Министерство науки и высшего образования РФ

Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа киберфизических систем и управления

УДК 004.421

УТВЕРЖДАЮ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_г.

Отчет

по дисциплине «Теория и технология программирования»

Лабораторная работа №10 «Класс для работы со строками CString» вариант 17

Выполнил:

Студент

гр.3530902/90001 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Непушкин С.А. подпись, дата

Доцент ВШКФСиУ,

Кандидат технических наук

С. В. Хлопин \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Санкт-Петербург

2020

**1. Цель работы.**

Цель задания - изучить методы работы с классом CString.

**2. Задание:**

Написать текстовый редактор с параметрами (индивидуальным заданием), представленными в системной матрице:

* Добавление в конец
* Замена всех первых строчных букв слов прописными
* Вставка текста из файла

**3. Текст программы**

#include<conio.h>

#include<iostream>

#include<fstream>

#include<string>

#include<vector>

#include<filesystem>

#include<exception>

#include<windows.h>

using namespace std;

struct Text {

vector<string> strings;

void add(string app)

{

strings.push\_back(app);

}

void capitalized()

{

for (int i = 0; i < strings.size(); i++)

capitalize(strings[i]);

}

void print()

{

if (strings.size() != 0)

{

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_TEXT START\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl;

for (int i = 0; i < strings.size(); i++)

cout << strings[i];

cout << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_TEXT END \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl;

}

}

bool isalpha(char c)

{

if (c >= 'a' && c <= 'z')

return true;

if (c >= 'A' && c <= 'Z')

return true;

if (c >= 'а' && c <= 'я')

return true;

if (c >= 'А' && c <= 'Я')

return true;

return false;

}

char toupper(char c)

{

if (c >= 'a' && c <= 'z')

return c - 'a' + 'A';

if (c >= 'а' && c <= 'я')

return c - 'а' + 'А';

return c;

}

void capitalize(string &str)

{

int state = 0;

for (int i = 0; str[i] != '\0'; i++)

{

if (isalpha(str[i]))

{

if (state == 0)

{

str[i] = toupper(str[i]);

}

state = 1;

}

else

{

state = 0;

}

}

}

};

string readString()

{

string result;

char ch;

cout << "Введите 'Ctrl+Z' для окончания ввода клавиш:" << endl;

do

{

switch (ch = \_getch())

{

case 13:

result += "\r\n";

case 26:

cout << endl;

break;

case 0:

break;

default:

result = result + ch;

cout << ch;

}

} while (ch != 26); //27 - esc, 13 - enter, 26 - ctrl+Z

return result;

}

bool fileExists(const string &name)

{

struct stat buf;

return (stat(name.c\_str(), &buf) == 0);

}

void appendFromFile(Text &text)

{

string filename;

cout << "Введите имя файла : ";

cin >> filename;

while (!fileExists(filename))

{

cout << "Файл не найден. Введите имя файла заново:" << endl;

cin >> filename;

}

ifstream is(filename);

while (!is.eof())

{

string temp;

getline(is, temp);

temp = temp + "\r\n";

text.add(temp);

}

is.close();

}

void printMenu()

{

system("cls");

cout << "------------------------------------------------------" << endl;

cout << "| ВЫБОР ДЕЙСТВИЯ |" << endl;

cout << "|0 - Выход |" << endl;

cout << "|3 - Добавление в конец текста |" << endl;

cout << "|7 - Замена всех первых строчных букв слов прописными|" << endl;

cout << "|9 - Вставка текста из файла |" << endl;

cout << "------------------------------------------------------" << endl;

}

bool process(Text &text)

{

switch (\_getch() - '0')

{

case 0:

return false;

case 3:

text.add(readString());

break;

case 7:

text.capitalized();

break;

case 9:

appendFromFile(text);

break;

}

return true;

}

void setRus()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

HANDLE hStdOut = GetStdHandle(STD\_OUTPUT\_HANDLE);

CONSOLE\_FONT\_INFOEX cfie;

ZeroMemory(&cfie, sizeof(cfie));

cfie.cbSize = sizeof(cfie);

lstrcpyW(cfie.FaceName, L"Lucida Console");

SetCurrentConsoleFontEx(hStdOut, false, &cfie);

}

int main()

{

setRus();

Text text;

do

{

printMenu();

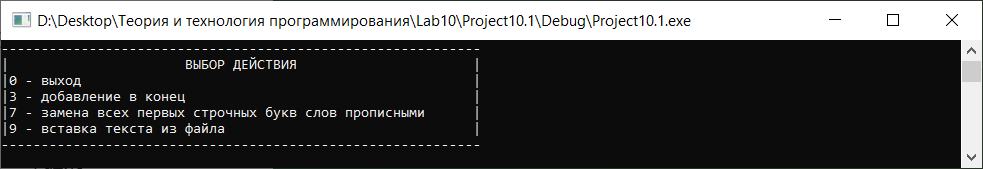
text.print();

