ФГБОУ ВПО «Чувашский Государственный Университет им. И.Н.Ульянова»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

**Лабораторная работа №2  
“Шифрование данных методом перестановки”**

Выполнил: студент группы ИВТ-21-12  
 Чесноков Александр  
 Проверил: Павлов П.В.

Чебоксары, 2015

**Цель работы**

Целью работы является знакомство с классическим криптографическим алгоритмом - алгоритмом шифрования данных при помощи перестановки.

**Индивидуальное задание**

| **№ п.п.** | ***V*** | ***m*** |
| --- | --- | --- |
| **1** | **{ а,б,..,я,a,b,..,z}** | **15** |

**Функция шифрования**

Для шифрования использую ключ:

int key[m] = { 6, 9, 1, 5, 10, 3, 15, 4, 12, 2, 13, 7, 11, 8, 14 };

Сначала вводим сообщение, длина сообщения проверятся на кратность 15, если нет, тонеобходимо последний блок дополнить с правой стороны нужным количеством символов, будем дополнять его символом “ъ”.

Далее вызываю функцию шифрования сообщения *Encryption()*, которая в качестве аргументов принимает исходное. Функция, расставляет символы исходного сообщения в порядке, установленном ключом. И возвращает получившуюся строку, в качестве результата шифрования.

string Encryption(string str)

{

string temp;

int step = 0;

for (int i = 0, j = 0; i < str.length(); i++)

{

if (i % m == 0 && i != 0)

step += m;

int index = key[i%m] - 1;

temp += str[index + step];

}

return temp;

}

**Текст программы, реализующей индивидуальное задание**

#include <iostream>

#include <string>

#include "windows.h"

using namespace std;

const int m = 15;

int key[m] = { 6, 9, 1, 5, 10, 3, 15, 4, 12, 2, 13, 7, 11, 8, 14 };

char buf[15];

string Encryption(string str);

string Decryption(string str);

void main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

string str;

string mess;

string encryptMess;

string decryptMess;

cout << "Введите сообщение, используя данный алфавит (\"a..z,а..я\"):\n";

cin >> str;

mess = str;

if (str.length() % m != 0)

str.append("ъъъъъъъъъъъъъъъ", m - (str.length() % m));

cout << "\nИсходное сообщение:\n" << mess << "\n\nСкорректированное сообщение: " << str;

encryptMess = Encryption(str);

str = encryptMess;

decryptMess = Decryption(str);

cout << "\nШифрограмма: " << encryptMess;

cout << "\nДешифрованное сообщение: " << decryptMess;

cout << "\n\n";

system("pause");

}

string Encryption(string str)

{

string temp;

int step = 0;

for (int i = 0, j = 0; i < str.length(); i++)

{

if (i % m == 0 && i != 0)

step += m;

int index = key[i%m] - 1;

temp += str[index + step];

}

return temp;

}

string Decryption(string str)

{

string temp;

int step = 0;

for (int i = 0, j = 0; i <= str.length(); i++)

{

if (i % m == 0 && i != 0)

{

temp += buf;

j = 0;

step += m;

}

int index = key[j] - 1;

buf[index] = str[i];

j++;

}

return temp;

}

**Пример открытого текста и соответствующей ему шифрограммы**

X = “тестированиепрограммы”.

Разбиваем текст на блоки m = 15:

“тестированиепро”, ”граммыъъъъъъъъъ”, необходимо последний блок дополнить с правой стороны нужным количеством символов, будем дополнять его символом “ъ”.

Словарь шифрования будет такой:

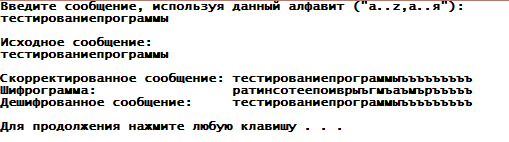
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CN(i)* | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| *i* | 6 | 9 | 1 | 5 | 10 | 3 | 15 | 4 | 12 | 2 | 13 | 7 | 11 | 8 | 14 |

В каждом пятнадцатисимвольном блоке шифртекста, на первой позиции будет стоять символ, занимающий 6-ю позицию в блоке открытого текста, на второй позиции - символ, занимающий 9-ю позицию в блоке открытого текста, и т.д.

Таким образом, наш открытый текст даст следующие блоки шифртекста:

“ратинсотеепоивр”, “ыъгмъаъмъръъъъъ”.

Пример работы программы:



**Вывод**

Я ознакомился с классическим криптографическим алгоритмом - алгоритмом шифрования данных при помощи перестановки и реализовал его в среде программирования VS2013 на языке C++.