Министерство образования и науки Российской Федерации   
Федеральное государственное бюджетное образовательное учреждение   
высшего образования

«Пермский национальный исследовательский политехнический университет»

Кафедра «Информационные технологии и автоматизированные системы

**Отчёт   
по лабораторной работе №8**

Дисциплина: основы алгоритмизации и программирования

Тема: «Блоковый ввод-вывод»

Вариант №14

Выполнил работу:  
студент группы РИС-20-2б  
Вичугов Алексей Дмитриевич

Проверила:  
доцент кафедры ИТАС  
Полякова Ольга Андреевна

Пермь 2021

**Цель работы**

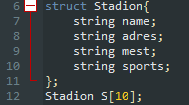
Работа с двоичными файлами, организация ввода-вывода структурированной информации и ее хранение на внешних носителях.

**Постановка задачи**

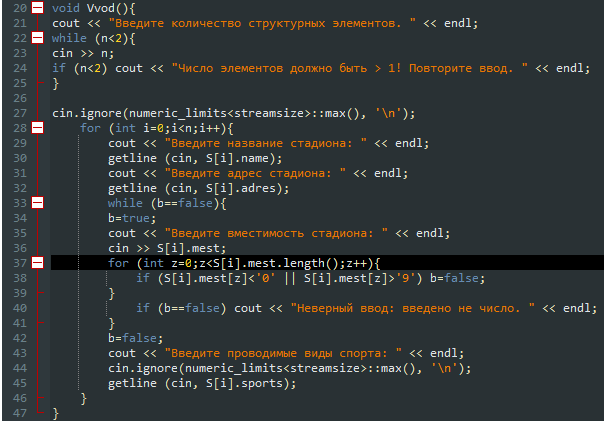
Сформировать двоичный файл из элементов, заданной в варианте структуры, распечатать его содержимое, выполнить удаление и добавление элементов в соответствии со своим вариантом, используя для поиска удаляемых или добавляемых элементов функцию. Формирование, печать, добавление и удаление элементов оформить в виде функций. Предусмотреть сообщения об ошибках при открытии файла и выполнении операций ввода/вывода.

**Анализ задачи**

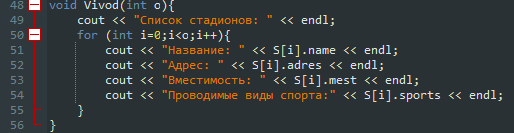
1. Для решения задачи необходимо:
   1. Сформировать структуру данных и создать массив структурных элементов:



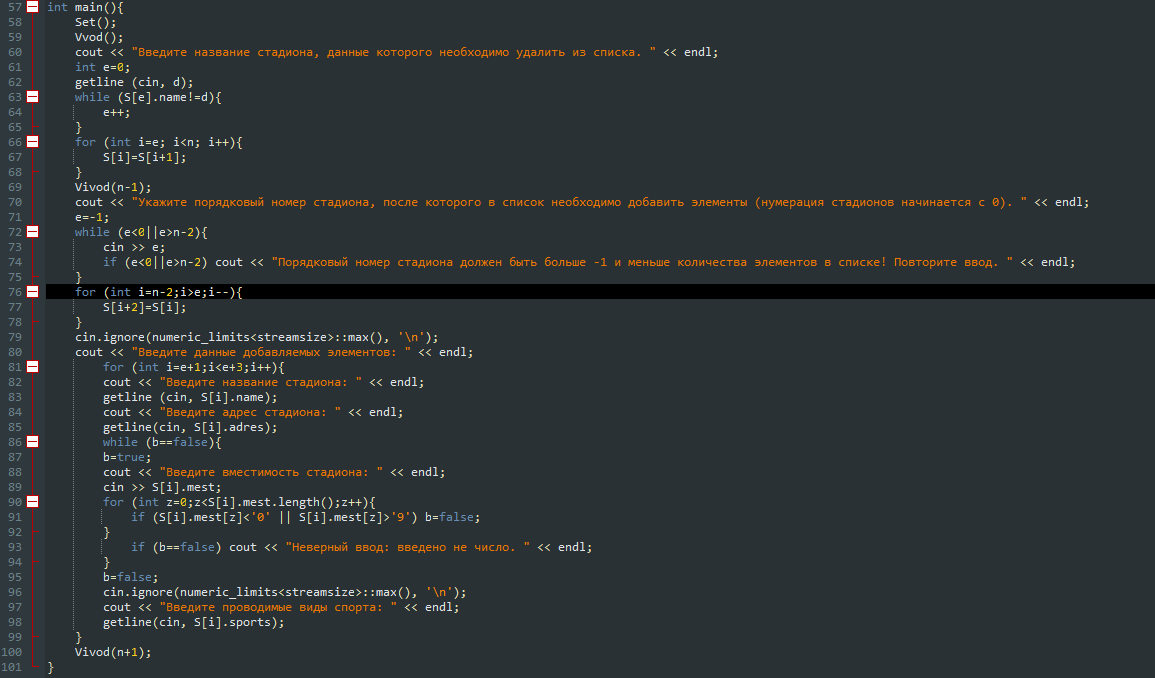
* 1. Создать функцию Vvod(), производящую ввод данных в поля элементов структуры:



* 1. Создать функцию Vivod(), выводящую данные полей элементов структур:



* 1. Реализовать применение этих функций в функции main(), используя их после действий с массивом (сдвига элементов и добавления элементов):



1. В ходе работы были использованы типы данных:
   1. Для полей элементов структуры тип string:  
      string name;  
      string adres;  
      string mest;  
      string sports;
   2. Для вычислительных операций и управления циклами переменные типа int:  
      int n, e, i;
   3. В качестве флажка поискового цикла переменная типа bool:  
      bool b;
   4. Для функции void Vivod используются следующие аргументы:   
      Целочисленная переменная int o:  
      void Vivod(int o)
2. Данные для решения задачи представлены в виде массива структурных элементов:

Stadion S[n+1];

1. Все поля структуры будут иметь тип string. Список полей:

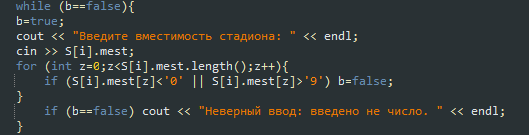
string name;

string adres;

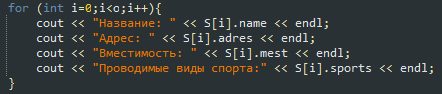
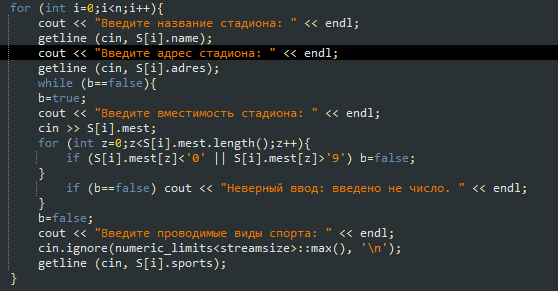
string mest;

string sports;

1. Ввод данных в поля элементов структур производится с помощью операторов и функций:
   1. Оператор cin для ввода целочисленных переменных и поля элементво структуры mest (вместимость стадиона):  
      cin >> e;  
      cin >> mest;
   2. Функция getline() для ввод значений в поля элементов структуры:  
      getline (cin, S[i].name);  
      getline (cin, S[i].adres);  
      getline (cin, S[i].sports);
2. При решении поставленных задач использовались циклы:
   1. Для проверки правильности ввода и подсчёта порядкового номера искомого элемента цикл while:



* 1. Для ввода, вывода и сдвига элементов массива цикл for:



**Полный код программы**

#include <iostream>

#include <string>

#include <limits>

using namespace std;

struct Stadion{

string name;

string adres;

string mest;

string sports;

};

Stadion S[10];

int n=0;

string d;

bool b=false;

void Set(){

const int fon = system("Color F0");

system ("chcp 1251>nul");

}

void Vvod(){

cout << "Введите количество структурных элементов. " << endl;

while (n<2){

cin >> n;

if (n<2) cout << "Число элементов должно быть > 1! Повторите ввод. " << endl;