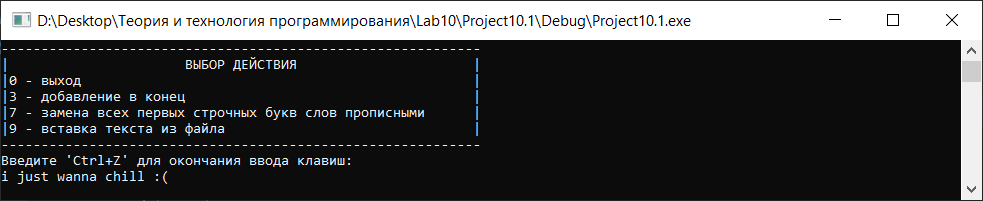
} while (process(text));

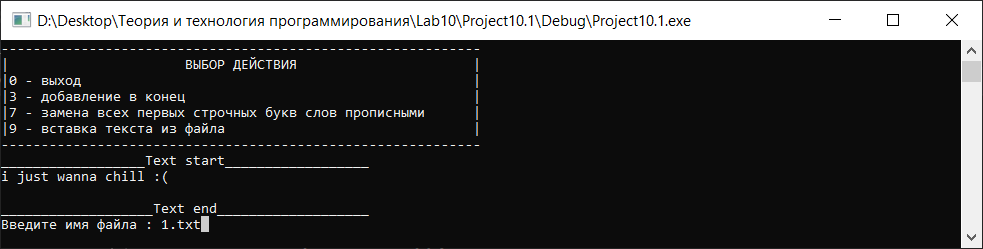
return 0;

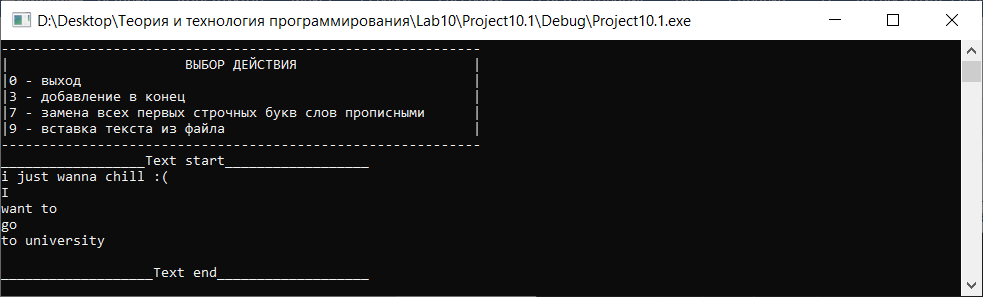
}

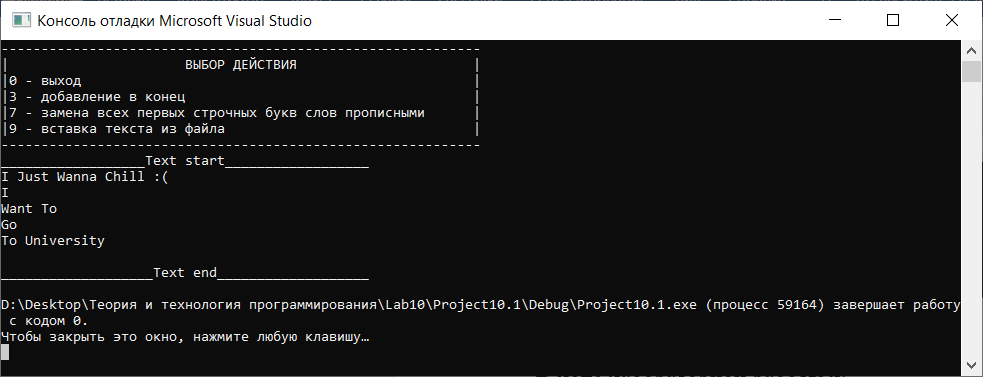
**4. Пример работы программы**











**Вывод**

В ходе лабораторной работы я:

-узнал о функции «system(“cls”)»

-столкнулся с такой проблемой, как вставка текста из файла, но успешно решил ее

-вспомнил, что текстовые файлы нужно закрывать после их использования

-узнал о такой функции, как «strings.push\_back()»

-столкнулся с такой проблемой, закрытия консоли после ввода данных, но успешно решил ее

-решил проблему с выводом русских символов