Министерство образования Республики Беларусь

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Отчет к лабораторной работе № 2:

«Исследование криптографических шифров на основе перестановки симвалов

Выполнила:

студентка 3 курса 2 группы

Шастовская М. С.

Вариант: 12

Преподаватель:

Сазонова Д.В.

Минск 2023

**Цель**: изучение и приобретение практических навыков разработки и использования приложений для реализации перестановочных.

**Задачи**:

* Закрепить теоретические знания по алгебраическому описанию, алгоритмам реализации операций зашифрования/расшифрования и оценке криптостойкости перестановочных шифров.
* Ознакомиться с особенностями реализации и свойствами различных перестановочных шифров на основе готового программного средства (L\_LUX).
* Разработать приложение для реализации указанных преподавателем методов перестановочного зашифрования/расшифрования.
* Выполнить исследование криптостойкости шифров на основе статистических данных о частотах появления символов в исходном и зашифрованном сообщениях.
* Оценить скорость зашифрования/расшифрования реализованных способов шифров.
* Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения эксперимента с использованием приложения и результатов эксперимента.

1 Теоретические сведения

Сущность перестановочного шифрования состоит в том, что исходный текст (М) и зашифрованный текст (С) основаны на использовании одного и того же алфавита, а тайной или ключевой информацией является алгоритм перестановки.

Шифры перестановки относятся к классу симметричных. Элементами текста могут быть отдельные символы (самый распространенный случай), пары, тройки букв и т. д.

Классическими примерами перестановочных шифров являются анаграммы. Анаграмма (от греч. ana – снова и gramma – запись) – литературный прием, состоящий в перестановке букв (или звуков), что в результате дает другое слово или словосочетание, например: проездной – подрезной, листовка – вокалист, апельсин – спаниель.

В классической криптографии шифры перестановки делятся на два подкласса:

* Шифры простой, или одинарной, перестановки – при зашифровании символы открытого текста Мi перемещаются с исходных позиций в новые (в шифртексте Сi) один раз.
* Шифры сложной, или множественной, перестановки – при зашифровании символы открытого текста Мi перемещаются с исходных позиций в новые (в шифртексте Сi) несколько раз.

2 Практическая часть

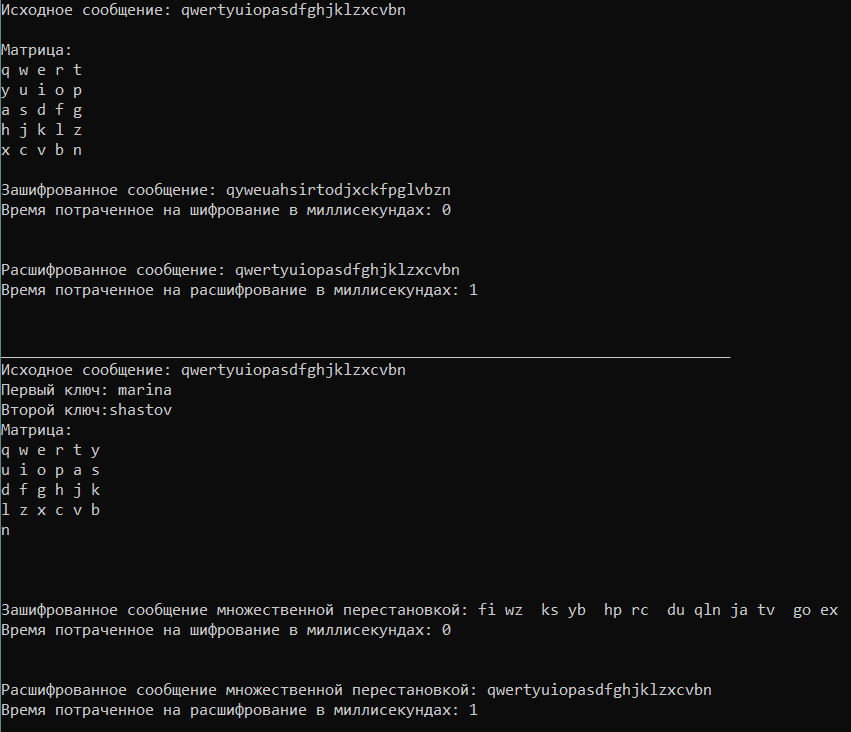


Рисунок 2.1 – Результат шифрования/дешифрования исходного сообщения методами маршрута змейкой и множественной перестановкой

На данном скриншоте отображен процесс работы программы с шифрованием и дешифрованием исходного сообщения методами маршрута змейкой и множественной перестановкой. Так же выведены вспомогательные средства: затраченное время, таблицы и прочее.

