Федеральное государственное автономное

образовательное учреждение

высшего профессионального образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра вычислительной техники

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7**

**“Обработка односвязных списков”**

Преподаватель Покидышева Л. И. \_

подпись, дата

Студент Максимович Ю. В. Группа КИ15-07Б \_

подпись, дата

Красноярск 2016

**Цели работы:**

Получить практические навыки решения задач с использованием односвязных списков.

**Порядок выполнения работы:**

Выполнить задания, указанные в экспериментальной части,   подготовить отчет и защитить лабораторную работу преподавателю.

Выполняется только одно из упражнений №7\_1,  №7\_2 или №7\_3 для получения соответственно удовлетворительной,  хорошей или отличной оценки согласно вариантам.

*Внимание!* В данной работе разработать **укрупненную** графическую схему алгоритма, т.е. отражающую только **основные действия**.

**Упражнение №7\_2**

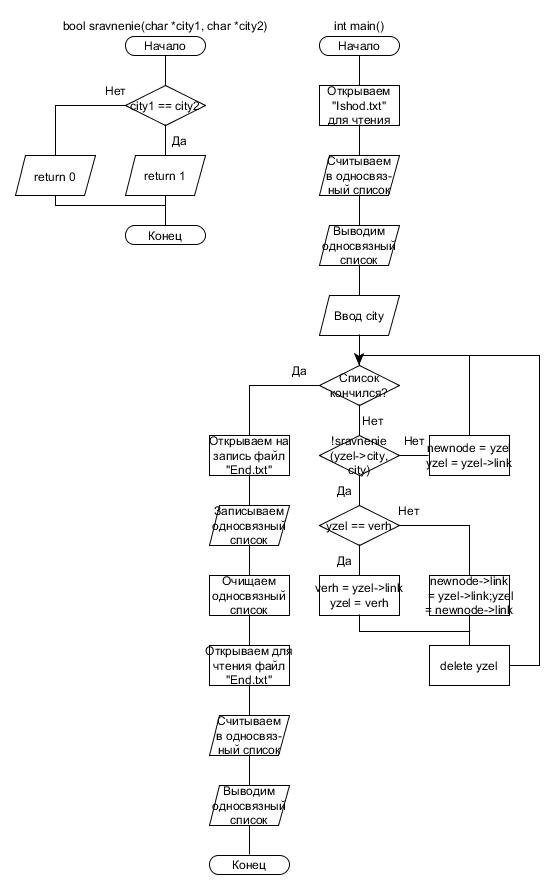
**(по индивидуальным вариантам, 80% - хор)**

С помощью текстового редактора создать файл, в котором содержатся данные, указанные в таблице №2 (в соответствии вариантами).  Данные о каждом объекте  располагаются в отдельной строке и разделены пробелами.

Разработать, отладить, продемонстрировать и защитить преподавателю графическую схему алгоритма и программу для решения следующей задачи: прочитать из текстового файла данные в *динамический односвязный список*, отфильтровать данные в соответствии с указаниями в таблице №2 и сохранить отфильтрованные данные в двоичном файле (имя файла вводится с клавиатуры). Вывести содержимое текстового и двоичного файла на монитор в виде таблицы.

|  |  |  |
| --- | --- | --- |
| **3.** | Туристы: фамилия, возраст, город, образование (высшее, среднее, среднее профессиональное) | Данные  о туристах, проживающих в городе, название которого вводится с клавиатуры |

Графическая схема:



Код программы:

#include <iostream>

#include <fstream>

#include <conio.h>

#include <stdio.h>

#include <clocale>

#include <cstdlib>

#include <string.h>

using namespace std;

bool sravnenie(char \*city1, char \*city2)

{

if (strlen(city1) == strlen(city2))

{

for (int i = 0; i < strlen(city1); i++)

{

if (city1[i] != city2[i]) return 0;

}

return 1;

}

else return 0;

}

int main()

{

setlocale(LC\_ALL, "Rus");

struct node

{

char surname[15];

int year;

char city[15];

char education[25];

node \*link;

};

node \*verh, \*yzel;

ifstream txt("Ishod.txt");

verh = NULL;

while (!txt.eof())

{

yzel = new node;

yzel->link = verh;

txt >> yzel->surname;

txt >> yzel->year;

txt >> yzel->city;

txt >> yzel->education;

verh = yzel;

}

txt.close();

cout << "Текстовый файл:" << endl;

yzel = verh;

cout << "Фамилия:" << " " << "Возраст:" << " "

<< "Город:" << " " << "Образование:" << endl;

while (yzel != NULL)

{

cout << yzel->surname << " " << yzel->year << " "

<< yzel->city << " " << yzel->education << endl;

yzel = yzel->link;

}

yzel = verh;

char city[15];

cout << "Введите название города: ";

cin >> city;

node \*newnode;

while (yzel != NULL)

{

if (!sravnenie(yzel->city, city))

{

if (yzel == verh)

{ // Если совпавший - вершина

verh = yzel->link;

delete yzel;

yzel = verh;

}

else

{ // Ecли совпавший - не веpшина

newnode->link = yzel->link;

delete yzel;

yzel = newnode->link;

}

}

else

{ // Eсли не совпавший

newnode = yzel;

yzel = yzel->link;

}

}

newnode = verh;

ofstream bin\_write("End.txt", ios::binary);

while (yzel != NULL)

{

bin\_write << yzel->surname << ' ' << yzel->year

<< ' ' << yzel->city << ' ' << yzel->education;

delete yzel;

newnode = yzel->link;

}

bin\_write.close();

ifstream bin\_read("End.txt", ios::binary);

while (!bin\_read.eof())

{

yzel = new node;

yzel->link = verh;

bin\_read >> yzel->surname;

bin\_read >> yzel->year;

bin\_read >> yzel->city;

bin\_read >> yzel->education;

verh = yzel;

}

bin\_read.close();

cout << "Двоичный файл:" << endl;

cout << "Фамилия:" << " " << "Возраст:" << " "

<< "Город:" << " " << "Образование:" << endl;

yzel = verh;

while (yzel != NULL)

{

cout << yzel->surname << " " << yzel->year << " "

<< yzel->city << " " << yzel->education << endl;

yzel = yzel->link;

}

\_getch();

return 0;

}

Пример выполнения:

