Титулка

Зміст

[Список сокращений и условных обозначений 5](#_Toc511568885)

[Словарь терминов 6](#_Toc511568886)

[Введение 9](#_Toc511568887)

[1 Методы защиты информации 10](#_Toc511568888)

[Теоретические ведомости 10](#_Toc511568889)

[1 Симметричные криптосистемы 11](#_Toc511568890)

[2 Шифры простой замены 13](#_Toc511568891)

[3 Шифры сложной замены 14](#_Toc511568892)

[4 Гаммирование 15](#_Toc511568893)

[Задания 16](#_Toc511568894)

[Вопросы для самоконтроля 16](#_Toc511568895)

[2 Безопасности шифров. Частотная атака 18](#_Toc511568896)

[Теоретические ведомости 18](#_Toc511568897)

[Задания 21](#_Toc511568898)

[Ход работы 21](#_Toc511568899)

[Вопросы для самоконтроля 24](#_Toc511568900)

[3 Потоковые шифры. Гаммирование 25](#_Toc511568901)

[Теоретические ведомости 25](#_Toc511568902)

[1 Виды криптосистем 25](#_Toc511568903)

[2 Гаммирование 26](#_Toc511568904)

[Задания 32](#_Toc511568905)

[Ход работы 35](#_Toc511568906)

[Вопросы для самоконтроля 35](#_Toc511568907)

[4 Симметричные криптосистемы. Блочные шифры 36](#_Toc511568908)

[Теоретические ведомости 36](#_Toc511568909)

[1 Сферы применения криптографии. 37](#_Toc511568910)

[Задания 37](#_Toc511568911)

[Ход работы 38](#_Toc511568912)

[Вопросы для самоконтроля 39](#_Toc511568913)

[5 Линейный криптоанализ 39](#_Toc511568914)

[Теоретические ведомости 39](#_Toc511568915)

[1 Оценка надёжности шифров 39](#_Toc511568916)

[Задания 39](#_Toc511568917)

[Ход работы 39](#_Toc511568918)

[Вопросы для самоконтроля 39](#_Toc511568919)

[6 Асимметричные шифры. Часть 1 39](#_Toc511568920)

[Теоретические ведомости 39](#_Toc511568921)

[Задания 39](#_Toc511568922)

[Ход работы 39](#_Toc511568923)

[Вопросы для самоконтроля 39](#_Toc511568924)

[7 Асимметричные шифры. Часть 2 39](#_Toc511568925)

[Теоретические ведомости 39](#_Toc511568926)

[1 Криптографические протоколы 39](#_Toc511568927)

[2 39](#_Toc511568928)

[Задания 39](#_Toc511568929)

[Ход работы 39](#_Toc511568930)

[Вопросы для самоконтроля 39](#_Toc511568931)

[8 Электронно-цифровая подпись 39](#_Toc511568932)

[Теоретические ведомости 39](#_Toc511568933)

[Задания 39](#_Toc511568934)

[Ход работы 39](#_Toc511568935)

[Вопросы для самоконтроля 39](#_Toc511568936)

[Перелік використаних джерел 39](#_Toc511568937)

[Додаток Б 39](#_Toc511568938)

1. Список сокращений и условных обозначений
2. Словарь терминов

**Открытый (исходный) текст** — данные (не обязательно текстовые), пере­даваемые без использования криптографии.

**Шифротекст, шифрованный (закрытый) текст** — данные, полученные после применения криптосистемы.

**Шифр, криптосистема** — совокупность заранее оговоренных способов преобразования исходного секретного сообщения с целью его защиты.

**Символ** — это любой знак, в том числе буква, цифра или знак препинания. **Алфавит** — конечное множество используемых для кодирования информации символов. Стандартный алфавит может быть изменён или дополнен символами. **Ключ** — параметр шифра, определяющий выбор конкретного преобразования данного текста. В современных шифрах криптографическая стойкость шифра це­ликом определяется секретностью ключа (принцип Керкгоффса).

**Шифрование** — процесс нормального применения криптографического преобразования открытого текста на основе алгоритма и ключа, в результате ко­торого возникает шифрованный текст.

**Расшифровывание** — процесс нормального применения криптографиче­ского преобразования шифрованного текста в открытый.

**Асимметричный шифр, двухключевой шифр, шифр с открытым ключом** — шифр, в котором используются два ключа, шифрующий и расшифро­вывающий. При этом, зная лишь ключ зашифровывания, нельзя расшифровать сообщение, и наоборот.

**Открытый ключ** — тот из двух ключей асимметричной системы, который свободно распространяется. Шифрующий для секретной переписки и расшифро­вывающий — для электронной подписи.

**Секретный ключ, закрытый ключ** — тот из двух ключей асимметричной системы, который хранится в секрете. Криптоанализ — наука, изучающая мате­матические методы нарушения конфиденциальности и целостности информации.

**Система шифрования (шифрсистема)** — это любая система, которую мо­жно использовать для обратимого изменения текста сообщения с целью сделать его непонятным для всех, кроме адресата.

**Криптостойкостью** — это характеристика шифра, определяющая его стойкость к дешифрованию без знания ключа (т.е. способность противостоять криптоанализу).

**Криптоаналитик** — учёный, создающий и применяющий методы крипто­анализа. Криптография и криптоанализ составляют криптологию, как единую науку о создании и взломе шифров (такое деление привнесено с запада, до этого в СССР и России не применялось специального деления).

**Криптографическая атака** — попытка криптоаналитика вызвать откло­нения в атакуемой защищённой системе обмена информацией. Успешную криптографическую атаку называют взлом или вскрытие.

**Дешифрование (дешифровка)** — процесс извлечения открытого текста без знания криптографического ключа на основе известного шифрованного. Тер­мин дешифрование обычно применяют по отношению к процессу криптоанализа шифротекста (криптоанализ сам по себе, вообще говоря, может заключаться и в анализе криптосистемы, а не только зашифрованного ею открытого сообщения).

**Криптографическая стойкость** — способность криптографического алго­ритма противостоять криптоанализу.

**Имитозащита** — защита от навязывания ложной информации. Другими словами, текст остаётся открытым, но появляется возможность проверить, что его не изменяли ни случайно, ни намеренно. Имитозащита достигается обычно за счет включения в пакет передаваемых данных имитовставки.

**Имитовставка** — блок информации, применяемый для имитозащиты, за­висящий от ключа и данных.

**Электронная цифровая подпись(электронная подпись)** — асимметри­чная имитовставка (ключ защиты отличается от ключа проверки). Другими словами, такая имитовставка, которую проверяющий не может подделать.

**Центр сертификации** — сторона, чья честность неоспорима, а открытый ключ широко известен. Электронная подпись центра сертификации подтверждает подлинность открытого ключа.

**Хеш-функция** — функция, которая преобразует сообщение произвольной длины в число («свёртку») фиксированной длины. Для криптографической хеш- функции (в отличие от хеш-функции общего назначения) сложно вычислить обратную и даже найти два сообщения с общей хеш-функцией.

1. Введение

Использовать можно в двух системах:

Первый вариант — это выполняются первые 8 работ и получают нужную оценку.

Второй вариант — каждое задание добавляет балы, общая сумма балов определяет итоговую оценку. (Данная система более правильна и гибка, но тре­бует набирать балы за работу)

# Методы защиты информации

**Тема:** Методы защиты информации. Классификация криптосистем.

**Цель:** Изучить простые методы криптографической защиты информации, использовать полученные знания для сокрытия путём шифрования.

## Теоретические ведомости

Появление новых информационных технологий и развитие мощных компьютерных систем хранения и обработки информации повысили уровни защиты информации и вызвали необходимость того, чтобы эффективность защиты информации росла вместе со сложностью архитектуры хранения данных. Постепенно защита информации становится обязательной: разрабатываются всевозможные документы по защите информации; формируются рекомендации; даже проводится ФЗ о защите информации, который рассматривает проблемы и задачи защиты информации, а также решает некоторые уникальные вопросы защиты информации.

