#include <iostream>

#include <string>

#include <fstream>

#include <stdio.h>

#include <string.h>

#include <cstdlib>

#include <windows.h>

using namespace std;

//generowanie napisu początkowego

void wstep()

{

//tworzenie dwóch tablic znaków, z przodu i z tyłu głównego napisu

char tab\_front[16] = { 'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O', '\0' };

char tab\_back[16] = { 'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O', '\0' };

for (int i = 0; i < 16; i++)

{

tab\_front[i] = int(tab\_front[i] + 85); //konwersja znaków na "znaki specjalne"

tab\_back[i] = int(tab\_back[i] + 159); //przy wykorzystaniu tablicy ASCII

}

//wypisanie tablicy znaków z przodu głównego napisu

for (int i = 0; i < 16; i++)

{

cout << tab\_front[i] << " ";

}

cout << " SZYFR CEZARA "; //główny napis wstępu

//wypisanie tablicy znaków z tyłu głównego napisu

for (int i = 0; i < 16; i++)

{

cout << tab\_back[i] << " ";

}

cout << "\n" << endl;

}

void read\_from\_file(string path)

{

//otworzenie pliku do odczytu

ifstream file(path);

//sprawdzenie czy plik jest otwarty

if (file.is\_open())

{

cout << "Plik wejsciowy otwarto pomyslnie!" << endl;

}

else if(!file.is\_open())

{

cout << "Nie mozna otworzyc pliku." << endl;

cout << "Wcisnij enter aby kontynuowac...";

system("PAUSE");

exit(0);

}

file.close();

}

void encryption(string path\_1,string path\_2, int shifts, int uppercase, int lowercase)

{

char tab[1024] = { '\0' }; //tablica przechowujaca stringi z pliku

string loading; //string do ktorego pobierana będzie linijka

string result;

ifstream file\_1(path\_1);

while (getline(file\_1, loading)) //wczytywanie tekstu z pliku do stringu linijka po linijce

{

strcpy\_s(tab, loading.c\_str()); //kopiowanie stringa do tablicy znakowej

for (int i = 0; i < loading.length(); i++) //algorytmy szyfrowania/odszyfrowania:

{

if (int(tab[i] >= 65 && tab[i] <= 90)) //zamiana duzych liter

result += char(int(tab[i] + shifts - uppercase) % 26 + uppercase);

else if (int(tab[i] >= 32 && tab[i] <= 64))

result += int(tab[i] = tab[i]); //jesli znaki specjalne, to nie zmieniaj

else

result += char(int(tab[i] + shifts - lowercase) % 26 + lowercase); //zamiana malych liter

}

}

ofstream file\_2(path\_2); //zapis rezultatu do drugiego pliku

if (file\_2)

{

file\_2 << result;

file\_2.close();

}

else

{

cout << "BLAD!!!" << endl;

exit(0);

}

file\_1.close();

}

void interpretation(string path)

{

ifstream file(path);

if (file.is\_open())

{

//udane otworzenie pliku

string loading; //zmienna do przechowywania odczytanych wierszy tekstu

while (getline(file, loading)) //czyta aż do końca pliku

{

cout << loading << endl; // wyświetlenie w konsoli

}

}

else

{

cout << "BŁĄD: nie można otworzyć pliku do odczytu." << endl;

exit(0);

}

file.close();

}

int main()

{

HANDLE hOut;

hOut = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(hOut, FOREGROUND\_GREEN | FOREGROUND\_INTENSITY);

wstep(); //wywolanie funkcji przechowującej napis

string path\_1 = "file\_1.txt";

string path\_2 = "file\_2.txt";

read\_from\_file(path\_1); //sprawdzenie, czy pliki zostały otworzone prawidłowo

read\_from\_file(path\_2);

cout << "\n";

int uppercase, lowercase; //zmienne pomocnicze przy szyfrowaniu/odszyfrowaniu

int condition; //zmienna ktora zawiera warunek: szyfrowanie lub odszyfrowanie

int shifts; //rodzaj szyfrowania/odszyfrowania(ilosc przesuniec)

string error = "BLAD!!!\n";

cout << "Podaj o ile znakow chcesz zaszyfrowac/odszyfrowac tekst [-26...26]: ";

cin >> shifts;

cout << "" << endl;

while (shifts> 26 || shifts<-26)

{

SetConsoleTextAttribute(hOut, FOREGROUND\_RED);

cout << error << endl;

cout << "Wprowadz ponownie klucz przesuniecia: ";

cin >> shifts;

cout << "" << endl;

}

SetConsoleTextAttribute(hOut, FOREGROUND\_GREEN | FOREGROUND\_INTENSITY);

cout << "Szyfrowanie wcisnij [1] || Odszyfrowanie wcisnij [2]: ";

cin >> condition;

cout << " " << endl;

switch(condition)

{

case 1: //szyfrowanie

uppercase = 65; //pierwsza duża litera w kodzie ASCII

lowercase = 97;//pierwsza mała litera w kodzie ASCII

encryption(path\_1, path\_2, shifts, uppercase, lowercase);

break;

case 2: //odszyfrowanie

uppercase = 90; //ostatnia duża litera w kodzie ASCII

lowercase = 122; //ostatnia mała litera w kodzie ASCII

encryption(path\_1, path\_2, -shifts, uppercase, lowercase);

break;

default:

cout << error;

exit(0);

break;

}

if (condition==1)

{

cout << "Szyfrowanie zakonczone. \n" << endl;

cout << "Tekst przed szyfrowaniem: ";

interpretation(path\_1);

cout << "Tekst po szyfrowaniu: ";

interpretation(path\_2);

}

else if(condition == 2)

{

cout << "Odszyfrowanie zakonczone. \n" << endl;

cout << "Tekst przed odszyfrowaniem: ";

interpretation(path\_1);

cout << "Tekst po odszyfrowaniu: ";

interpretation(path\_2);

}

else

cout<<error<<endl;

cout << " " << endl;

system("PAUSE");

}