Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Информационные сети. Основы безопасности

ОТЧЕТ

к лабораторной работе №1

на тему

**ШИФР ЦЕЗАРЯ. ШИФР ВИЖЕНЕРА**

Студент А. В. Скворцов

Преподаватель Е. А. Лещенко

Минск 2024

СОДЕРЖАНИЕ

[Введение 3](#_Toc157471308)

[1 Результат выполнения 4](#_Toc157471309)

[Заключение 5](#_Toc157471310)

[Приложение А (обязательное) Листинг кода 6](#_Toc157471311)

[Приложение Б (обязательное) Блок-схема алгоритма 8](#_Toc157471312)

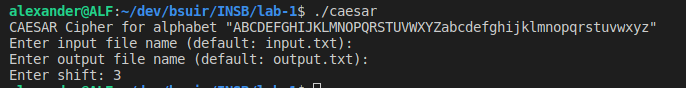
**ВВЕДЕНИЕ**

В ходе лабораторной работы нужно разработать программные средства для шифрования и дешифрования текстовых файлов с использованием двух классических методов шифрования: шифра Цезаря и шифра Виженера.

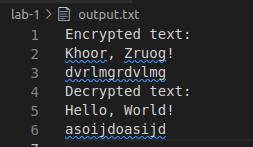
Основная цель работы – изучение принципов работы указанных алгоритмов и их реализация на языке программирования *С++*. В ходе работы будет осуществлено знакомство с методами шифрования, разработка алгоритмов шифрования и дешифрования, а также создание программного продукта, позволяющего осуществлять шифрование и дешифрование текстовых файлов.

1. **РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ**

В результате работы было создано два приложения для шифрования и дешифрования текстовых файлов: с использованием шифра Цезаря   
(рисунок 1) и шифра Виженера (рисунок 2).



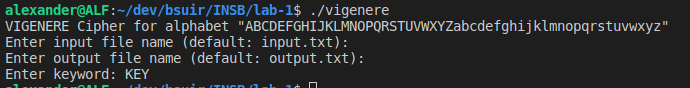
а



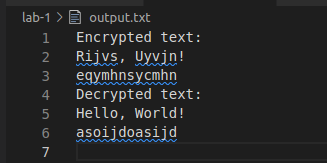
б

а – взаимодействие с программой; б – выходные данные

Рисунок 1 – Шифр Цезаря



а



б

а – взаимодействие с программой; б – выходные данные

Рисунок 2 – Шифр Виженера

# ЗАКЛЮЧЕНИЕ

В результате выполнения данной лабораторной работы были разработаны программные средства для шифрования и дешифрования текстовых файлов с применением шифра Цезаря и шифра Виженера на языке программирования *С++*.

Был проведен анализ методов шифрования и алгоритмов работы с текстовыми файлами, разработана блок-схема алгоритма, написаны и отлажены программы для шифрования и дешифрования файлов.

При выполнении работы были использованы теоретические сведения о принципах работы шифра Цезаря и шифра Виженера и получены практические навыки работы с данными алгоритмами.

# ПРИЛОЖЕНИЕ А (обязательное) Листинг кода

Листинг 1 – Файл *caesar.cpp*

#include <iostream>

#include <fstream>

#include <string>

#include <locale.h>

std::string encryptCaesar(std::string text, int shift) {

std::string result = "";

int length = text.length();

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

if (isalpha(text[i])) {

char ch = isupper(text[i]) ? 'A' : 'a';

result += (text[i] - ch + shift) % 26 + ch;

} else {

result += text[i];

}

}

return result;

}

std::string decryptCaesar(std::string text, int shift) {

return encryptCaesar(text, 26 - shift);

}

int main() {

std::string inputFileName, outputFileName, inputText;

std::cout << "CAESAR Cipher for alphabet \"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz\"" << std::endl;

std::cout << "Enter input file name: ";

std::getline(std::cin, inputFileName);

std::cout << "Enter output file name: ";

std::getline(std::cin, outputFileName);

std::ifstream inputFile(inputFileName);

std::ofstream outputFile(outputFileName);

if (inputFile.is\_open()) {

std::getline(inputFile, inputText);

inputFile.close();

int shift = 3;

std::string encryptedText = encryptCaesar(inputText, shift);

std::string decryptedText = decryptCaesar(encryptedText, shift);

outputFile << "Encrypted text: " << encryptedText << std::endl;

outputFile << "Decrypted text: " << decryptedText << std::endl;

outputFile.close();

} else {

std::cout << "Unable to open the input file." << std::endl;

}

return 0;

}

Листинг 2 – Файл *vigenere.cpp*

#include <iostream>

#include <fstream>

#include <tuple>

std::tuple<char, int, int> getBaseParameters(std::string keyword, char textChar, int keywordIndex) {

char base = isupper(textChar) ? 'A' : 'a';

char offset = isupper(keyword[keywordIndex]) ? 'A' : 'a';

int messageIndex = textChar - base;

int keyCharIndex = keyword[keywordIndex] - offset;

return std::tuple<char, int, int>(base, messageIndex, keyCharIndex);

}

std::string encryptVigenere(std::string text, std::string keyword) {

std::string result = "";

int keywordIndex = 0;

for (auto& ch: text) {

if (isalpha(ch)) {

auto baseParameters = getBaseParameters(keyword, ch, keywordIndex);

result += static\_cast<char>((std::get<1>(baseParameters) + std::get<2>(baseParameters)) % 26 + std::get<0>(baseParameters));

keywordIndex = (keywordIndex + 1) % keyword.length();

} else {

result += ch;

}

}

return result;

}

std::string decryptVigenere(std::string text, std::string keyword) {

std::string result = "";

int keywordIndex = 0;

for (auto& ch: text) {

if (isalpha(ch)) {

auto baseParameters = getBaseParameters(keyword, ch, keywordIndex);

result += static\_cast<char>((std::get<1>(baseParameters) - std::get<2>(baseParameters) +26) % 26 + std::get<0>(baseParameters));

keywordIndex = (keywordIndex + 1) % keyword.length();

} else {

result += ch;

}

}

return result;

}

int main() {

std::string inputFileName, outputFileName, inputText, keyword;

std::cout << "VIGENERE Cipher for alphabet \"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz\"" << std::endl;

std::cout << "Enter input file name: ";

std::getline(std::cin, inputFileName);

std::cout << "Enter output file name: ";

std::getline(std::cin, outputFileName);

std::cout << "Enter keyword: ";

std::getline(std::cin, keyword);

std::ifstream inputFile(inputFileName);

std::ofstream outputFile(outputFileName);

if (inputFile.is\_open()) {

std::getline(inputFile, inputText);

inputFile.close();

std::string encryptedText = encryptVigenere(inputText, keyword);

std::string decryptedText = decryptVigenere(encryptedText, keyword);

outputFile << "Encrypted text: " << encryptedText << std::endl;

outputFile << "Decrypted text: " << decryptedText << std::endl;

outputFile.close();

} else {

std::cout << "Unable to open the input file." << std::endl;

}

return 0;

}

# ПРИЛОЖЕНИЕ Б (обязательное) Блок-схема алгоритма