Таким образом, угроза защиты информации сделала средства обеспечения информационной безопасности одной из обязательных характеристик информационной системы.

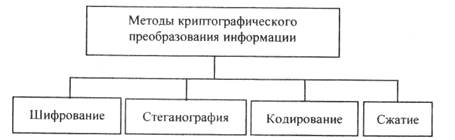


Рисунок 1.1 – Методы криптографического преобразования

### Симметричные криптосистемы

#### Шифры перестановки

В шифрах средних веков часто использовались таблицы, с помощью которых выполнялись простые процедуры шифрования, основанные на перестановке букв в сообщении. Ключом в данном случае является размеры таблицы. Например, сообщение «Неясное становится ещё более непонятным» записывается в таблицу из 5 строк и 7 столбцов по столбцам.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблиця 1.1 | | | | | |
| Н | О | Н | С | Б | Н | | Я |
| Е | Е | О | Я | О | Е | | Т |
| Я | С | В | Е | Л | П | | Н |
| С | Т | И | Щ | Е | О | | Ы |
| Н | А | Т | Ё | Е | Н | | М |

Для получения шифрованного сообщения текст считывается по строкам и группируется по 5 букв:

НОНСБ-НЯЕЕО-ЯОЕТЯ-СВЕЛП-НСТИЩ-ЕОЫНА-ТЕЕНМ

Несколько большей стойкостью к раскрытию обладает метод *одиночной перестановки* по ключу. Он отличается от предыдущего тем, что столбцы таблицы переставляются по ключевому слову, фразе или набору чисел длиной в строку таблицы. Используя в качестве ключа слово - **ЛУНАТИК**, получим следующую таблицу:

Таблиця 1.2 – Метод перестановки по ключу

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Л | У | Н | А | Т | И | К |  | А | И | К | Л | Н | Т | У |
| **4** | **7** | **5** | **1** | **6** | **2** | **3** |  | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| Н | О | Н | С | Б | Н | Я | → | С | Н | Я | Н | Н | Б | О |
| Е | Е | О | Я | О | Е | Т |  | Я | Е | Т | Е | О | О | Е |
| Я | С | В | Е | Л | П | Н | → | Е | П | Н | Я | В | Л | С |
| С | Т | И | Щ | Е | О | Ы |  | Щ | О | Ы | С | И | Е | Т |
| Н | А | Т | Е | Е | Н | М |  | Е | Н | М | Н | Т | Е | А |
| **До перестановки** | | | | | | |  | **После перестановки** | | | | | | |

В верхней строке левой таблицы записан ключ, а номера под буквами ключа определены в соответствии с естественным порядком соответствующих букв ключа в алфавите. Если в ключе встретились бы одинаковые буквы, они бы нумеровались слева направо. Получается шифровка:

СНЯНН-БОЯЕТ-ЕООЕЕ-ПНЯВЛ-СЩОЫС-ИЕТЕН-МНТЕА

Для обеспечения дополнительной скрытности можно повторно шифровать сообщение, которое уже было зашифровано. Для этого размер второй таблицы подбирают так, чтобы длины её строк и столбцов отличались от длин строк и столбцов первой таблицы. Лучше всего, если они будут взаимно простыми.

Кроме алгоритмов одиночных перестановок применяются алгоритмы двойных перестановок. Сначала в таблицу записывается текст сообщения, а потом поочередно переставляются столбцы, а затем строки. При расшифровке порядок перестановок был обратный. Пример данного метода шифрования показан в таблице

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Таблиця 1.3 – Метод перестановки по ключу | | | | | | | | | | | | | | | | |
|  | **2** | **4** | **1** | **3** |  | | **1** | **2** | **3** | **4** |  | | 1 | 2 | 3 | 4 |
| **4** | П | Р | И | Е |  | 4 | И | П | Е | Р |  | **1** | А | З | Ю | Ж |
| **1** | З | Ж | А | Ю | 1 | А | 3 | Ю | Ж | **2** | Е | - | С | Ш |
| **2** | - | Ш | Е | С | 2 | Е | - | С | Ш | **3** | Г | Т | О | О |
| **3** | Т | О | Г | О | 3 | Г | Т | О | О | **4** | И | П | Е | Р |

Ключом к шифру служат номера столбцов 2413 и номера строк 4123 исходной таблицы. В результате перестановки получена шифровка:

АЗЮЖЕ\_СШГТООИПЕР

Число вариантов двойной перестановки достаточно быстро возрастает с увеличением размера таблицы: для таблицы 3х3 их 36, для 4х4 их 576, а для 5х5 их 14400.

В средние века для шифрования применялись и магические квадраты. Магическими квадратами называются квадратные таблицы с вписанными в их клетки последовательными натуральными числами, начиная с единицы, которые дают в сумме по каждому столбцу, каждой строке и каждой диагонали одно и то же число. Для шифрования необходимо вписать исходный текст по приведённой в квадрате нумерации и затем переписать содержимое таблицы по строкам.

Таблиця 1.4 – Исходный текст с идентификаторами

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| П | Р | И | Е | З | Ж | А | Ю | \_ | Ш | Е | С | Т | О | Г | О |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

Таблиця 1.5 – Магический квадрат

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 16 | 3 | 2 | 13 |  | О | И | Р | Т |
| 5 | 10 | 11 | 8 | → | З | Ш | Е | Ю |
| 9 | 6 | 7 | 12 | ← | - | Ж | А | С |
| 4 | 15 | 14 | 1 |  | Е | Г | О | П |

В результате получается шифротекст, сформированный благодаря перестановке букв исходного сообщения.

Число магических квадратов очень резко возрастает с увеличением размера его сторон:

* для таблицы  существует только один квадрат;
* для таблицы ;
* для таблицы .

### Шифры простой замены

#### Система шифрования Цезаря.

Основан на замене каждой буквы сообщения на другую букву того же алфавита, путем смещения от исходной буквы на *K* букв. Известная фраза Юлия Цезаря **VENI VINI VICI** – пришел, увидел, победил, зашифрованная с помощью данного метода, преобразуется в **SBKF SFAF SFZF** (при смещении на 4 символа).

Греческим писателем Полибием за 100 лет до н.э. был изобретён так называемый полибианский квадрат размером 5 x 5, заполненный алфавитом в случайном порядке. Греческий алфавит имеет 24 буквы, а 25-м символом является пробел. Для шифрования на квадрате находили букву текста и записывали в шифротекст букву, расположенную ниже её в том же столбце. Если буква оказывалась в нижней строке таблицы, то брали верхнюю букву из того же столбца.

### Шифры сложной замены

#### Шифр Гронсфельда

Это модификация шифра Цезаря с использованием числовым ключом. Для этого под буквами сообщения записывают цифры числового ключа. Если ключ короче сообщения, то его запись циклически повторяют. Шифротекст получают примерно так же, как в шифре Цезаря, но отсчитывают не третью букву по алфавиту (как в шифре Цезаря), а ту, которая смещена по алфавиту на соответствующую цифру ключа.

1. Пусть в качестве ключа используется группа из трех цифр – 314.
2. Тогда Сообщение СОВЕРШЕННО СЕКРЕТНО.
3. Ключ 3143143143143143143.
4. Шифровка ФПЖИСЬИОССАХИЛФИУСС.

В шифрах *многоалфавитной замены* для шифрования каждого символа исходного сообщения применяется свой шифр простой замены (свой алфавит).

В компьютере операция шифрования соответствует сложению кодов ASCII символов сообщения и ключа по модулю 256.

### Гаммирование

Процесс зашифрования заключается в генерации гаммы шифра и наложении этой гаммы на исходный открытый текст. Перед шифрованием открытые данные разбиваются на блоки одинаковой длины (по 64 бита). Гамма шифра вырабатывается в виде последовательности блоков аналогичной длины  ,  где - побитовое сложение,.

Процесс расшифрования сводится к повторной генерации шифра текста и наложение этой гаммы на зашифрованные данные.

