**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСТИЕТ**

Лабораторная работа №1

по дисциплине «Программирование»

Группа: АВТ-907

Студент: Курасова А.Р.

Преподаватель: Булатов А.Д

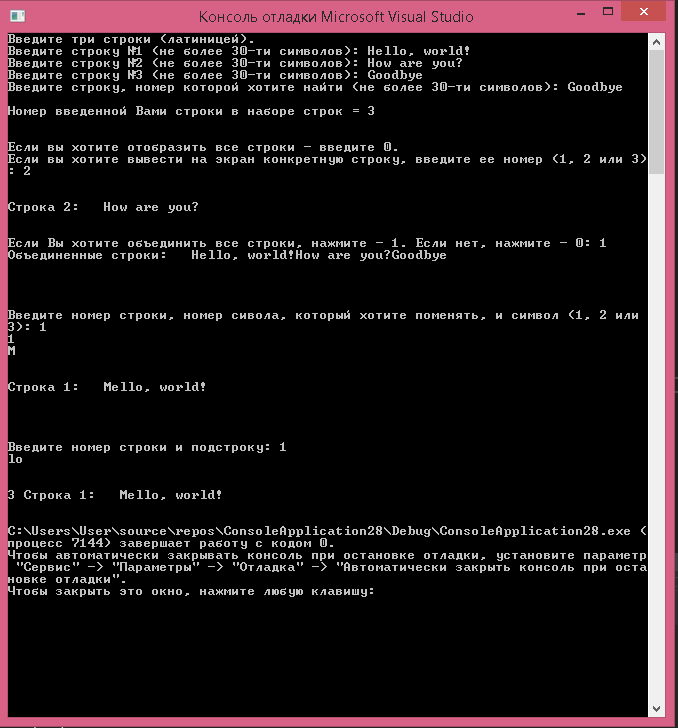
НОВОСИБИРСК 2020

**Задание**

Вариант 10.

Постpоить класс для pаботы со cтpоками. Класс должен включать следующие поля: указатель на стpоку, ее длину. Класс должен обеспечивать пpостейшие функции для pаботы со строками: изменение стpоки, вывод стpоки на экран, нахождение подстpоки в cтpоке, объединение двух строк и т.д.

**Пример работы программы**

****

**Текст программы**

#include <iostream>

#include <new>

#include <cstring>

#include <cstdlib>

#include <string>

#include <iostream>

#pragma warning(disable: 4996)

using namespace std;

class Strings

{

private:

int amount;//количество строк вцелом

static int length;//длина строк

int x; //для записи возвращаемого значения функцией searchStr()

int varNumber;//номер строки, которую пользователь захочет вывести на экран

bool varConcat;//выбор пользователя (объединять строки или нет)

//указатели на начало строк, которые будут заполняться пользователем

char\* str1;

char\* str2;

char\* str3;

char\* strSearch;//для поиска заданной строки

char\* strConcat;//для записи всех введенных строк в одну строку

public:

Strings()//количество строк и максимальная длина строк

{

amount = 3;//количество

}

int searchStr()//метод, который определяет наличие в наборе строк заданной строки

{

//строка strSearch и все остальные будут заданы в методе setString()

//там же и будет вызван ткущий метод searchStr()

if (strcmp(strSearch, str1) == 0) return 1;// функция strcmp сравнит две строки и вернет 0, если они идентичны

else if (strcmp(strSearch, str2) == 0) return 2;

else if (strcmp(strSearch, str3) == 0) return 3;

else return -1;

}

void setString()// задание строки, с заданным номером;

{

str1 = new char[length];//выделяем память для строк из 30-ти символов и один символ для '\0'

str2 = new char[length];

str3 = new char[length];

strSearch = new char[length];

strConcat = new char[length \* amount];//выделяем память для строки, в которую можно записать все введенные пользователем строки

for (int i = 0; i < length; i++)//заполним все строки символами \0, чтобы очистить память от ненужных значений в строках

{

str1[i] = '\0';

str2[i] = '\0';

str3[i] = '\0';

strSearch[i] = '\0';

}

for (int i = 0; i < length \* amount; i++)//заполним строку символами \0

{

strConcat[i] = '\0';

}

cout << "Введите три строки (латиницей)." << endl;

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

{

cout << "Введите строку №" << i + 1 << " (не более 30-ти символов): ";

if (i == 0)cin.getline(str1, length);

if (i == 1)cin.getline(str2, length);

if (i == 2) cin.getline(str3, length);

