Министерство образования и науки Российской Федерации

Ульяновский Технический университет

Кафедра: вычислительная техника

Дисциплина: Информатика

Лабораторная работа №1.

Шифрование - дешифрование

Вариант 12

Выполнил:

Студент: ИВТАП Бд-11

Кондратьев Павел Сергеевич

Проверила:

Валюх Вероника Валерьевна

Ульяновск, 2016

Содержание

|  |  |
| --- | --- |
| 1.Введение…………………………………………………………………… | 2 |
| 2. Постановка задач ………………………………………………………… | 3 |
| 3. Выполнение работы …………………………………………………….. | 3 |
| 4. Список литературы ………………………………………………………. | 5 |
| 5. Приложение № 1……………………………………………………….. | 6 |
| 6. Приложение № 1…………………………………………………………... | 7 |

**Введение**

**Шифр Цезаря**, также известный как шифр сдвига, код Цезаря или сдвиг Цезаря - один из самых простых и наиболее широко известных методов шифрования. Шифр Цезаря - это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите. Например, в шифре со сдвигом вправо на 3, А была бы заменена на Г, Б станет Д, и так далее. Шифр назван в честь римского императора Гая Юлия Цезаря, использовавшего его для секретной переписки со своими генералами.

**Шаг шифрования**, выполняемый шифром Цезаря, часто включается как часть более сложных схем, таких как шифр **Виженера**, и всё ещё имеет современное приложение в системе ROT13. Как и все моноалфавитные шифры, шифр Цезаря легко взламывается и не имеет почти никакого применения на практике.

**Взлом шифра:**

Шифр Цезаря может быть легко взломан даже в случае, когда взломщик знает только зашифрованный текст. Можно рассмотреть две ситуации:

1)Взломщик знает (или предполагает), что использовался простой шифр подстановки, но не знает, что это — схема Цезаря.

2)Взломщик знает, что использовался шифр Цезаря, но не знает значение сдвига.

В первом случае шифр может быть взломан, используя те же самые методы что и для простого шифра подстановки, такие как частотный анализ и т. д.

Во втором случае взлом шифра является даже более простым. Существует не так много вариантов значений сдвига (26 для английского языка), все они могут быть проверены методом грубой силы[15].

Другой способ применения этого метода - это написать алфавит под каждой буквой зашифрованного текста, начиная с этой буквы. Метод может быть ускорен, если использовать заранее подготовленные полоски с алфавитом. Для этого нужно сложить полоски так, чтобы в одной строке образовался зашифрованый текст, тогда в некоторой другой строке мы увидим открытый текст.

Другой подход к применению метода грубой силы для взлома - проверить частоты встречаемости букв. Изобразив диаграммой частоты встречания букв в зашифрованном тексте, и зная ожидаемое распределение букв для обычного текста на рассматриваемом языке, можно легко определить сдвиг, взглянув на смещение некоторых характерных черт на диаграмме.

**Техническое Задание:**

Требуется написать реализацию кода шифра Цезаря в visual studio(используя работу с файлами). Программа должна переписывать каждый символ в файле на постоянное значение key, которое меняет символы на key вперед(key = 3; а меняется на г). Изучить шифры и коды, а также получить практические навыки шифрования и дешифрования текстовых данных.

**Входные данные и Выходные данные**

Представляет собой файл:

1) cipher code, в котором лежит текст.

2) cipher decode, в котором лежит закодированный текст

3) result, в котором лежит текст после декодирования.

**Примеры**

|  |
| --- |
| **входные данные** |
| Write Code |
| **выходные данные** |
| Zulwh Frgh |
| **входные данные** |
| Как правильно дебажить код |
| **выходные данные** |
| Нгн тугелоярс зидгйлхя нсз |
| **входные данные** |
| Шаманский бубен |
| **выходные данные** |
| Ыгпгрфнлм дцдир |

**Выполнение работы:**

**І**

**Шифрование**

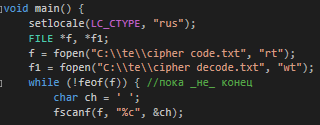
1) Первым делом создадим 3 const:

1)Ключ

2) Длинна русского алфавита

3) Длинна английского алфавита

2) Для программы понадобятся 2 файла: 1 – для считывания текста, 2 же для вывода кодирования текста по ключу. Создаем их. После чего считываем посимвольно пока не закончиться символы в файле, после чего нам остается всего лишь проверить эти символы и с помощью ключа зашифровать их(переписывая их во второй файл).