Исходное сообщение из букв русского алфавита преобразуется в числовое сообщение заменой каждой его буквы числом по следующей таблице:

Таблиця 1.6 – Числовая замена букв

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| А | Б | В | Г | Д | Е | Ж | З | И | К | Л | М | Н | О | П |
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 |
| Р | С | Т | У | Ф | Х | Ц | Ч | Ш | Щ | Ь | Ы | Э | Ю | Я |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |



Исходное сообщение ОТДУШКА. Для шифрования числового сообщения используется шифрующий отрезок последовательности А1, А2, ... подходящей длины, начинающийся с, , , , , .

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Исходное сообщение | О | Т | Д | У | Ш | К | А |
| Числовое исходное сообщение | 13 | 17 | 4 | 18 | 23 | 9 | 0 |
| Шифрующий отрезок | 1 | 5 | 6 | 17 | 8 | 19 | 3 |
| Числовое шифрованное сообщение | 14 | 23 | 10 | 5 | 1 | 28 | 3 |
| Шифрованное сообщение | П | Ш | Л | Е | Б | Ю | Г |

Помним, что в выбранном алфавите 30 символов.

## Задания

В соответствии с вашим вариантом из табл. Таблиця 1.7 зашифровать текст используя методы криптографической защиты представленные ниже. Регистр должен быть учтён.

1. Шифры перестановки:
   1. метод перестановки по ключу;
   2. алгоритм двойной перестановки;
   3. магические квадраты.
2. Шифры замены:
   1. шифр Цезаря;
   2. Аффиный шифр;
   3. шифр Виженера;
   4. шифра Плейфера.
3. Выполнить шифрование методом гаммирования.

Записать результаты шифрования в отчёт, сравнить методы, выбрать оптимальный для заданной фразы и аргументировать свой выбор.

## Вопросы для самоконтроля

1. Криптография и её роль в обществе.
2. Объяснить цель и задачи криптографии.
3. Пояснить какие бывают криптографические методы.
4. Что такое шифрование?
5. Как происходит процесс шифровки/дешифровки сообщения?
6. Как наложение гаммы влияет на исходный текст?

Таблиця 1.7 – Список фраз для шифрования

|  |  |  |
| --- | --- | --- |
| 1 | 6 х 6 | небольшое сообщение для тестирования |
| 2 | 3 х 13 | В атмосфере происходит около 1800 гроз. |
| 3 | 7 х 4 | федеральное законодательство |
| 4 | 4 х 6 | Международные стандарты; |
| 5 | 3 х 13 | самая дорогая пицца в мире стоит $1000. |
| 6 | 4 х 9 | применение информационных технологий |
| 7 | 3 х 12 | административный уровень секретности |
| 8 | 3 х 11 | 83% младших братьев выше старших |
| 9 | 3 х 11 | обеспечение доступа к информации |
| 10 | 10 х 4 | Индонезия расположена на 17508 островах. |
| 11 | 4 х 7 | у рыбы сарган зеленые кости. |
| 12 | 8 х 4 | язык хамелеона длиннее его тела |
| 13 | 6 х 4 | защищенность информации. |
| 14 | 5 х 6 | в озеро Байкал впадает 336 рек |
| 15 | 3 х 10 | гоночный болид едет по трассе. |
| 16 | 8 х 3 | опасность ’’открывается” |
| 17 | 4 х 8 | процесс обеспечения целостности |
| 18 | 4 х 9 | рекомендация использования терминов |
| 19 | 5 х 6 | обеспечивающее ее формирование |
| 20 | 5 х 5 | общегосударственный орган |
| 21 | 8 х 4 | технических средств ее передачи |
| 22 | 3 х 11 | ворон и ворона — два разных вида. |
| 23 | 5 х 6 | наибольший ущерб субъектам ИБ |
| 24 | 9 х 3 | информационная безопасность |
| 25 | 8 х 4 | ущерб при сервисном обслуживании |
| 26 | 5 х 7 | свойство аутентичности пользователя |
| 27 | 8 х 4 | данные были действия выполнены?! |
| 28 | 6 х 6 | законодательный уровень безопасности |
| 29 | 8 х 4 | из множества потенциально угроз |
| 30 | 4 х 9 | Защита процессов, процедур, программ |

# Безопасности шифров. Частотная атака

**Тема:** Исследование безопасности шифров. Применение частотного анализа для взлома шифротекста.

**Цель:** Изучить методы взлома криптосистем, провести частотный анализ и расшифровать текст.

## Теоретические ведомости

**Моноалфавитный подстановочный шифр** – шифр, в котором каждой букве исходного алфавита поставлена в соответствие одна буква шифра.

Например, возьмем слово **КУКУРУЗА**. Пусть букве **К** текста соответствует буква А шифра, букве **У** текста соответствует буква Б шифра, букве **Р** текста соответствует буква В шифра, букве **З** текста соответствует буква Г шифра, букве **А** текста соответствует буква Д шифра. После подстановки букв шифра вместо букв исходного теста слово **КУКУРУЗА** в зашифрованном виде будет выглядеть как АБАБВБГД. Недостатком подобного шифрования является то, что, если какая-то буква встречается в исходном тексте чаще всего, то и соответствующая ей буква шифра в зашифрованном тексте также встречается чаще всего. На рисунке **Ошибка! Источник ссылки не найден.** приведены частоты встречаемости букв в английском тексте.

Рисунок 2.1 –Частота использования букв в Английском языке

Для примера мы взяли текст из ресурса Wikipedia[5]. Данный текст предложенный для рассмотрения принципа частотной атаки, с подробным руководством можете ознакомиться на соответствующем ресурсе.

LIVITCSWPIYVEWHEVSRIQMXLEYVEOIEWHRXEXIPFEMVEWHKVSTYLXZIXLIKIIXPIJVSZEYPERRGERIMWQLMGLMXQERIWGPSRIHMXQEREKIETXMJTPRGEVEKEITREWHEXXLEXXMZITWAWSQWXSWEXTVEPMRXRSJGSTVRIEYVIEXCVMUIMWERGMIWXMJMGCSMWXSJOMIQXLIVIQIVIXQSVSTWHKPEGARCSXRWIEVSWIIBXVIZMXFSJXLIKEGAEWHEPSWYSWIWIEVXLISXLIVXLIRGEPIRQIVIIBGIIHMWYPFLEVHEWHYPSRRFQMXLEPPXLIECCIEVEWGISJKTVWMRLIHYSPHXLIQIMYLXSJXLIMWRIGXQEROIVFVIZEVAEKPIEWHXEAMWYEPPXLMWYRMWXSGSWRMHIVEXMSWMGSTPHLEVHPFKPEZINTCMIVJSVLMRSCMWMSWVIRCIGXMWYMX

Рисунок 2.2 – Частотный анализ шифрованного текста

Как видно из графика на рис. Рисунок 2.2 частоты использования символов неясны. Прямой корреляции между символами не наблюдается. Всегда нужно помнить, что частотный анализ не даёт нам полной картины. В каждом тексте соотношение символов будет различное. Так же данный вид взлома будет работать с разными показателями качества для разных языков.

На данном примере мы видим, что по частоте использования в стандарте английского языка преобладает буква **E**. В нашем зашифрованном тексте это может быть буква **I**. Так же можем предположить, что **X** является **T**. Таким образом мы можем продолжить сравнение и получить график на рис. Рисунок 2.3.

Рисунок 2.3 – Результат сравнения частот стандарта с расшифрованном текстом

Hereupon Legrand arose, with a grave and stately air, and brought me the beetle from a glass case in which it was enclosed. It was a beautiful scarabaeus, and, at that time, unknown to naturalists—of course a great prize in a scientific point of view. There were two round black spots near one extremity of the back, and a long one near the other. The scales were exceedingly hard and glossy, with all the appearance of burnished gold. The weight of the insect was very remarkable, and, taking all things into consideration, I could hardly blame Jupiter for his opinion respecting it.

