**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и управление»

Кафедра ИУ5. Курс «Основы информатики»

Отчет по лабораторной работе №7

# «Шифрование текстовых файлов»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-11 |  | преподаватель каф. ИУ5 |
| Анцифров Никита |  | Аксёнова М.В. |
|  |  |  |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2019 г.

**Постановка задачи**

1. Разработать программу для шифрования и дешифрирования текста, хранящегося в файле, основанную на методе Цезаря, но более «изощренную»: ключ, используемый для кодирования символа, будет зависеть от позиции символа в исходном тексте.
2. Ключи должны представлять собой целые числа, определяемые путем преобразования слов кодового блокнота по следующему алгоритму:

* ключ для k-го по порядку символа в исходном тексте вычисляется как сумма (по модулю 256) кодов символов k-го по порядку слова кодового блокнота.
* если кодовый блокнот имеет слов меньше, чем количество символов в исходном тексте, то по исчерпании слов в нём перейти к первому слову и продолжить кодирование

Получение массива ключей оформить в виде отдельной функции.

1. Исследовать статистическую зависимость символов в закодированном тексте от соответствующих им символов исходного текста (сколько каких кодов одного и того же исходного символа получено) от размера кодового блокнота для заданного исходного текста. Чем меньше полученная статистическая зависимость, тем сложнее распознать текст, не зная алгоритма кодирования и текста кодового блокнота. Статистические результаты по конкретному символу исходного текста представить в виде таблицы 16х16, по аналогии с кодировочными таблицами.

**Описание входных, выходных и вспомогательных данных**

char **answ\_char** – символ, используемый в качестве ответа пользователя

int **wordcounter** – счетчик количества слов

int\* **words** – ссылочная переменная на массив для хранения ключей

int **stats []** – массив для хранения статистики

int **i** и **j** – счетчики цикла

ifstream **fin** – переменная для чтения файла

char unsigned **tempchar** – переменная для хранения символа

char **tch** - переменная для хранения символа

char **c** – вспомогательная переменная

int **sum** – счетчик

ofstream **fout** – переменная для записи в файл

**Описание прототипов функций**

int **how\_many\_words();** - функция возвращает значение количества слов в key.txt

void **key\_arr\_creator(int\*, int\*)** – функция принимает адрес на массив и адрес на значение количества слов для заполнения этого массива ключами

void **encrypter(int\*, int\*, int\*)** – функция принимает адрес на массив, адрес на значение количества слов и адрес на массив для статистики и производит шифрование и сбор статистики для символа

void **decrypter(int\*, int\*, int\*)** – функция принимает адрес на массив, адрес на значение количества слов и адрес на массив для статистики и производит процесс расшифровки файла

**Алгоритм**

Функция **main**



Функция **how\_many\_words**



Функция **key\_arr\_creator**



Функция **encrypter**



Функция **decrypter**



**Текст программы**

*L7.cpp*

#include <iostream>

#include <fstream>

#include "func\_proto.h"

using namespace std;

void main()

{

system("chcp 1251 > nul");

cout << "Убедитесь, что в папке с проектом находятся файл с ключом \"key.txt\" и файл с текстом \"text.txt\"." << endl;

cout << "Зашифрованный файл \"encrypted.txt\" и расшифрованный файл \"decrypted.txt\" будут также находиться в папке с проектом" << endl << endl;

char answ\_char = 'y';

do

{

int wordcounter=0;

wordcounter=how\_many\_words();

int\* words=new int[wordcounter];

key\_arr\_creator(words, &wordcounter);

int stats[256];

for (int i=0; i<256; i++)

{

stats[i]=0;

}

encrypter(words, &wordcounter, stats);

decrypter(words, &wordcounter);

cout << endl;

cout << "Статистика: " << endl;

for (int i=0; i<16; i++)

{

for (int j=0; j<16; j++)

{cout << stats[16\*i+j] << " ";}

cout << endl;

}

delete[]words;

cout << endl;

cout << "Повторить? (y/n) ";

cin >> answ\_char;

}

while (answ\_char == 'y');

system("pause");

}

*func.cpp*

#include <iostream>

#include <fstream>

using namespace std;

int how\_many\_words()

{

ifstream fin;

char unsigned tempchar;

char tch;

int sum=0;

fin.open("key.txt", ios\_base::binary);

if (!fin)

{

cout << "Ошибка открытия файла ключей. Проверьте расположение файла в папке с проектом" << endl;

system("pause");

}

else

{

cout << "Файл с ключами успешно открыт." << endl;

while (!fin.eof())

{

tch=fin.get();

tempchar=tch;

if ((tempchar==' ') || (tempchar=='\n'))

{

sum++;

}

}

fin.close();

}

return (sum+1);

}

void key\_arr\_creator(int\* words, int\* wordcounter)

{

fstream fin;

int i=0;

char unsigned tempchar;

for (i=0; i<(\*wordcounter); i++)

{

words[i]=0;

}

i=0;

fin.open("key.txt");

fin.clear();

fin.seekg(0);

while (!fin.eof())

{

tempchar=fin.get();

if ((tempchar==' ') || (tempchar=='\n'))

{

i++;

}

else

{

words[i]+=(tempchar);

}

}

fin.close();

}

void encrypter(int\* words, int\* wordcounter, int\* stats)

{

ifstream fin;

ofstream fout;

int i=0;

char unsigned tempchar, c;

char tch;

fin.open("text.txt", ios\_base::binary);

if (!fin)

{

cout << "Ошибка открытия файла текста." << endl;

system("pause");