}

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

for (int i=0;i<n;i++){

cout << "Введите название стадиона: " << endl;

getline (cin, S[i].name);

cout << "Введите адрес стадиона: " << endl;

getline (cin, S[i].adres);

while (b==false){

b=true;

cout << "Введите вместимость стадиона: " << endl;

cin >> S[i].mest;

for (int z=0;z<S[i].mest.length();z++){

if (S[i].mest[z]<'0' || S[i].mest[z]>'9') b=false;

}

if (b==false) cout << "Неверный ввод: введено не число. " << endl;

}

b=false;

cout << "Введите проводимые виды спорта: " << endl;

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

getline (cin, S[i].sports);

}

}

void Vivod(int o){

cout << "Список стадионов: " << endl;

for (int i=0;i<o;i++){

cout << "Название: " << S[i].name << endl;

cout << "Адрес: " << S[i].adres << endl;

cout << "Вместимость: " << S[i].mest << endl;

cout << "Проводимые виды спорта:" << S[i].sports << endl;

}

}

int main(){

Set();

Vvod();

cout << "Введите название стадиона, данные которого необходимо удалить из списка. " << endl;

int e=0;

getline (cin, d);

while (S[e].name!=d){

e++;

}

for (int i=e; i<n; i++){

S[i]=S[i+1];

}

Vivod(n-1);

cout << "Укажите порядковый номер стадиона, после которого в список необходимо добавить элементы (нумерация стадионов начинается с 0). " << endl;

e=-1;

while (e<0||e>n-2){

cin >> e;

if (e<0||e>n-2) cout << "Порядковый номер стадиона должен быть больше -1 и меньше количества элементов в списке! Повторите ввод. " << endl;

}

for (int i=n-2;i>e;i--){

S[i+2]=S[i];

}

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите данные добавляемых элементов: " << endl;

for (int i=e+1;i<e+3;i++){

cout << "Введите название стадиона: " << endl;

getline (cin, S[i].name);

cout << "Введите адрес стадиона: " << endl;

getline(cin, S[i].adres);

while (b==false){

b=true;

cout << "Введите вместимость стадиона: " << endl;

cin >> S[i].mest;

for (int z=0;z<S[i].mest.length();z++){

if (S[i].mest[z]<'0' || S[i].mest[z]>'9') b=false;

}

if (b==false) cout << "Неверный ввод: введено не число. " << endl;

}

b=false;

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите проводимые виды спорта: " << endl;