Частотность существенно зависит, однако, не только от длины текста, но и от его характера. Например, в техническом тексте обычно редкая буква **F** может появляться гораздо чаще. Поэтому для надёжного определения средней частоты букв желательно иметь набор различных текстов[2].

Для маскировки частот появления тех или иных букв в тексте используется ***полиалфавитный шифр***. В котором шифрование очередного символа открытого текста согласно некоторому правилу.

## Задания

Проработав материал изучить основные виды взлома. Ознакомившись с теоретическим материалом расшифровать предоставленный текст используя частотный анализ для определения исходного текста. Варианты для задания соответствуют первому номеру в папке «Материалы/ЛР2».

Способы выполнения лабораторной работы:

* 1. используя предоставленное ПО;
  2. использовать альтернативное ПО (без функции автоматической расшифровки текста);
  3. ручным способом.

В результате выполнения лабораторной работы написать отчёт, в котором должны быть предоставлены:

* отрывок шифрованного(исходного) текста;
* данные частотного анализа, предоставлены графики;
* описаны основные этапы выполнение работы;
* предоставлена таблица переменных для замены символов;
* отрывок расшифрованного текста;
* вывод к выполненной работе;

## Ход работы

1. Откройте приложение. Для лучшей обработки файла включена «Подсветка изменений» и «Смена регистров». Их можно выключить во вкладке «Параметры» (рис. Рисунок 2.4).

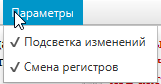
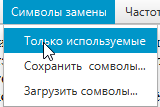
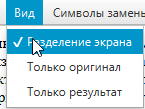


Рисунок 2.4 – Параметры для настройки

1. Выберите меню «Файл», далее пункт «Открыть шифр…» и выберите файл с текстом варианта. После открытия файла появятся вся нужная информация.

Приложение разделено на две области:

* 1. Текстовая (в ней предоставлен исходный/обработанный текст);
  2. Символьная (в ней предоставлены символы замены).

Работа с программой предполагает замену определённых символов на их шифрованные аналоги. В результате данной обработки мы получим исходный текст.

Для корректной замены нужно выполнить следующие шаги (рис. Рисунок 2.5):

1. Выбрать символ для замены.
2. Предоставить заменяющий символ в поле и подтвердить изменение нажав клавишу «Enter».
3. Включить данный символ в активный режим.
4. Нажать кнопку обновить текст.

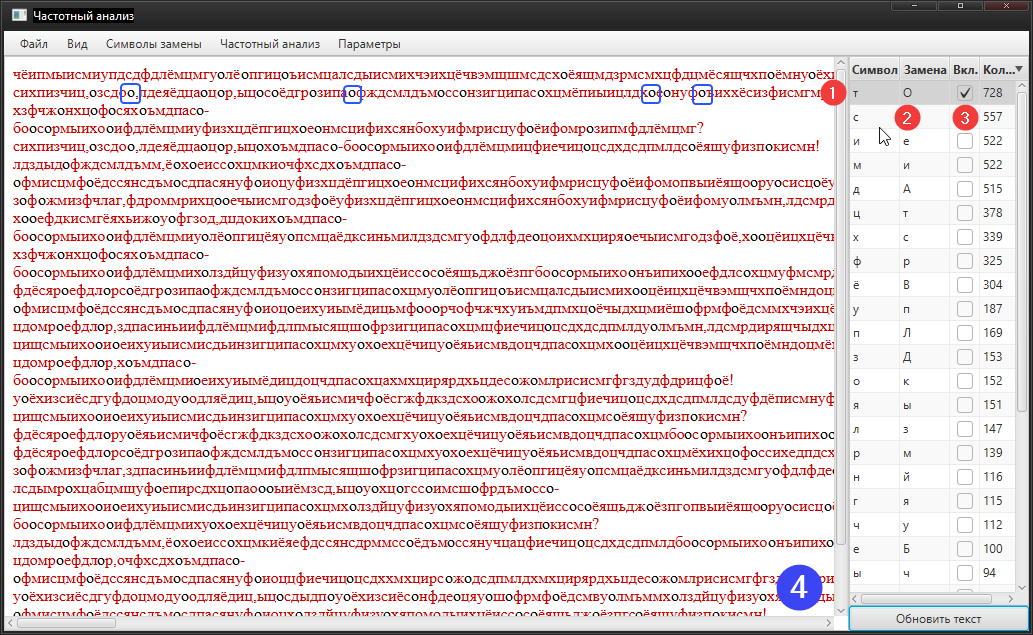


Рисунок 2.5 – Пример замены символа в тексте

1. Проведите частотный анализ открыв соответствующую вкладку в меню (рис. Рисунок 2.6). Запишите в отчёт свои предположения на счёт частотных совпадений.

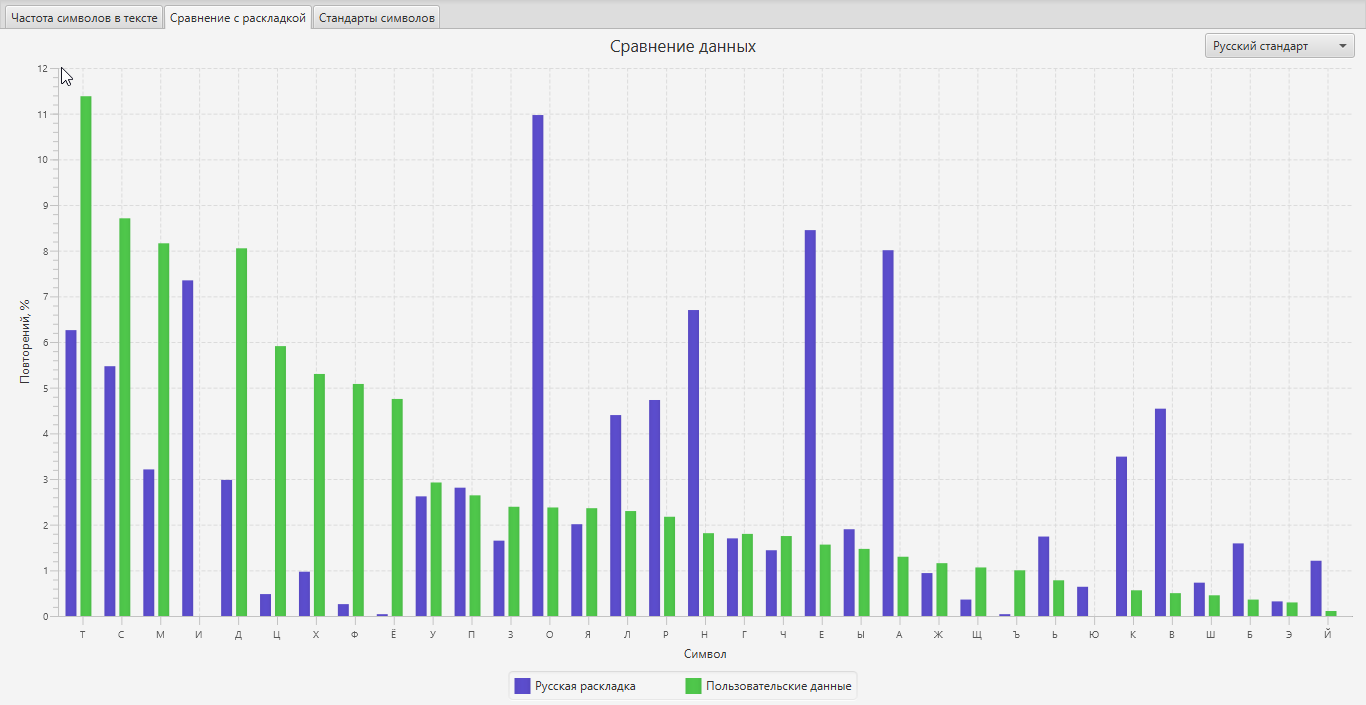


Рисунок 2.6 – Визуальное сравнение частот символов текста и стандарта

1. В результате расшифровки текста (рис. Рисунок 2.7) сохранить его через пункт меню «Файл»-«Сохранить результат…». Символы замены возможно сохранить через «Символы замены»-«Сохранить символы…».
2. Записать данные в отчёт, включая таблицу символов и расшифрованный текст.