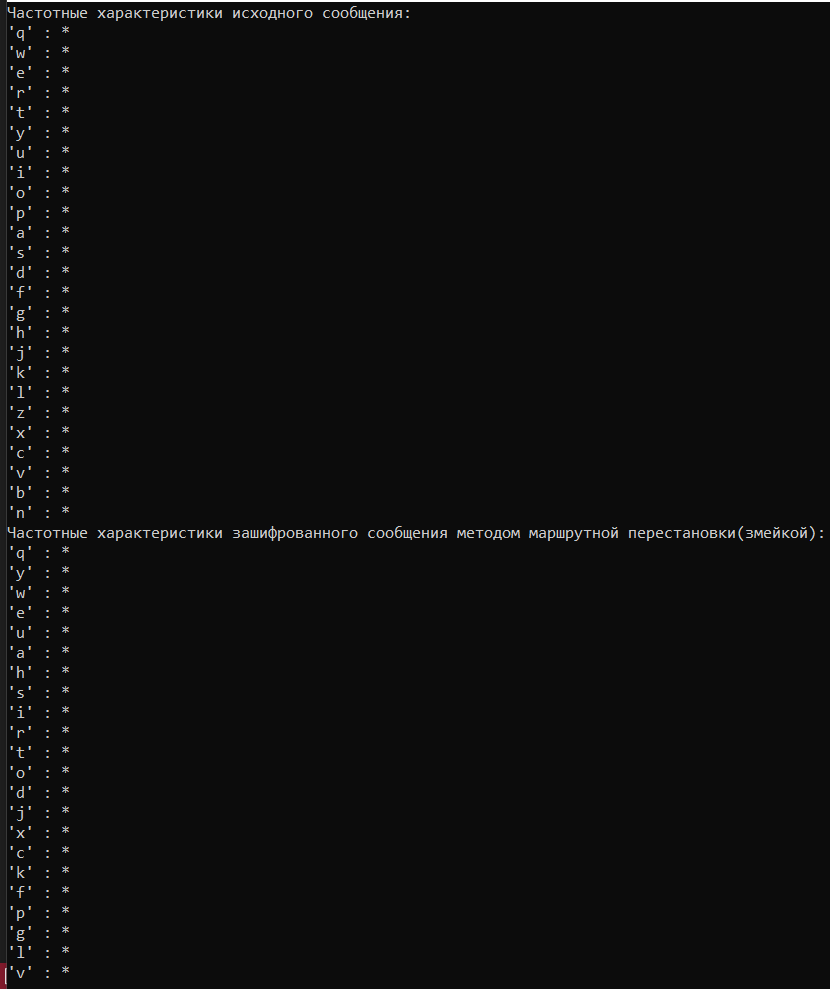


Рисунок 2.2 – Частоты появления символов в исходном и зашифрованных сообщениях

На данном скриншоте отображены гистограммы частот появления символов для исходного и зашифрованного сообщений.

Основные функции программы представлены ниже.

public static string SnakePermutationEncryptionOfGerman(string message) //шифрование перестановочным методом по маршруту(змейка)

{

string encryptedMessage = "";

char[,] matrix = new char[5,5];

const int n = 5;

string[] arr = new string[n \* n];

for (int i = 0; i < n; i++)

{

for(int j = 0; j < n;j++)

{

matrix[i, j] = message[i \* n + j];

}

}

Console.WriteLine("\nМатрица:");

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

Console.Write(matrix[i, j] + " ");

}

Console.WriteLine();

}

int k, lim;

k = 0;

for (int i = 0; i < 2 \* n; ++i)

{

lim = i >= n ? n - 1 : i;

if (i % 2 == 0)

for (int j = lim; j >= i - lim; --j)

arr[k++] += matrix[i - j, j];

else

for (int j = i - lim; j <= lim; ++j)

arr[k++] += matrix[i - j, j];

}

for (int i = 0; i < n \* n; ++i)

{

encryptedMessage += arr[i];

}

return encryptedMessage;

}

public static string MultiplePermutationEncryptionOfGerman(string message, string firstKey, string secondKey) //шифрование методом множественной перестановки

{

string fill = " ";

for (int i = message.Length; i < firstKey.Length \* secondKey.Length; i++)

{

message += fill;

}

string newText = "";

// Матрица в которой производим шифрование

char[,] matrix = new char[secondKey.Length, firstKey.Length];

// Счетчик символов в строке

int countSymbols = 0;

// Переводим строки в массивы типа char

char[] charsFirstKey = firstKey.ToCharArray();

char[] charsSecondKey = secondKey.ToCharArray();

char[] charStringUser = message.ToCharArray();

// Создаем списки в которых будут храниться символы и порядковые номера символов

listCharNumFirst =

new List<CharNum>(firstKey.Length);

listCharNumSecond =

new List<CharNum>(secondKey.Length);

// Заполняем символами из ключей

listCharNumFirst = FillListKey(charsFirstKey);

listCharNumSecond = FillListKey(charsSecondKey);

// Заполняем порядковыми номерами

listCharNumFirst = FillingSerialsNumber(listCharNumFirst);

listCharNumSecond = FillingSerialsNumber(listCharNumSecond);

// Заполнение матрицы строкой пользователя

for (int i = 0; i < listCharNumSecond.Count; i++)

{

for (int j = 0; j < listCharNumFirst.Count; j++)

{

matrix[i, j] = charStringUser[countSymbols++];

}

}

Console.WriteLine("Мтарица:");

for (int i = 0; i < listCharNumSecond.Count; i++)

{

for (int j = 0; j < listCharNumFirst.Count; j++)

{

Console.Write(matrix[i, j] + " ");

}

Console.WriteLine();

}

countSymbols = 0;

// Заполнение матрицы с учетом шифрования.

// Переставляем столбцы по порядку следования в первом ключе.

// Затем переставляем строки по порядку следования во втором ключа.

for (int i = 0; i < listCharNumSecond.Count; i++)

{

for (int j = 0; j < listCharNumFirst.Count; j++)

{

matrix[listCharNumSecond[i].NumberInWord,

listCharNumFirst[j].NumberInWord] = charStringUser[countSymbols++];

}

}

for (int i = 0; i < listCharNumFirst.Count; i++)

{

for (int j = 0; j < listCharNumSecond.Count; j++)

{

newText += matrix[j, i];

}

}

//return newText;

return newText;

}

Вывод

В ходе выполнения лабораторной работы были изучены и приобретены практические навыки разработки и использования приложений для реализации перестановочных шифров.