}

cout << "Введите строку, номер которой хотите найти (не более 30-ти символов): ";

cin.getline(strSearch, length);

x = searchStr();//присвоим переменной х значение, которое вернет функция searchStr()

if (x == 1) cout << "\nНомер введенной Вами строки в наборе строк = 1" << endl;

else if (x == 2) cout << "\nНомер введенной Вами строки в наборе строк = 2" << endl;

else if (x == 3) cout << "\nНомер введенной Вами строки в наборе строк = 3" << endl;

else cout << "\nСтрока не найдена!!!\n\n";

}

void getString()//вывод строк на экран

{

cout << "\n\n";

cout << "Если вы хотите отобразить все строки - введите 0.";

cout << "\nЕсли вы хотите вывести на экран конкретную строку, введите ее номер (1, 2 или 3): ";

cin >> varNumber;

cout << "\n\n";

if (varNumber == 1) cout << "Строка 1: " << str1 << endl << endl;

else if (varNumber == 2) cout << "Строка 2: " << str2 << endl << endl;

else if (varNumber == 3) cout << "Строка 3: " << str3 << endl << endl;

else//в данном случае если пользователь ввел 0

{

cout << "Строка 1: " << str1 << endl << endl;

cout << "Строка 2: " << str2 << endl << endl;

cout << "Строка 3: " << str3 << endl << endl;

}

}

void searchString()

{

string sub;

cout << "\n\n";

cout << "\nВведите номер строки и подстроку: ";

cin >> varNumber;

cin >> sub;

size\_t n = 0, from = 0;

cout << "\n\n";

string n1 = str1;

string n2 = str2;

string n3 = str3;

if (varNumber == 1) {

while ((n = n1.find(sub, from)) != string::npos) {

cout << n << " ";

from = n + sub.size();

}

cout << "Строка 1: " << str1 << endl << endl;

}

else if (varNumber == 2) {

while ((n = n2.find(sub, from)) != string::npos) {

cout << n << " ";

from = n + sub.size();

}

cout << "Строка 2: " << str2 << endl << endl;

}

else if (varNumber == 3) {

while ((n = n3.find(sub, from)) != string::npos) {

cout << n << " ";

from = n + sub.size();

} cout << "Строка 3: " << str3 << endl << endl;

}

}

void concat()//метод объединения строк;

{

cout << "\nЕсли Вы хотите объединить все строки, нажмите - 1. Если нет, нажмите - 0: ";

cin >> varConcat;

if (!varConcat) cout << "Строки не будут объединены!" << endl;//если varConcat = 0(ложь)

else //если varConcat = 1

{

strcat(strConcat, str1);

strcat(strConcat, str2);

strcat(strConcat, str3);

cout << "Объединенные строки: " << strConcat << endl << endl;//если пользователь ранее объединил строки, выводим их одной строкой

}

}

void change()//изменение строк

{

int n;

char m;

cout << "\n\n";

cout << "\nВведите номер строки, номер сивола, который хотите поменять, и символ (1, 2 или 3): ";

cin >> varNumber;

cin >> n >> m;

cout << "\n\n";

if (varNumber == 1) {

str1[n-1] = m;

cout << "Строка 1: " << str1 << endl << endl; }

else if (varNumber == 2) {

str2[n-1] = m;

cout << "Строка 2: " << str2 << endl << endl;

}

else if (varNumber == 3) {

str3[n-1] = m; cout << "Строка 3: " << str3 << endl << endl;

}

else//в данном случае если пользователь ввел 0

{

cout << "Строка 1: " << str1 << endl << endl;

cout << "Строка 2: " << str2 << endl << endl;

cout << "Строка 3: " << str3 << endl << endl;

}

}

void del()//метод очистки памяти по завершении работы программы

{

delete strConcat;

delete strSearch;

delete str3;

delete str2;

delete str1;

}

};

int Strings::length = 31;//максимальная длина

int main()

{

setlocale(LC\_ALL, "rus");

Strings strings;//создаем объект

strings.setString();//вызываем метод задания строк в котором также вызывается метод сравнения строк

strings.getString();// вывод строк на экран

strings.concat();// метод конкатенации (пользователь выбирает объединять строки или нет)

strings.change();//изменение строк

strings.searchString();//поиск подстроки

strings.del();//чистим память

return 0;

}

**Вывод**

В ходе выполнения лабораторной работы я ознакомилась с структурой класса, механизмами создания и использования, описания членов-данных класса и методов доступа к ним.