МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Чувашский государственный университет имени И.Н. Ульянова»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

Лабораторная работа №1

По дисциплине “Защита информация”

на тему:

«Шифрование данных методом подстановки»

Вариант 6

Выполнил: ст. гр. ИВТ-11-21

Еремеев С. А.

Проверил: канд. техн. наук. Ковалев С.В.

Чебоксары 2024

## Теоретическая часть

## Цель работы

Целью работы является знакомство с классическим криптографическим алгоритмом - алгоритмом шифрования данных при помощи подстановки.

## Основные сведения

В современной криптографии рассматриваются два типа криптографических алгоритмов. Это классические криптографические алгоритмы, основанные на использовании секретных ключей, и новые криптографические алгоритмы с открытым ключом, основанные на использовании ключей двух типов: секретного (закрытого) и открытого.

В классической криптографии ("криптографии с секретным ключом" или "одноключевой криптографии") используется только одна единица секретной информации - ключ, знание которого позволяет отправителю зашифровать информацию в шифртекст, а получателю - расшифровать его. Операция шифрования/дешифрования с большой вероятностью невыполнима без знания секретного ключа. Поскольку при использовании классических криптографических алгоритмов ключ шифрования и ключ дешифрования совпадают и такие криптосистемы называются симметричными.

Подстановочное шифрование основывается на использовании некоторой взаимно однозначной функции , где *V* - алфавит шифруемых сообщений, *m* - длина блока открытого текста и блока шифрограммы. В процессе шифрования открытый текст X разбивается на *m*‑символьные блоки *x1*, *x2*, ..., *xl*, каждый из которых заменяется *m*‑символьным блоком . Дешифрование сводится к обратной замене *m*‑символьных блоков *y* на *m*-символьные блоки .

Например, пусть алфавит *V* = {0,1, .., 9, <пробел>, A, B, .., Z, a, b, .., z, А, Б, .., Я, а, б, .., я}, длина блока шифрограммы и блока открытого текста *m* = 3. Допустим, что необходимо зашифровать открытый текст X = “Произвольный блок открытого текста”. Разобьем открытый текст X на *m*-символьные (трехсимвольные в нашем примере) блоки: “Про”, “изв”, “оль”, “ный”, “\_бл”, “ок\_“, “отк”, “рыт”, “ого”, “\_те”, “кст”, “а\_\_”. Пробелы обозначены символом \_, при необходимости последний блок может быть дополнен с правой стороны необходимым количеством пробелов. Если пробел не входит в алфавит языка, то его функцию (функцию разделительного элемента) может выполнять любой другой символ алфавита, если стороны, обменивающиеся сообщениями, достигли соответствующей договоренности.

Для шифрования необходимо иметь функцию *CV*, ставящую каждому трехсимвольному блоку открытого текста трехсимвольный блок шифртекста. Такая функция может быть задана, например, при помощи таблицы:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *xi* | Про | изв | оль | ный | \_бл | ок\_ | отк | рыт | ого | \_те | кст | а\_\_ | … |
| *CV(xi)* | Атр | ф7ы | нрв | св\_ | ркк | ыт0 | мкф | ц\_й | 1ся | щн\_ | ы34 | вхш | … |

Каждый блок открытого текста заменяется при помощи функции *CV* соответствующим блоком шифртекста. Таким образом, для рассматриваемого примера шифртекст будет выглядеть следующим образом: “Атрф7ынрвсв\_рккыт0мкфц\_й1сящн\_ы34вхш”.

Поскольку функция *CV* является взаимно однозначной, эта же таблица используется и для дешифрации шифртекста.

Очевидно, что приведенные в этом примере алфавит и принятый размер блока открытого текста требуют очень большой таблицы, задающей функцию шифрования: эта таблица должна задавать все возможные трехсимвольные сочетания из русских и латинских букв, а также цифр. Если в качестве алфавита рассматривать двоичный алфавит {0, 1}, а размер блока открытого текста принять равным 7 (как это имеет место в случае обычного ASCII-кода), то для задания функции шифрования требуется таблица со 128 столбцами. В общем случае, требуется определить  значений функции, где  - мощность множества *V*, то есть количество элементов алфавита. Разумеется, если заранее известно, что некоторые комбинации символов открытого текста являются недопустимыми, то указанное значение может быть уменьшено.

**Практическая часть**

## Индивидуальное задание

| Задание | | | Для информации | |
| --- | --- | --- | --- | --- |
| № п.п. | *V* | *m* |  |  |
| 1 | {0,1} | 5 | 2 | 32 |

**Функция шифрования**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *xi* | А | Б | В | Г | Д | Е | Ё | Ж | З |
| *CV(xi)* | Я | Ю | Э | Ь | Ы | Щ | Ш | Ч | Ц |
|  | | | | | | | | | |
| *xi* | И | Й | К | Л | М | Н | О | П | Р |
| *CV(xi)* | Х | Ф | У | Т | С | Р | П | О | Н |
|  | | | | | | | | | |
| *xi* | С | Т | У | Ф | Х | Ц | Ч | Ш | Щ |
| *CV(xi)* | М | Л | К | Й | И | З | Ж | Ё | Е |
|  | | | | | | | | | |
| *xi* | Ы | Ь | Э | Ю | Я |  | | | |
| *CV(xi)* | Д | Г | В | Б | А |

Функция шифрования ставит каждому символьному блоку открытого текста символьный блок шифртекста. Эта же таблица используется и для дешифрации шифртекста.

**Текст программы, реализующей индивидуальное задание**

Сначала заданы две коллекции, первая из которых будет содержать символы алфавита, а вторая содержит алфавит шифртекста. Далее создаем функцию EncrypText(), которая будет преобразовывать текст.

Полный текст программы:

using System.Globalization;

List<char> originalLetters = new List<char>{

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

};

List<char> encryptedLetters = new List<char>{

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

};

Console.Write("Введите текст: ");

string text = Console.ReadLine()?.ToUpper() ?? string.Empty;

static string EncrypText(string text, List<char> originalLetters, List<char> encryptedLetters)

{

char[] encryptedText = new char[text.Length];

for (int i = 0; i < text.Length; i++)

{

char currentChar = text[i];

int index = originalLetters.IndexOf(currentChar);

encryptedText[i] = encryptedLetters[index];

}

return new string(encryptedText);

}

string text2 = EncrypText(text, originalLetters, encryptedLetters);

Console.WriteLine($"Исходный текст: {text}");

Console.WriteLine($"Зашифрованный текст: {text2}");

**Пример открытого текста и соответствующей ему шифрограммы**

Открытый текст: ПРИВЕТ МИР

Шифртекст: ОНХЭЩЛ\_СХН

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *xi* | П | Р | И | В | Е | Т |  | М | И | Р |
| *CV*  *(xi)* | О | Н | Х | Э | Щ | Л | \_ | С | Х | Н |

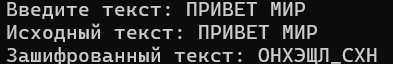


Рисунок –Пример работы программы

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *xi* | Е | Р | Е | М | Е | Е | В |  |  |  |
| *CV*  *(xi)* | Щ | Н | Щ | С | Щ | Щ | Э |  |  |  |

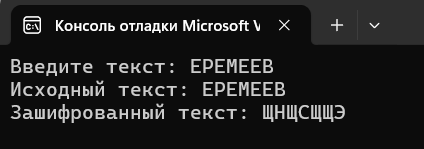


Рисунок 2 –Пример работы программы

**Вывод**

Ознакомился с классическим криптографическим алгоритмом – алгоритмом шифрования данных при помощи подстановки