Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Пенза 2020

Лабораторная работа № 1

**«Разработка программы шифрования/дешифрирования текста с**

**использованием перестановочного шифра.»**

## Отчёт по дисциплине

**«Защита информации в сети Интернет»**

**Выполнили с-ты гр. 16ВВ1:**

Гаевский Илья

Воеков Иван

Немцев Александр

**Приняли:**

к.т.н. Дубравин А.В.  
Кармышева Н.С.

**Цель работы.**

Научиться разрабатывать программы шифрования/дешифрования текста с использованием перестановочного шифра.

**Задание.**

Разработать программу, выполняющую шифрование и расшифровывание произвольного текстового файла с использованием перестановочного шифра используя в качестве ключа последовательность, соответствующую номеру варианта (табл. 1). Выполнить проверку путем двоичного сравнения исходного файла и фала, полученного после расшифровывания.

Таблица 1 - Вариант задания.

|  |  |
| --- | --- |
| № Варианта | Последовательность перестановки |
| 3 | 3 9 10 5 7 1 2 8 6 4 |

**Листинг.**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <fstream>

#include <conio.h>

#include <vector>

#include <iostream>

#include <windows.h>

#include <locale.h>

using namespace std;

int c;

void encryption(int key[])

{

int j = 0, a = 0;

char\* v1;

ifstream ifs("Text.txt", ios::binary);

size\_t lSize = ifs.seekg(0, ios::end).tellg();

size\_t block = 1024;

ifs.seekg(0);

c = 10 - lSize % 10;

v1 = new char[lSize + c];

char \*buffer = new char[lSize + c];

ifs.read(buffer, lSize);//Считывает заданное число байт из файла в память

for (int i = 0; i < lSize + c; i++) {

v1[i] = buffer[j + key[a] - 1];

if (a > 8) {

a = 0;

j = j + 10;

}

else {

a++;

};

}

ofstream fout("Result.txt", ios\_base::trunc | ios::binary);//trunc обрезает файл, чтобы он был опустошен

for (int i = 0; i < (lSize + c); i++)

{

fout << v1[i];

}

delete[] buffer;

fout.close();

}

void decryption(int key[]) {

int j = 0, a = 0;

char\* v1;

ifstream ifs("Result.txt", ios::binary);

size\_t lSize = ifs.seekg(0, ios::end).tellg();

size\_t block = 1024;

ifs.seekg(0);

v1 = new char[lSize];

char \*buffer = new char[lSize];

ifs.read(buffer, lSize);//Считывает заданное число байт из файла в память

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

{

v1[j + key[a] - 1] = buffer[i];

if (a > 8) {

a = 0;

j = j + 10;

}

else {

a++;

};

}

ofstream fout("New.txt", ios\_base::trunc | ios::binary);//trunc обрезает файл, чтобы он был опустошен

for (int i = 0; i < lSize - c; i++)

{

fout << v1[i];

}

delete[] buffer;

fout.close();

}

void menu() {

char ch\_key;

int c, key[10] = { 3, 9, 10, 5, 7, 1, 2, 8, 6, 4 };

cout << "1 - Encrypt file.\n";

cout << "2 - Decrypt file.\n";

cout << "ESC - Exit.\n";

switch ((ch\_key = \_getch()))

{

case '1':

cout << "File Encryption...\n";

encryption(key);

cout << "Done.\n";

system("pause");

break;

case '2':

cout << "Decryption file...\n";

decryption(key);

cout << "Done.\n";

system("pause");

break;

case 27:

cout << "Exit the program....\n";

exit(0);

break;

default:

break;

}

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(LC\_ALL, "Russian");

while (true) {

system("cls");

menu();

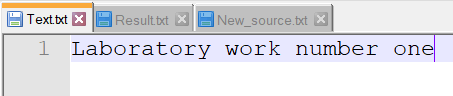
}

system("pause");

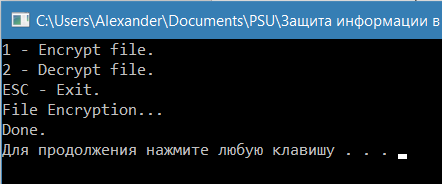
}

**Результат работы.**

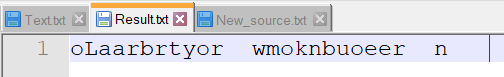
Результат работы программы показан на рисунках 1-5.



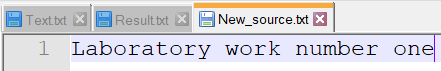
**Рисунок 1 - Исходный файл.**



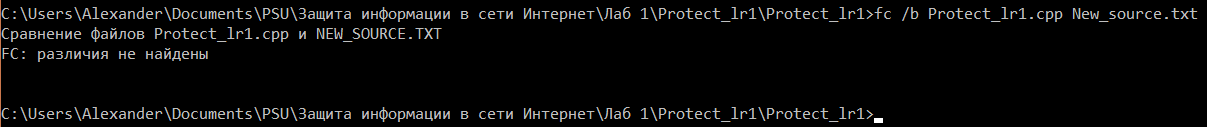
**Рисунок 2 - Меню программы.**



**Рисунок 3 - Результат после шифрования.**



**Рисунок 4 - Результат после дешифрования.**



**Рисунок 5 - Результат проверки файлов.**

Исходный текст файла совпал с результатом дешифрования.

**Вывод.**

Научились разрабатывать программы шифрования/дешифрования текста с использованием перестановочного шифра.