Рисунок 2.7 – Результат расшифровки текста

## Вопросы для самоконтроля

1. **Вот тут я хрен знает что писать. Напишу в самом конце для всех работ**

# Потокові шифри. Гамування

**Тема:** Вивчення потокових шифрів. Моделювання роботи скремблера.

**Цель:** Опанувати на практиці застосування режиму однократного гамування. Дослідити побітове неперервне шифрування даних. Ознайомитися з шифруванням інформації за допомогою скремблера.

## Теоретичні відомості

Криптографія являє собою сукупність методів перетворення даних, спрямованих на те, щоб зробити ці дані непотрібними для зловмисника. Такі перетворення дозволяють вирішити два головні питання, що стосуються безпеки інформації:

* захисту конфіденційності;
* захисту цілісності.

Проблеми захисту конфіденційності та цілісності інформації тісно пов'язані між собою, тому методи вирішення однієї з них часто застосовні для вирішення іншої.

### Види криптосистем

Для багатьох термін «криптографія» означає щось загадкове і таємниче. Однак в даний час різні види шифрування можна зустріти буквально всюди - це і прості кодові замки на дипломатів, і багаторівневі системи захисту секретних файлів. Люди стикаються з нею, коли вставляють в банкомат картку, здійснюють грошові перекази, купують через інтернет товари, спілкуються по мессенджерам, відправляють листи на електронну пошту. Будь-які справи, пов'язані з інформацією, так чи інакше мають відношення до криптографії. Але, незважаючи на все різноманіття сфер застосування, в даний час існує всього кілька способів шифрування. Всі ці методи криптографії відносяться до двох видів криптографічних систем: симетричним (з секретним ключем) і асиметричним (з відкритим ключем).

Симетричні системи дозволяють шифрувати і розшифровувати інформацію за допомогою одного і того ж ключа. Розшифрувати криптографічну систему без секретного ключа неможливо, якщо дешифрувальник не володіє секретним ключем.

Стійкість системи цілком залежить від внутрішньої структури генератора ключової послідовності. Якщо генератор видає послідовність з невеликим періодом, то стійкість системи буде невелика. Навпаки, якщо генератор буде видавати нескінченну послідовність істинно випадкових біт, то ми отримаємо «схему одноразового використання» з ідеальною стійкістю.

Реальна стійкість потокових шифрів лежить десь посередині між стійкістю простої моноалфавітної підстановки і «стрічки одноразового використання». Генератор ключової послідовності видає потік бітів, який виглядає випадковим, але в дійсності є детермінованим і може бути в точності відтворений на приймальній стороні. Чим більше генерується потік схожий на випадковий, тим більше зусиль потрібно від криптоаналітика для злому шифру.

### Гамування

Найпростішою і в той же час найбільш надійною з усіх схем шифрування є так звана ***схема одноразового використання*** (рис.4.1), винахід, який найчастіше пов'язують з ім'ям Г.С. Вернама.

***Гамування*** - це накладення (зняття) на відкриті (зашифровані) дані криптографічної гами, тобто послідовності елементів даних, що виробляються за допомогою деякого криптографічного алгоритму, для отримання зашифрованих (відкритих) даних.

З точки зору теорії криптоаналізу, метод шифрування випадковою одноразовою рівноймовірною гамою тієї ж довжини, що і відкритий текст, є таким, що неможливо зламати (далі для стислості будемо вживати термін "одноразове гамування", тримаючи в розумі все сказане вище).

Крім того, навіть розкривши частину повідомлення, дешифрувальник не зможе хоч скільки-небудь поліпшити фінансове становище - інформація про розкриту ділянку гамми не дає інформації про її інші частини [4].

Припустимо, в таємному діловому листуванні використовується метод одноразового накладення гамми на відкритий текст. "Накладення" гамми - не що інше, як виконання операції додавання по модулю 2 (xor) її елементів та елементів відкритого тексту. Ця операція в мові програмування С ++ позначається знаком ^, а в математиці - знаком ⊕.

Стандартні операції над бітами:



Гамування є симетричним алгоритмом. Оскільки подвійне додавання однієї і тієї ж величини по модулю 2 відновлює початкове значення, шифрування і дешифрування виконується однією і тією ж програмою.

Режим шифрування одноразового гамування реалізується наступним чином:

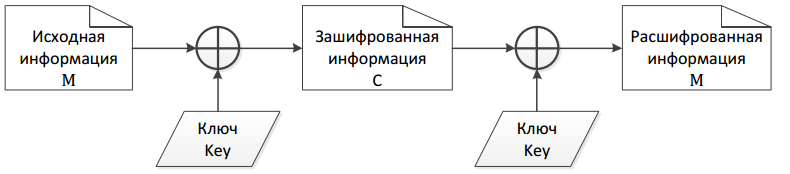


Рисунок 3.1 – Схема однократного гамування Вернама

Завдання знаходження шифротексту при відомому ключі і відкритому тексті полягає в застосуванні наступного правила до кожного символу відкритого тексту:



де  – *i*-й символ отриманого зашифрованого тексту,  – *i*-й символ відкритого тексту,  – *i*-й символ ключа, де . Розмірності відкритого тексту і ключа повинні співпадати, тоді отриманий шифротекст буде такої ж довжини.

Завдання знаходження ключа за відомим шифротекстом і відкритим текстом може бути вирішена, виходячи з (3.1). Для цього потрібно обидві частини рівності додати з по модулю 2:





Таким чином виходять формули для вирішення обох поставлених завдань. Оскільки відкритий текст представлений в символьному вигляді, а ключ - у своєму шістнадцятковому представленні, то у відповідності з таблицею ASCII-кодів можна представити ключ в символьному вигляді. Тоді вже будуть можливі операції (3.1), (3.3), необхідні для вирішення поставлених завдань.

К. Шенноном було доведено, що якщо ключ є фрагментом істинно випадкової двійкової послідовності з рівномірним законом розподілу, причому його довжина дорівнює довжині початкового повідомлення, і використовується цей ключ тільки один раз, після чого знищується, то такий шифр є ***абсолютно стійким***, навіть якщо криптоаналітик володіє необмеженим ресурсом часу і необмеженим набором обчислювальних ресурсів. Дійсно, зловмиснику відомо тільки зашифроване повідомлення , при цьому всі різні ключові послідовності можливі і різновірогідні, а значить, можливі і будь-які повідомлення , тобто **криптоалгоритм не дає ніякої інформації про відкритий текст** [3].

Необхідні і достатні умови абсолютної стійкості шифру:

* повна випадковість ключа;
* рівність довжин ключа і відкритого тексту;
* однократне використання ключа.

#### Приклад шифру абсолютної стійкості

В даному прикладі відображено, як абоненти переписки Аліса та Боб передають зашифроване повідомлення. В їх переписку втрутилася Єва, яка прослуховує всі дані, але не має ключів для дешифровки.

1. Аліса пише повідомлення:

Текст: Буду в 9:00  
 HEX: D0 91 D1 83 D0 B4 D1 83 20 D0 B2 20 39 3A 30 30

Ми можемо зображати текст в шістнадцятковому форматі (HEX), тоді англійська буква буде відповідати парі, а російська – чотирьом символам.

Ключ: Ключ Message  
 HEX: D0 9A D0 BB D1 8E D1 87 20 4D 65 73 73 61 67 65

1. Шифрує його за допомогою стрічки однократного використання.