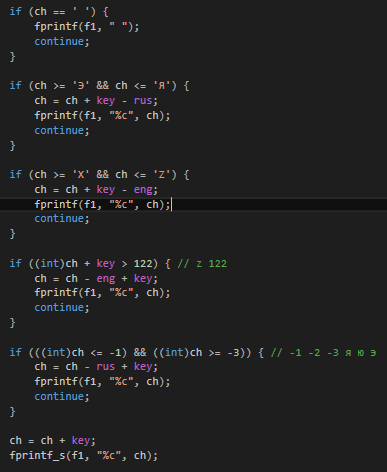


3) В цикле while мы проверяем условия:

Во-первых, если встретиться пробел в файле, то его же переписываем во 2 файл

Во-вторых проверка на заглавные буквы русского и английского алфавита (последние)

В третьих, проверка на вылет за конец алфавита  
В четвертых, если из выше перечисленного ничего не выполнилось, то к символу прибавить ключ



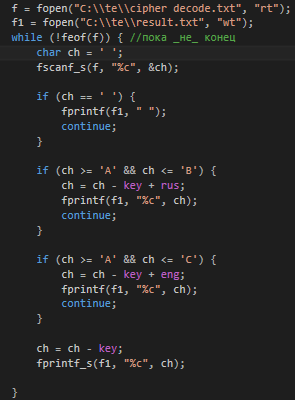
После того как мы выполнили полную зашифровку всего текста, мы закрываем файлы.

**ІІ**

**Дешифрование**

Для дешифрования нам потребуется тоже самое что и для шифрования только мы будем идти в обратном направлении.

1. Создадим 3 файл под названием result, в который записывается результат декодирования.
2. Точно также мы считываем символы пока не конец файла (из зашифрованного файла) и проверяем всего лишь на заглавные русского и английского алфавита + последнее условие: вычитаем ключ от символа(чтобы вернуть первоначальный символ до кодирования)
3. Полученные символы записываем в файл и закрываем после того как закончили декодирование.



**Список литературы:**

1) Лабораторная работа Лапшов Ю. А. «Основы программирования»

2) лаб.Информатика ВАЛЮХ В. В . от 04.10.2016

3) Лекция Власенко О. Ф. «Программирование»

**Приложение № 1 (Исходный код): Шифрование**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <iostream> using namespace std;

#define key 3 #define rus 32 #define eng 26

void main() {

setlocale(LC\_CTYPE, "rus");

FILE \*f, \*f1;

f = fopen("C:\\te\\cipher code.txt", "rt");

f1 = fopen("C:\\te\\cipher decode.txt", "wt");

while (!feof(f)) { //пока \_не\_ конец

char ch = ' ';

fscanf(f, "%c", &ch);

if (ch == ' ') {

fprintf(f1, " ");

continue;

}

if (ch >= 'Э' && ch <= 'Я') {

ch = ch + key - rus;

fprintf(f1, "%c", ch);

continue;

}

if (ch >= 'X' && ch <= 'Z') {

ch = ch + key - eng;

fprintf(f1, "%c", ch);

continue;

}

if ((int)ch + key > 122) { // z 122

ch = ch - eng + key;

fprintf(f1, "%c", ch);

continue;

}

if (((int)ch <= -1) && ((int)ch >= -3)) { // -1 -2 -3 я ю э

ch = ch - rus + key;

fprintf(f1, "%c", ch);

continue;

}

ch = ch + key;

fprintf\_s(f1, "%c", ch);

}

fclose(f);fclose(f1);

}

**Приложение № 2 (Исходный код): Дешифрование**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <iostream>

using namespace std;

#define key 3

#define rus 32

#define eng 26

void main() {

setlocale(LC\_CTYPE, "rus");

FILE \*f, \*f1;

f = fopen("C:\\te\\cipher decode.txt", "rt");

f1 = fopen("C:\\te\\result.txt", "wt");

while (!feof(f)) { //пока \_не\_ конец

char ch = ' ';

fscanf\_s(f, "%c", &ch);

if (ch == ' ') {

fprintf(f1, " ");

continue;

}

if (ch >= 'А' && ch <= 'В') {

ch = ch - key + rus;

fprintf(f1, "%c", ch);

continue;

}

if (ch >= 'A' && ch <= 'C') {

ch = ch - key + eng;

fprintf(f1, "%c", ch);

continue;

}

ch = ch - key;

fprintf\_s(f1, "%c", ch);

}

fclose(f);

fclose(f1);

}