом лекций не более 10 минут.

Практическое задание выполняется на ПЭВМ. При проверке практического задания, оцениваются: правильность моделирования ситуаций и методики исследования.