Ключ: 8: ޗSJ[WU  
Шифр: 00 0B 01 38 01 3A 00 04 00 9D D7 53 4A 5B 57 55

1. Відправляє шифротекст через канал зв’язку.
2. Боб, прослуховуючи канал зв’язку, зчитує шифротекст:

Шифр: 00 0B 01 38 01 3A 00 04 00 9D D7 53 4A 5B 57 55

1. Підставляючи вірну гамму:

Гамма: Ключ Message  
 HEX: D0 9A D0 BB D1 8E D1 87 20 4D 65 73 73 61 67 65

1. В результаті Боб отримує розшифроване повідомлення Аліси:

HEX: D0 91 D1 83 D0 B4 D1 83 20 D0 B2 20 39 3A 30 30  
Текст: Буду в 21:00

Припустимо, що в зв’язок втрутилася Єва. Вона може тільки прослуховувати канал, не змінюючи дані. Як тільки Аліса відправила дані Бобу, Єва отримала зашифрований текст:

Шифр: 00 0B 01 38 01 3A 00 04 00 9D D7 53 4A 5B 57 55

Але не знаючи вірної гами, вона не зможе розшифрувати текст. Припустимо, що вона знайшла один із варіантів ключів:

Ключ: Ключ Mes}age  
 HEX: D0 9A D0 BB D1 8E D1 87 20 4D 65 73 7d 61 67 65

В результаті чого здійснила дешифрування тексту:

HEX: D0 91 D1 83 D0 B4 D1 83 20 D0 B2 20 37 3A 30 30  
Текст: Буду в 7:00

Пробуючи нові ключі, вони будуть бачити всілякі можливі тексти заданої довжини.

Режим шифрування одноразового гамування одним ключем двох видів відкритого тексту реалізується в такий спосіб:

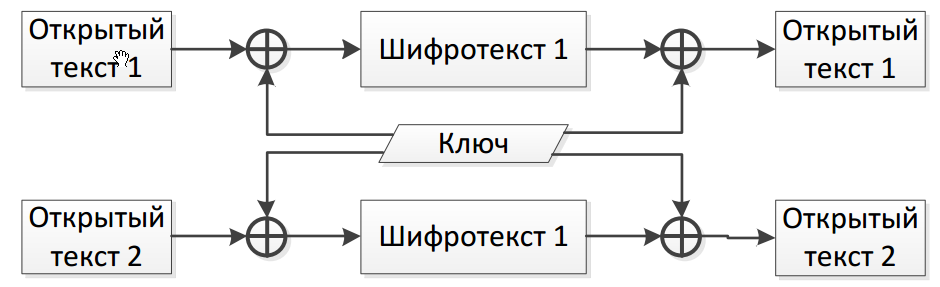


Рисунок 3.2 – Загальна схема шифрування двох різних текстів одним ключем

За допомогою формул режиму однократного гамування отримуються шифротексти обох телеграм:



Завдання знаходження відкритого тексту за відомим шифротекстом двох телеграм, зашифрованих одним ключем, може бути вирішено, використовуючи (3.4). Для цього потрібно скласти по модулю 2 обидві рівності (3.4). З огляду на такі властивості операції XOR, як



можна отримати



Припустимо, що одна з телеграм є "рыбой", тобто має фіксований формат, в який вписуються значення полів, і зловмисникові достеменно відомий цей формат. Тоді він отримує досить багато пар  (відомий вид обох шифротекстів) і, припустимо, . Тоді з урахуванням (3.5) виходить:



Таким чином, зловмисник отримує можливість визначити ті символи повідомлення , які знаходяться на позиціях відомої " рыбы" повідомлення . Здогадуючись за логікою повідомлення , зловмисник має реальний шанс дізнатися ще кілька символів повідомлення . Потім він може використовувати (3.7), замість , підставляючи раніше визначені символи повідомлення . І так далі. Діючи таким чином, зловмисник якщо навіть не прочитає обидва повідомлення, то значно зменшить простір їх пошуку.

### Скремблер

Скремблером називається програмна або апаратна реалізація алгоритму, що дозволяє побітово шифрувати безперервні потоки інформації.

Розглянемо зсувний регістр зі зворотним зв'язком (Linear Feedback Shift Register, скорочено LFSR) - логічний пристрій, схема якого показана на рисунку 3.3.

Зсувний регістр представляє собою послідовність біт. Кількість біт визначається довжиною зсувного регістру. Якщо довжина дорівнює п біт, то регістр називається n-бітовим зсувними регістром. Всякий раз, коли потрібно витягти біт, всі біти зсувного регістру зсуваються вправо на 1 позицію. Новий крайній лівий біт є функцією всіх інших бітів регістра. На виході зсувного регістру опиняється молодший значущий біт [3].

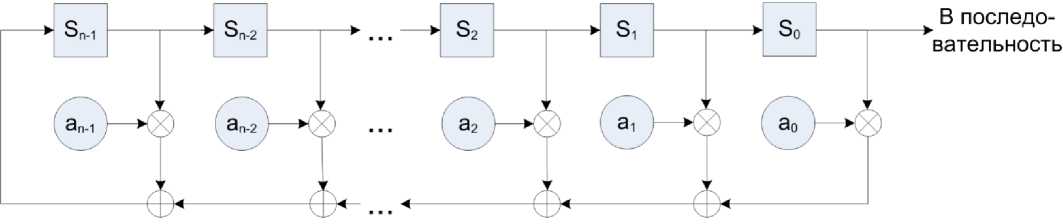


Рисунок 3.3 – Схема LFSR

LFSR складається з *n* комірок пам'яті, двійкові стани яких в моменти часу LFSR  характеризуються значеннями. Виходи комірок пам’яті пов’язані не тільки послідовно один з одним, але і з суматорами 0 у відповідності з коефіцієнтами передачі : якщо , то значення  *i*-ї комірки передається на один із входів суматора; якщо ж  , то така передача відсутня. Зазвичай коефіцієнти передачі задаються за допомогою полінома:



Стан LFSR в поточний момент часу t задається двійковим n-вектор-стовпцем .

Зміст комірок LFSR з плином часу змінюється наступним чином, визначаючи тим самим динаміку станів LFSR:



Поточні значення нульової комірки регістра використовуються в якості елементів LFSR двійкової квазівипадкової послідовності (див. рисунок 3.3).

Дана послідовність витягнутих біт повинна володіти трьома властивостями:

##### Збалансованість:

для кожного інтервалу послідовності кількість двійкових одиниць повинно відрізнятися від числа двійкових нулів не більш, ніж на кілька відсотків від їх загальної кількості на інтервалі.

##### Циклічність:

безперервну послідовність однакових двійкових чисел називають циклом. Поява іншої двійкової цифри автоматично починає новий цикл. Довжина циклу дорівнює кількості однакових цифр в ньому. Необхідно, щоб половина всіх «смужок» (поспіль ідентичних компонентів послідовності) мала довжину 1, одна четверта - довжину 2, одна восьма - довжину 3, і т.д.

##### Кореляція:

якщо частина послідовності і її циклічно зсунута копія поелементно порівнюються, бажано, щоб число збігів відрізнялося від числа розбіжностей не більше, ніж на кілька відсотків від довжини послідовності.

При досить довгій роботі скремблера неминуче виникає його зациклення. За виконання певного числа тактів в осередках скремблера створиться комбінація біт, яка в ньому вже одного разу виявлялася, і з цього моменту шифруюча послідовність почне циклічно повторюватись з фіксованим періодом. Дана проблема непереборна за своєю природою, так як в *N* розрядах скремблера не може перебувати більше  комбінацій біт, і, відповідно, максимум через  ітерації циклу повтор комбінації неодмінно відбудеться. Послідовність біт, що генерується таким скремблером, називається ***послідовністю найбільшої довжини***.

Щоб побудувати N-розрядний скремблер, що створює послідовність максимальної довжини, користуються примітивними багаточленами. **Примітивний (базовий) багаточлен** **ступеня** *n* по модулю 2 - це неприводимий многочлен, який є дільником, але не є дільником  для всіх *d*, на які ділиться  . ***Неприводимий многочлен ступеня*** (*n*) не можна представити у вигляді множення ніяких інших многочленів, крім нього самого і одиничного.

