МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ

И.С. ТУРГЕНЕВА»

#### Отчет по лабораторной работе №7

по дисциплине «Императивное программирование»

|  |
| --- |
| Работу выполнил(а) студент(ка)  **Тупикин М.Н.**  группа **91-ИТ**  **г. Орел** |
| Работу проверил(а)  **Амелина О.В.** |

Отметка о зачете дата «\_\_\_\_» \_\_\_\_\_\_\_\_\_20 г.

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**г. Орел, 20 г.**

Лабораторная работа №7

Тема: «Регистрация участников при помощи бинарного дерева»

#include <iostream>

#include <string>

#include <clocale>

#include <cstring>

#include <sstream>

using namespace std;

struct Person{

Person \*left;

char \*surname;

char \*organization;

Person \*right;

};

struct Cities{

Cities \*left;

char \*city;

Person \*pers;

Cities \*right;

};

Cities \*Tree;

string city, surname, organization;

Person\* InsertPerson(Person \*P, char \*val)

{

if (P == NULL)

{

P = (Person\*)malloc(sizeof( Person));

P->surname = strdup(val);

P->left = NULL;

P->right = NULL;

}

else if (\*(P->surname) > \*(val))

P->right = InsertPerson(P->right, val);

else if (\*(P->surname) < \*(val))

P->left = InsertPerson(P->left, val);

return P;

}

Cities\* InsertCities(Cities \*tr, char \*g, char \*f)

{

if (tr == NULL)

{

tr = (Cities\*)malloc(sizeof( Cities));

tr->city = strdup(g);

tr->left = NULL;

tr->right = NULL;

tr->pers = NULL;

tr->pers = InsertPerson(tr->pers, f);

}

else if (\*(tr->city) < \*(g))

tr->right = InsertCities(tr->right, g, f);

else if (\*(tr->city) > \*(g))

tr->left = InsertCities(tr->left, g, f);

else

tr->pers = InsertPerson(tr->pers, f);

return tr;

}

void PrintPeople(Person \*per){

if (per != NULL){

PrintPeople (per->right);

cout<<" Участники: "<<per->surname<<endl;

PrintPeople (per->left);

}

}

void Print ( Cities \*cit){

cout<<endl;

if (cit != NULL){

Print(cit->left);

cout<<" Город: "<<cit->city<<endl;

PrintPeople(cit->pers);

Print (cit->right);

}

}

int main(){

setlocale(0, "");

Tree = nullptr;

int com = 0;

while (true){

cout<<"1. Для продолжения нажмите '1'\n";

cout<<"2. Для вывода списка нажмите '2'\n";

cout<<"3. Для выхода нажмите '3'\n";

cin>>com;

switch(com){

case 1:{

cout<<" Введите город: "<<endl;

cin>>city;

/\*Перевожу из string в \*char\*/

char \*g = new char;

strcpy(g, city.c\_str());

/\*Перевожу из string в \*char\*/

cout<<" Введите фирму: "<<endl;

cin>>organization;

cout<<" Введите фамилию: "<<endl;

cin>>surname;

/\*Перевожу из string в \*char\*/

char \*f = new char;

strcpy(f, surname.c\_str());

/\*Перевожу из string в \*char\*/

Tree = InsertCities(Tree, g, f);

cout<<endl;

break;

}

case 2:

cout<<endl;

Print(Tree);

break;

case 3:

exit(0);

break;

default:

cout<<"Неверная команда!\n";

break;

}

}

system("pause");

return 0;

}





