Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №2

по курсу «Защита информации в сети Internet»

на тему «Разработка программы шифрования/дешифрирования двоичных файлов с использованием перестановочного шифра»

Выполнили  
студенты группы 18ВВ1:

Амплеев Д.О.  
Колобов К.А.

Приняли:

к.т.н., доцент Дубравин А.В.

к.т.н., доцент Карамышева Н.С.

2020

**Цель работы:** разработать программу шифрования/дешифрирования двоичных файлов с использованием перестановочного шифра.

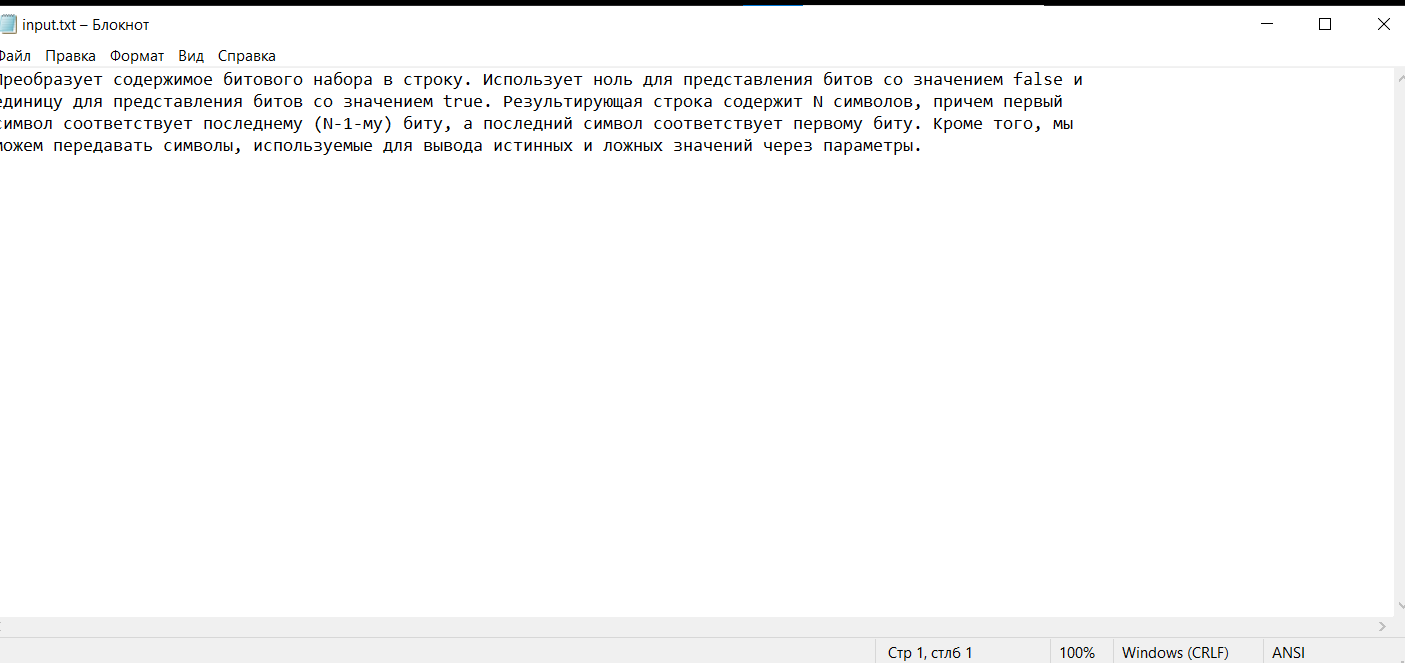
**Задания:** разработать программу, выполняющую шифрование и расшифровывание произвольного двоичного файла с использованием перестановочного шифра используя в качестве ключа последовательность, соответствующую номеру варианта. Выполнить проверку путем двоичного сравнения исходного файла и фала, полученного после расшифровывания.

|  |  |
| --- | --- |
| Номер варианта | Последовательность перестановки |
| 1 | 2 22 14 1 29 0 28 23 21 15 11 17 24 19 5 25 3 30 26 6 31 16 4 13 12 18 9 8 27 20 10 7 |

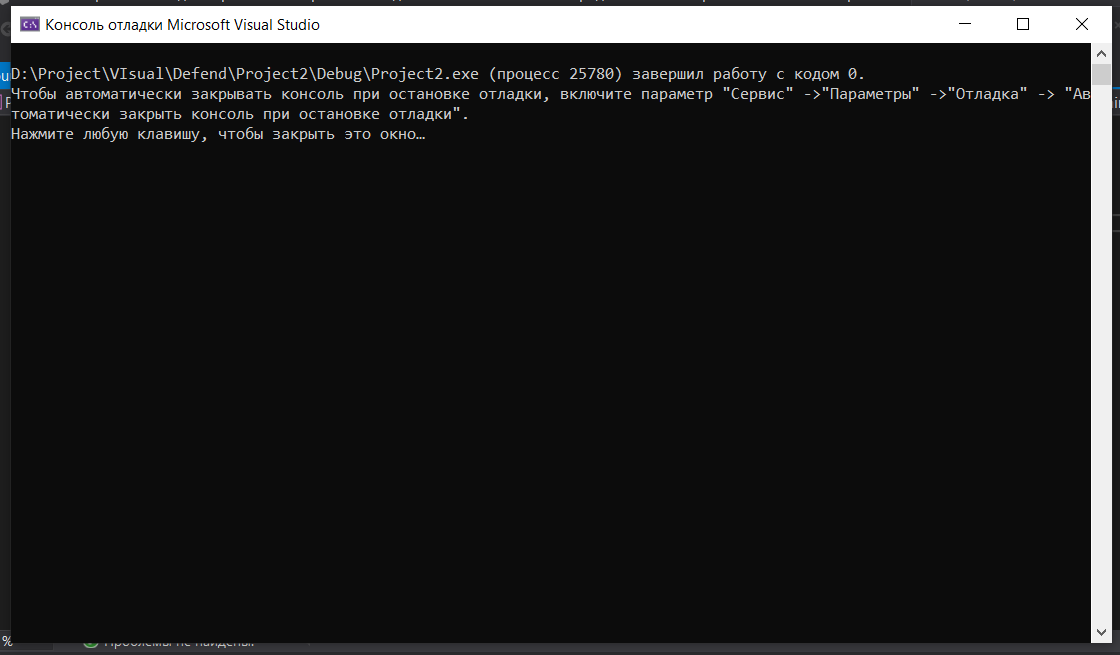
Работа программы:

Для осуществления шифрования и расшифровывания необходимо создать текстовый файл, где будет отображена последовательность перестановки. Также нужно создать файл, над которым будет проводиться шифрование.

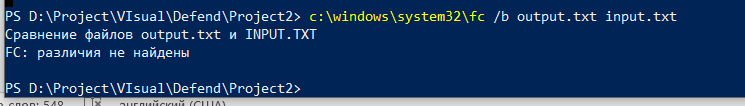
Для демонстрации работы программы создали следующий текстовый файл:



Вывод программы:







Листинг:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <string>

#include <fstream>

#include <vector>

using namespace std;

const int keySize = 32;

int keyPosition[keySize] = { 24, 10, 2, 8, 21, 26, 28, 0, 18, 12,

11, 25, 5, 22, 1, 16, 3, 9, 13, 4,

19, 17, 6, 15, 14, 20, 7, 23, 30, 31, 27, 29 };

int keySource[keySize];

void init() {

for (int i = 0; i < keySize; i++) {

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

if (keyPosition[j] == i)

keySource[i] = j;

}

}

}

vector<char> readText(const string& filename) {

ifstream fin(filename, ios::binary | ios::in);

vector<char> text((istreambuf\_iterator<char>(fin)),

istreambuf\_iterator<char>());

fin.close();

return text;

}

void writeText(const string& filename, const vector<char>& text) {

ofstream fout(filename, ios::binary | ios::out);

for (const char& c : text)

fout << c;

fout.close();

}

vector<char> encodeString(const vector<char>& source) {

vector<char> encoded = source;

for (int i = 0; i < source.size() / 4; ++i) {

unsigned int buf = 0;

for (int j = 0; j < 4; ++j) {

buf |= ((unsigned char)source[i \* 4 + j]) << (j \* 8);

}

unsigned int newBuf = 0;

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

newBuf |= ((buf >> j) & 1) << keyPosition[j];

}

for (int j = 0; j < 4; ++j) {

encoded[i \* 4 + j] = (newBuf >> (j \* 8)) & 255;

}

}

return encoded;

}

vector<char> decodeString(const vector<char>& source) {

vector<char> decoded = source;

for (int i = 0; i < source.size() / 4; ++i) {

unsigned int buf = 0;

for (int j = 0; j < 4; ++j) {

buf |= ((unsigned char)source[i \* 4 + j]) << (j \* 8);

}

unsigned int newBuf = 0;

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

newBuf |= ((buf >> j) & 1) << keySource[j];

}

for (int j = 0; j < 4; ++j) {

decoded[i \* 4 + j] = (newBuf >> (j \* 8)) & 255;

}

}

return decoded;

}

int main() {

init();

vector<char> text = readText("input.txt");

vector<char> encrypted = encodeString(text);

writeText("inputShifr.txt", encrypted);

vector<char> encryptedText = readText("inputShifr.txt");

vector<char> decodedText = decodeString(encryptedText);

writeText("output.txt", decodedText);

return 0;

}

Вывод: в ходе лабораторной работы разработали программу шифрования/дешифрирования двоичных файлов с использованием перестановочного шифра.