#### Министерство науки и высшего образования Российской Федерации

#### Федеральное государственное бюджетное образовательное учреждение

#### высшего образования

#### «Владимирский государственный университет

#### имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №4

по дисциплине

"Администратирование и безопасность программных информационных систем"

Выполнил:

ст. гр. ПРИ-117

Куппе Р.О

Принял:

Спирин И.В

Владимир, 2020 г.

## **Цель работы**

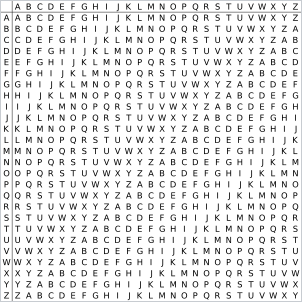
Целью работы является реализация криптографического алгоритма шифрования с использованием открытого текста в качестве ключа по таблице Виженера.

**Ход работы**

Весь код можно найти на репозитории github: https://github.com/Kuppuch/Vigenere-cipher.git

Описание алгоритма.

Алгоритм основан на движении букв по таблице. При этом каждая новая строчка, начинается на следующую букву алфавита в общем случае (когда ключ - алфавит) или на следующую букву ключа. Когда алфавит кончается, нужно начать алфавит с начала и продолжить до тех пор, пока в строке не будет столько же значений, сколько и в основном алфавите. Для того чтобы зашифровать текст, необходимо найти пересечение значений алфавита и ключа. Как в таблице Пифагора.



**Рисунок 1. Квадрат Виженера, или таблица Виженера.**

Код программы:

using System;

using System.IO;

namespace Vigenere\_algorithm {

class Program {

static char[] mass = new 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', 'ъ', 'ы', 'ь', 'э', 'ю', 'я' };

static string key = "роман";

static int m = 3;

static void Main(string[] args) {

//Вывод таблицы шифрования

/\*\*

var k = 0;

char[,] alpha = new char[33, 33];

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

for (int j = 0; j < 33; j++) {

Console.Write(mass[(i + key[k++ % key.Length]) % mass.Length]);

}

Console.WriteLine();

}

\*/

Console.WriteLine("Выберите пункт меню:");

Console.WriteLine("1 - Чтобы зашифровать");

Console.WriteLine("2 - Чтобы расшифровать");

var choice = Console.ReadLine();

if (choice == "1") {

Crypt(false);

} else {

Crypt(true);

}

Console.ReadKey();

}

public static void Crypt(bool decrypt) {

string[] text = File.ReadAllLines(!decrypt ? "input.txt" : "code.txt");

string[] output = new string[text.Length];

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

Console.Write(text[i] + " ");

text[i] = text[i].ToLower();

for (int j = 0; j < text[i].Length; j++) {

if (text[i][j] != ' ') {

output[i] += text[i][j];

}

}

//Console.Write(output[i]);

}

Console.WriteLine();

Console.WriteLine();

text = output;

output = new string[text.Length];

var k = 0;

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

output[i] = "";

for (int j = 0; j < text[i].Length; j++) {

var f1 = Find(text[i][j]); // Все переменные считаются отдельно для проверки их значений.

if (f1 == -1) {

output[i] += text[i][j];

k++;

continue;

}

var f2 = Find(key[k++ % key.Length]);

if (decrypt) {

var pos = f1 - f2;

while (pos < 0) {

pos += mass.Length;

}

output[i] += mass[(pos) % mass.Length];

} else {

output[i] += mass[(f1 + f2) % mass.Length];

}

}

Console.Write(output[i]);

}

}

public static int Find(char p) {

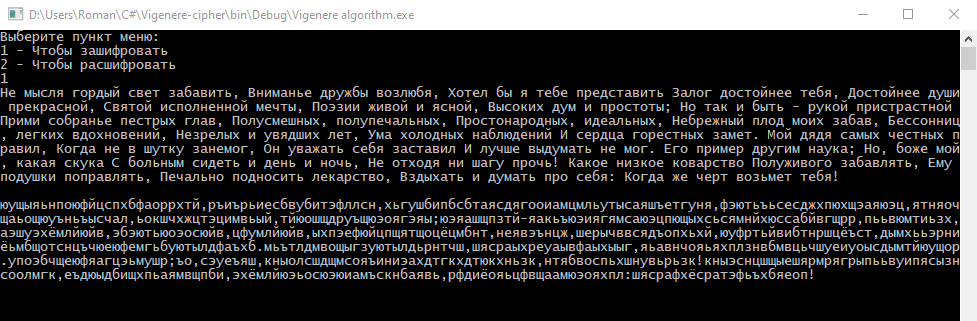
return Array.IndexOf(mass, p);

}

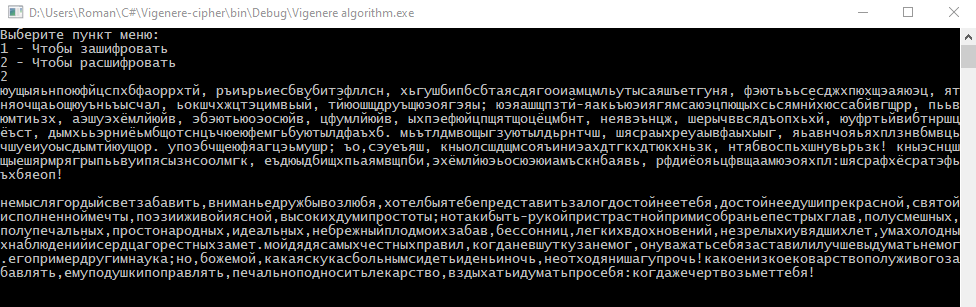
}

}

Скриншот рабочей программы.



**Рисунок 2. Демонстрация работы шифрования.**



**Рисунок 3. Демонстрация работы дешифрования.**

### **Вывод**

В ходе выполнения лабораторной работы был реализован криптографический алгоритм шифрования с использованием открытого текста в качестве ключа по таблице Виженера.