Математические основы защиты информации и информационной безопасности.

Лабораторная работа №2.

Подмогильный Иван Александрович.

Содержание

List of Figures

List of Tables

1 Цель работы

Освоить на практике шифрование гаммированием

2 Задание

1. Реализовать шифрование функцию шифрования гаммированием конечной гаммой

3 Выполнение лабораторной работы

Написал код для зашивровки кодов конечной гаммой. Тело функции:

```
//
// Created by pi on 17.09.2022.
//

#include "../include/CipherFiniteGammaHelper.h"

std::string CipherFiniteGammaHelper::engAlphabetLower = "abcdefghijklmnopqrstuvwxyz !,";
std::string CipherFiniteGammaHelper::engAlphabetUpper = "ABCDEFGHIJKLMNOPQRSTUVWXYZ !,";
std::vector<std::string> CipherFiniteGammaHelper::rusAlphabetLower = {"а", "б", "в", "г", "д", "е", "ё", "ж", "з", "и", "й", "к",
                                                                      "л", "м", "н", "о", "п", "р", "с", "т", "у", "ф", "х", "ц",
                                                                      "ч", "ш", "щ", "ъ", "ы", "ь", "э", "ю", "я"};
std::vector<std::string> CipherFiniteGammaHelper::rusAlphabetUpper = {"А","Б", "В", "Г", "Д", "Е", "Ё", "Ж", "З", "И", "Й", "К",
                                                                      "Л", "М", "Н", "О", "П", "Р", "С", "Т", "У", "Ф", "Х", "Ц",
                                                                      "Ч", "Ш", "Щ", "Ъ", "Ы", "Ь", "Э", "Ю", "Я"};


template <typename T>
void CipherFiniteGammaHelper::cipher(const T &msg, const T &pwd, T &encrypted){
    int modulo = 26;
    char msgIndexes[msg.size()], pwdIndexes[pwd.size()];
    for (int i = 0; i < msg.size(); i++){
        msgIndexes[i] = engAlphabetLower.find(msg[i]);
    }
    for (int i = 0; i < pwd.size(); i++){
        pwdIndexes[i] = engAlphabetLower.find(pwd[i]);
    }

    for (int i = 0; i < msg.size(); i++){
        int index = (msgIndexes[i] + pwdIndexes[i % pwd.size()]) % modulo;
        encrypted[i] = engAlphabetLower[index];
    }
}

template <typename T>
void CipherFiniteGammaHelper::decipher(const T &enc, const T &pwd, T &decr) {
    cipher(enc, pwd, decr);
}


// so, this is as an easy fix. but if so, you need to list all the possible usages. Let's say that it's ok. By now
// the program was tested with vectors and string. You can always switch back to the template definition inside .h file
// then you don't need these lines.
template void CipherFiniteGammaHelper::cipher(const std::string &msg, const std::string &pwd, std::string &encrypted);
template void CipherFiniteGammaHelper::decipher(const std::string &enc, const std::string &pwd, std::string &decr);
template void CipherFiniteGammaHelper::cipher(const std::vector<char> &msg, const std::vector<char> &pwd, std::vector<char> &encrypted);
```

Figure 3.1: зашивровки кодов конечной гаммой

Заголовочный файл:

Figure 3.2: заголовочный файл

Написал CMakeLists.txt файл.

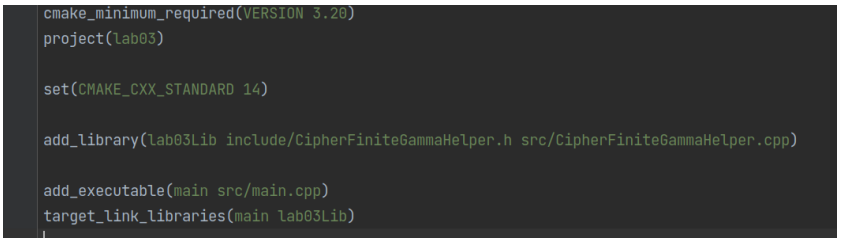

Figure 3.3: CMakeLists.txt файл

Написал main.cpp файл, в котором есть тесты реализованных функций.

```cpp
#include <iostream>
#include "../include/CipherFiniteGammaHelper.h"

int main() {
    std::string test1 = "hello, world!";
    std::string enc1( test1.size(),  c: ' ');
    std::string dec1( test1.size(),  c: ' ');
    std::string pwd1( s: "parol");

    std::vector<char> test2 = {'h', 'e', 'l', 'l', 'o'};
    std::vector<char> enc2(test2.size(),  value: ' ');
    std::vector<char> dec2(test2.size(),  value: ' ');
    std::vector<char> pwd2 = {'p', 'a', 'r', 'o', 'l'};

    CipherFiniteGammaHelper::cipher(test1, pwd1,  &: enc1);
    std::cout << enc1 << std::endl;

    CipherFiniteGammaHelper::cipher(test2, pwd2,  &: enc2);
    for (int i = 0; i < enc2.size(); i++){
        std::cout << enc2[i];
    }
    std::cout << std::endl;

//    CipherFiniteGammaHelper::decipher(enc1, pwd1, dec1);
//    std::cout << dec1 << std::endl;


    return 0;
}
```

Figure 3.4: main.cpp файл

Результаты тестов.

```
/home/pi/education/pfur_masters/matOsnovyInfBez/labs/lab03/cmake-build-debug/main
weczzranccads
weczz

Process finished with exit code 0
```

Figure 3.5: Результаты тестов

4 Выводы

Освоил на практике применения метода шифрования гаммированием с конечной гаммой