Знайдений примітивний багаточлен ступеня записується в двійковому вигляді, потім відкидається одиниця, що відповідає найстаршому розряду.

Наведемо приклад 7-розрядного скремблера [1], що генерує послідовність з періодом, що дорівнює . Нехай початкове значення стану буде рівне . Для цього регістра зсуву новий біт генерується за наступною схемою:

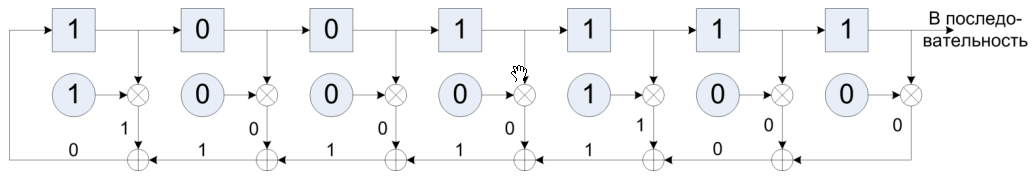


Рисунок 3.4 – Схема LFSR для багаточлена 

Код для цього LFSR на мові C виглядає наступним чином:

|  |  |
| --- | --- |
|  | **int LFSR ()**  **{**  **/\* Начальное состояние \*/**  **static unsigned long ShiftRegister = 79;**  **ShiftRegister =**  **((((ShiftRegister >> 6) ^ (ShiftRegister >> 2)) & 0x01) << 6) | (ShiftRegister >> 1);**  **return ShiftRegister & 0x01;**  **}** |

Послідовність зміни стану скремблера має вигляд:

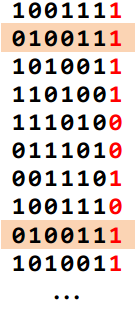


Рисунок 3.5 – Зміна стану скремблера 

Як видно з рисунка 3.5, період послідовності, що генерується скремблером, дорівнює 7. Квазівипадкова послідовність, яка використовується в якості гами:.

## Завдання

**Частина перша.**

Завдання не передбачають використання додаткового ПЗ, виконання завдань математичного розрахунку повинні бути виконані вручну:

1. 1) На основі варіантів з табл. 3.1 визначити поліном скремблера і його початкове значення.
2. Скласти схему скремблера.
3. Скласти математичну модель роботи скремблера для даних розміром 16-біт.
4. Продемонструвати роботу скремблера з минулого пункту на прикладі випадкової послідовності.

**Частина друга.**

Використовуючи будь-яку мову програмування, будь то C ++, Java, C #, JS, Matlab або модель в Simulink. Реалізувати додаток для шифрування/ дешифрування повідомлень:

1. Текст, що шифрується (із табл. Таблиця 1.7), повинен зберігатися в файлі.
2. Ключ шифрування повинен задаватися випадковим чином. Також передбачити можливість ручної зміни ключа.
3. Зашифрований текст повинен зберігатися в один файл, а ключ, що був використаний при шифруванні – відображатися на екрані або в зберігатися в окремий файл.
4. Відобразити час шифрування скремблера.
5. Дослідити послідовність на властивості збалансованості, циклічності, кореляції.

**Звіт повинен містити:** дані варіанта, схема скремблера, математичну модель, демонстрацію роботи, реалізацію алгоритму шифрування, а також результати аналізу властивостей і висновки до роботи.

## Варіанти

Таблиця 3.1 – Варіанти скремблерів

|  |  |  |
| --- | --- | --- |
| **Варіант** | **Скремблери** | |
| 0 |  |  |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |

## Питання для самоконтролю

1. Переваги та недоліки однократного гамування.
2. Чому розмірність відкритого тексту повинна співпадати з ключем?
3. Як по відкритому тексту і шифротексту отримати ключ?
4. Необхідні та достатні умови абсолютної стійкості шифру.
5. Як, знаючи текст одного з повідомлень(или ), визначити інше, не намагаючись визначити ключ?
6. Переваги та недоліки використання скремблера.
7. Властивості, якими повинна володіти псевдовипадкова послідовність, що генерується скремблером.

# Симметричные криптосистемы. Блочные шифры

**Тема:** Исследование симметричных методов шифрование.

**Цель:** Изучить основные типы шифров. Получить практические навыки в написании и понимании симметричных методов шифровки сообщений.

## Теоретические ведомости

### Блочные шифры.

Это функция шифрования, которая применяется к блокам текста фиксированной длины. Текущее поколение блочных шифров работает с блоками текста длиной 256 бит (32 байт). Такой шифр принимает на вход 256-битовый открытый текст и выдаёт 256-битовый шифрованный текст.

Блочный шифр является обратимым: существует функция дешифрования, которая принимает на вход 256-битовый шифрованный текст и выдаёт исходный 256-битовый открытый текст. Открытый и шифрованный текст всегда имеет один и тот же размер, который называется размером блока (block size) round N-l output (64 bits)

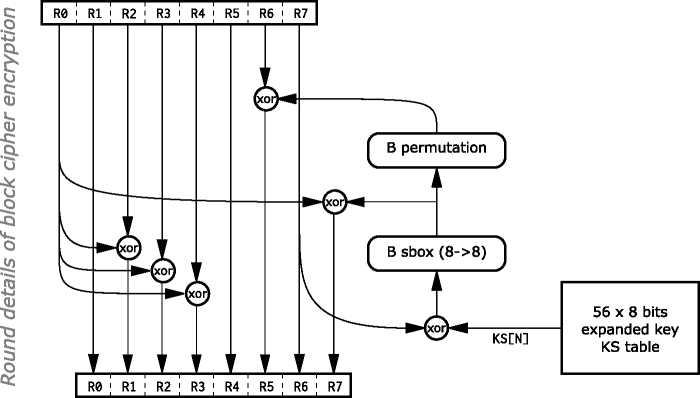


Рисунок 4.1 – Подробный разбор алгоритма блочного скремблирования DVB

### Сферы применения криптографии.

В настоящее время криптография прочно вошла в нашу жизнь. Перечислим лишь некоторые сферы применения криптографии в современном информатизированном обществе:

* шифрование данных при передаче по открытым каналам связи  
  (например, при совершении покупки в Интернете сведения о сделке);
* обслуживание банковских пластиковых карт;
* хранение и обработка паролей пользователей в сети;
* сдача бухгалтерских и иных отчётов через удалённые каналы связи;
* банковское обслуживание предприятий через локальную/глобальную сеть;
* безопасное от несанкционированного доступа хранение данных на жёстком диске компьютера.

Средства обеспечения информационной безопасности можно разбить на четыре группы:

* организационные средства (действия общего характера, предпринимаемые руководством организации, имеющие дело с людьми);
* законодательные средства (стандарты, законы, нормативные акты и т.д.);
* программно-аппаратные средства (системы идентификации и аутентификации; системы шифрования дисковых данных;
* системы аутентификации электронных данных и т.д.);
* криптографические средства (электронная цифровая подпись, шифрования, аутентификация и др.).

## Задания

Выполнение первой работы предусматривает две части. Нужно реализовать шифровку ручными методами, не используя ПО. Вторая часть предусматривает использование специализированного ПО. После выполнения обеих частей требуется сравнить результате, записать выводы.

Выполнение действий должно сопровождаться соответствующими заметками в отчёте:

1. Данные соответствующие варианту из таблицы Таблиця 4.1.
2. Описание алгоритма, заданного вариантом.
3. Блок-схема алгоритма с описанием блоков.
4. Математическая модель алгоритма. Шифрование/дешифровка сообщения по варианту из таблицы Таблиця 1.7.
5. Результаты шифрования сообщения заданным ключом.

Для получения максимального балла требуется выполнить дополнительные задания:

1. Написать алгоритм шифровки-дешифровки заданным способом. Алгоритм должен быть представлен в виде подробной блок-схемы.

