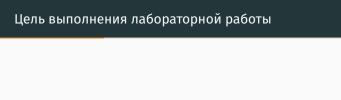
Лабораторная работа №1

Селезнев Василий Александрович - студент группы НПМмд-02-22 16.09.2022 Шифры простой замены —

Прагматика выполнения

Умение пользоваться шифрами Цезаря и Атбаша



Освоить на практике использование шифров Цезаря и Атбаша



Написать функции, которые реализуют шифрование шифрами Цезаря и Атбаша

Результаты выполнения лабораторной работы. Часть 1

```
#include "../include/CipherHelper.h"
void CipherCaesar::cipher(const std::string& message, int Key, std::string& encrypted){
        throw std::invalid_argument( s: "encrypted is not empty!" );
    if (!message.emptv()){
```

Figure 1: Caesar cipher

Написал код для дешивровки кодов шифром Цезаря

Figure 2: Caesar decipher

Написал код для зашивровки кодов шифром Атбаша

Figure 3: Atbash cipher

Написал код для дешивровки кодов шифром Атбаша

Figure 4: Atbash decipher

Написал заголовочный файл для класса реализации CipherHelper

```
#ifndef LAB01_CIPHERHELPER_H
 class CipherHelper{
 lass CipherCaesar : CipherHelper {
    static void cipher (const std::string& message, int Key, std::string& encrypted);
    static void decipher (const std::string& message, int Key, std::string& decrypted);
 lass CipherAtbash : CipherHelper {
    static void cipher (const std::string& message, std::string& encrypted);
```

Figure 5: Header file

Написал CMakeLists.txt файл, который создаёт библиотеку из класса CipherHelper и бинарник main

```
case_attainum_required(rESISM 3.50)

project(labSI)

set(CAME_CAN_STADAMS 14)

add_tbrany(labd1 src/cajberfelper.cpp)

add_tbrany(labd1 src/cajberfelper.cpp)

target_link_tbranies@min labdi)
```

Figure 6: CmakeLists.txt file

Написал main.cpp файл, в котором есть тесты реализованных функций. Часть шифра Цезаря:

```
int main(){
   std::string msq1 = "MY NAME IS VASYA";
   std::string msq2 = "I LOVE STUDY";
   std::string msq3 = "RUDN IS THE BEST UNIVERSITY";
   std::string enc1 = "", enc2 = "", enc3 = "";
   std::string dec1 = "", dec2 = "", dec3 = "";
   CipherCaesar::cipher( message: msq1, Key: 3, &: enc1);
   CipherCaesar::cipher( message: msq2, Key: 3, &: enc2);
   CipherCaesar::cipher( message: msg3, Key: 3, &: enc3);
   CipherCaesar::decipher( message: enc1, Key: 3, &: dec1);
   CipherCaesar::decipher( message: enc2, Key: 3, &: dec2):
   CipherCaesar::decipher( message: enc3, Key: 3, &: dec3);
   std::cout << enc3 << std::endl << std::endl:
   std::cout << dec1 << std::endl:
   std::cout << dec2 << std::endl;
   std::cout << dec3 << std::endl << std::endl;
```

Часть шифра Атбаша:

```
enc1 = "", enc2 = "", enc3 = "";
dec1 = "", dec2 = "", dec3 = "";
CipherAtbash::cipher( message: msq1, &: enc1);
CipherAtbash::cipher( message: msq2, &: enc2);
CipherAtbash::cipher( message: msq3, &: enc3);
std::cout << enc1 << std::endl;
std::cout << enc2 << std::endl;
std::cout << enc3 << std::endl << std::endl;</pre>
CipherAtbash::cipher( message: enc1, &: dec1);
CipherAtbash::cipher( message: enc2, &: dec2);
CipherAtbash::cipher( message: enc3, &: dec3);
std::cout << dec1 << std::endl;
std::cout << dec2 << std::endl:
std::cout << dec3 << std::endl << std::endl;</pre>
```

Особенности реализации

- Реализовано на С++, поэтому код быстрый
- Функции статические, поэтому экономятся ресурсы. Особенно это актуально если нам нужно шифровать примерно 1000 сообщений в секунду. Для этих 1000 сообщений не будут создаваться 1000 экземпляров классов
- · Все собрано в проект CMake, поэтому код кросс-платформенный



Освоил на практике применение шифрования шифрами Цезаря и Атбаш.