}

else

{

cout << "Файл с текстом успешно открыт." << endl;

fout.open("encrypted.txt", ios\_base::binary);

cout << "Запуск процесса шифрования файла." << endl;

char unsigned charforstat;

cout << "Введите символ для сбора статистики: ";

cin >> charforstat;

tch=fin.get();

while (!fin.eof())

{

tempchar=tch;

c=((words)[i]+tempchar)%256;

if (tempchar==charforstat)

{

stats[c]++;

}

fout << c;

i++;

if (i==(\*wordcounter))

{

i=0;

}

tch=fin.get();

}

}

fin.close();

fout.close();

cout << "Процесс шифрования завершен." << endl;

}

void decrypter(int\* words, int\* wordcounter)

{

ifstream fin;

ofstream fout;

char unsigned tempchar;

char tch;

int i=0;

fin.open("encrypted.txt", ios\_base::binary);

fout.open("decrypted.txt", ios\_base::binary);

tch=fin.get();

while (!fin.eof())

{

tempchar=tch;

tempchar=(tempchar-(words)[i])%256;

fout << tempchar;

i++;

if (i=(\*wordcounter))

{

i=0;

}

tch=fin.get();

}

fin.close();

fout.close();

cout << "Расшифрованный файл создан." << endl;

}

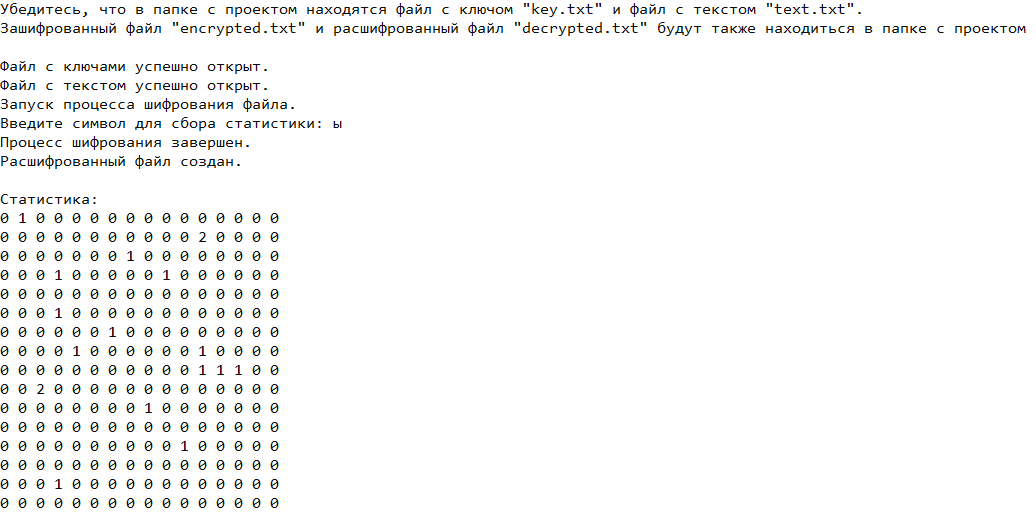
*func\_proto.cpp*

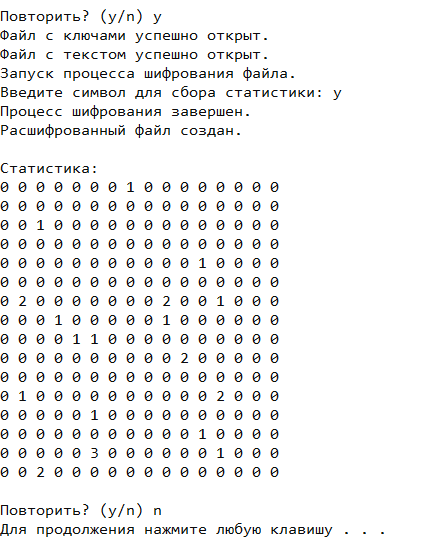
int how\_many\_words();

void key\_arr\_creator(int\*, int\*);

void encrypter(int\*, int\*, int\*);

void decrypter(int\*, int\*);

**Анализ результатов**



Файлы

*text.txt и decrypted.txt*

А.С. Пушкин

"У лукоморья дуб зеленый..."

У лукоморья дуб зелёный;

Златая цепь на дубе том:

И днём и ночью кот учёный

Всё ходит по цепи кругом;

Идёт направо — песнь заводит,

Налево — сказку говорит.

Там чудеса: там леший бродит,

Русалка на ветвях сидит;

Там на неведомых дорожках

Следы невиданных зверей;

Избушка там на курьих ножках

Стоит без окон, без дверей;

Там лес и дол видений полны;

Там о заре прихлынут волны

На брег песчаный и пустой,

И тридцать витязей прекрасных

Чредой из вод выходят ясных,

И с ними дядька их морской;

Там королевич мимоходом

Пленяет грозного царя;

Там в облаках перед народом

Через леса, через моря

Колдун несёт богатыря;

В темнице там царевна тужит,

А бурый волк ей верно служит;

Там ступа с Бабою Ягой

Идёт, бредёт сама собой,

Там царь Кащей над златом чахнет;

Там русский дух… там Русью пахнет!

И там я был, и мёд я пил;

У моря видел дуб зелёный;

Под ним сидел, и кот учёный

Свои мне сказки говорил.

*key.txt*

Александр Сергеевич Пушкин (26 мая [6 июня] 1799, Москва — 29 января [10 февраля] 1837, Санкт-Петербург) — русский поэт, драматург и прозаик, заложивший основы русского реалистического направления, критик и теоретик литературы, историк, публицист; один из самых авторитетных литературных деятелей первой трети XIX века.