## Ход работы

1. Запишите свои ФИО и группу на английском языке:

My name: Ivanov Alexander | My group: SYA-14-2

1. Выберите ключ для шифрования, в нашем случае это будет:

MyVariant0

1. Следующим шагом выберите алгоритм для шифрования. В нашем случае это будет
2. Составить шаблон для шифрования.
3. Выбор методов шифрования. Зашифровать для начала ручным способом, после чего написать алгоритм автоматической шифровки сообщения на любом языке. Для примера можно использовать один из следующих алгоритмов:
   * + AES (американский стандарт шифрования);
     + ГОСТ 28147-89 (российский стандарт шифрования);
     + DES (стандарт шифрования данных в США);
     + 3DES (тройной DES);
     + RC2 (Шифр Ривеста);
     + RC5;
     + Другие блочные алгоритмы…
4. Инструкция
5. Выбрать данные из таблицы. Разложить данные
6. Расшифровать криптотекст требуемым методом.
7. Что записать в отчёт.

## Вопросы для самоконтроля

1. Отличие симметричных и асимметричный шифров.
2. Пояснить что такое исходный текст, шифр, ключ.
3. Принцип подбора ключа в симметричных криптосистемах.
4. Принцип работы симметричных шифров. Приведите примеры.
5. Различие между блочным и потоковым шифрованием.

Таблиця 4.1 – Варианты для блочного шифрования

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |

# Линейный криптоанализ

**Тема:** Изучение метода криптоаналитического вскрытия.

**Цель:** Опишите цель

## Теоретические ведомости

### Оценка надёжности шифров

Методы оценки качества криптоалгоритмов, используемые на практике:

1. всевозможные попытки их вскрытия;
2. анализ сложности алгоритма дешифрования;
3. оценка статистической безопасности шифра.

В первом случае многое зависит от квалификации, опыта, интуиции криптоаналитика и от правильной оценки возможностей противника. Обычно считается, что противник знает шифр, имеет возможность его изучения, знает некоторые характеристики открытых защищаемых данных, например тематику сообщений, их стиль, стандарты, форматы и т. п. Рассмотрим следующие примеры возможностей противника:

1. противник может перехватывать все зашифрованные сообщения, но не имеет соответствующих им открытых текстов;
2. противник может перехватывать все зашифрованные сообщения и добывать соответствующие им открытые тексты;
3. противник имеет доступ к шифру (но не ключам!) и поэтому может зашифровывать и расшифровывать любую информацию.

Во втором случае оценку стойкости шифра заменяют оценкой минимальной сложности алгоритма его вскрытия. Однако получение строго доказуемых оценок нижней границы сложности алгоритмов рассматриваемого типа не представляется возможным. Иными словами, всегда возможна ситуация, когда алгоритм вскрытия шифра, сложность которого анализируется, оказывается вовсе не самым эффективным.

Сложность вычислительных алгоритмов можно оценивать числом выполняемых элементарных операций, при этом, естественно, необходимо учитывать их стоимость и затраты на их выполнение. В общем случае это число должно иметь строгую нижнюю оценку и выходить за пределы возможностей современных компьютерных систем. Качественный шифр невозможно раскрыть способом более эффективным, чем полный перебор по всему ключевому пространству, при этом криптограф должен рассчитывать только на то, что у противника не хватит времени и ресурсов, чтобы это сделать.

Алгоритм полного перебора по всему ключевому пространству это пример так называемого экспоненциального алгоритма. Если сложность алгоритма выражается неким многочленом (полиномом) от п, где п - число элементарных операций, такой алгоритм носит название полиномиального.

В третьем случае считается, что надежная криптосистема с точки зрения противника является «чёрным ящиком», входная и выходная информационные последовательности которого взаимно независимы, при этом выходная зашифрованная последовательность является псевдослучайной. Поэтому смысл испытаний заключается в проведении статистических тестов, устанавливающих зависимость изменений в зашифрованном тексте от изменений символов или битов в исходном тексте или ключе, а также анализирующих, насколько выходная зашифрованная последовательность по своим статистическим свойствам приближается к истинно случайной последовательности. Случайность текста шифровки можно приближённо оценивать степенью её сжатия при использовании алгоритма Лемпела-Зива, применяемого в архиваторах IBM PC. Если степень сжатия больше 10%, то можно считать криптосистему несостоятельной.

## Задания

## Ход работы

## Вопросы для самоконтроля

# Асимметричные шифры. Часть 1

**Тема:** Тема.

**Цель:** Опишите цель

## Теоретические ведомости

#### Асимметричные криптосистемы.

В криптографических системах с открытым ключом пользователи обладают собственным открытым и частным закрытым ключами. К открытому ключу имеют доступ все пользователи, и информация шифруется именно с его помощью. А вот для расшифровки необходим частный ключ, находящийся у конечного пользователя. В отличие от криптограмм с секретным ключом в такой системе участниками являются не две, а три стороны. Третья может представлять собой сотового провайдера или, например, банк. Однако эта сторона не заинтересована в хищении информации, поскольку она заинтересована в правильном функционировании системы и получении положительных результатов

## Задания

## Ход работы

## Вопросы для самоконтроля

# Асимметричные шифры. Часть 2

**Тема:** Тема.

**Цель:** Опишите цель

## Теоретические ведомости

### Криптографические протоколы

В современной криптографии большое внимание уделяется не только созданию и исследованию шифров, но и разработке криптографических протоколов.

**Протокол** – это последовательность шагов, которые предпринимают две или большее количество сторон для совместного решения задачи. Все шаги следуют в порядке строгой очередности, и ни один из них не может быть сделан прежде, чем закончится предыдущий. Кроме того, любой протокол подразумевает участие, по крайней мере, двух сторон.

**Криптографический протокол** – это такая процедура взаимодействия двух или более абонентов с использованием криптографических средств, в результате которой абоненты достигают своей цели, а их противники - не достигают. В основе протокола лежит набор правил, регламентирующих использование криптографических преобразований и алгоритмов в информационных процессах. Каждый криптографический протокол предназначен для решения определённой задачи.

Рассмотрим простейший протокол для обмена конфиденциальными сообщениями между двумя сторонами, которые будем называть абонент №1 и абонент №2. Пусть абонент №1 желает передать зашифрованное сообщение абоненту №2. В этом случае их последовательность действий должна быть следующей.

1. Абоненты выбирают систему шифрования (например, шифр Цезаря).
2. Абоненты договариваются о ключе шифрования.
3. Абонент №1 шифрует исходное сообщение с помощью ключа выбранным методом и получает зашифрованное сообщение.
4. Зашифрованное сообщение пересылается абоненту №2.
5. Абонент №2 расшифровывает зашифрованное сообщение с помощью ключа и получает открытое сообщение.

Этот протокол достаточно прост, однако он может действительно использоваться на практике. Криптографические протоколы могут быть простыми и сложными в зависимости от назначения.

### 

## Задания

## Ход работы

## Вопросы для самоконтроля

# Электронно-цифровая подпись

**Тема:** Тема.

**Цель:** Опишите цель

## Теоретические ведомости

## Задания

## Ход работы

## Вопросы для самоконтроля

1. Перелік використаних джерел

1. НГТУ Гаммирование. Моделирование работы скремблера // Лабораторные работы. 2012. C. 7.

2. НОУ «ИНТУИТ» Лекция 4: Методы криптоанализа // 03.2015 [Электронный ресурс]. URL: https://www.intuit.ru/studies/courses/600/456/ lecture/10198 (дата обращения: 08.04.2018).

3. Столлингс В. Криптография и защита сетей. Принципы и практика / В. Столлингс, М.: Издательский дом «Вильямс», 2001. 672 c.

4. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си / Б. Шнайер, М.: Триумф, 2002. 816 c.

5. Wiki inform // 08.01.2018 [Электронный ресурс]. URL: https://en.wikipedia.org/wiki/Frequency\_analysis (дата обращения: 07.04.2018).

1. Додаток Б