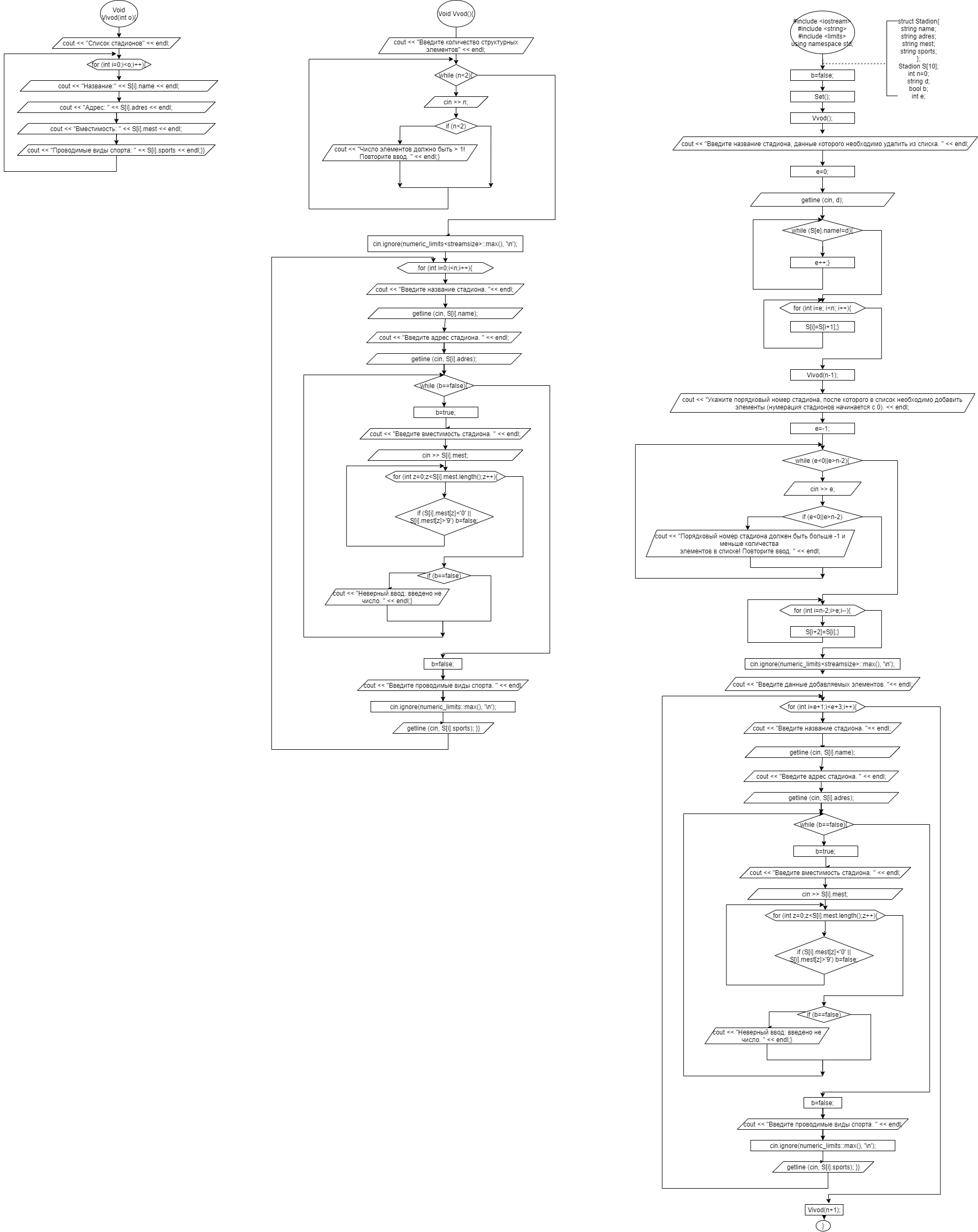
getline(cin, S[i].sports);

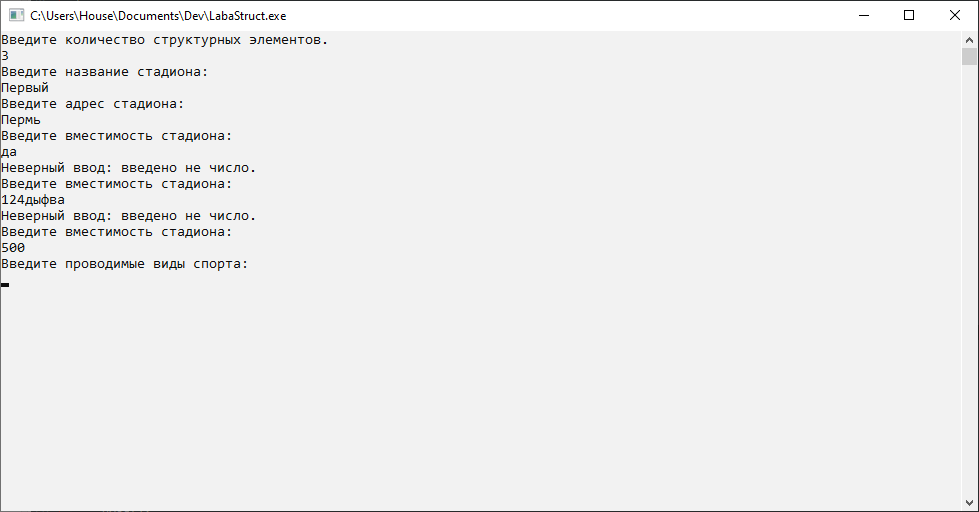
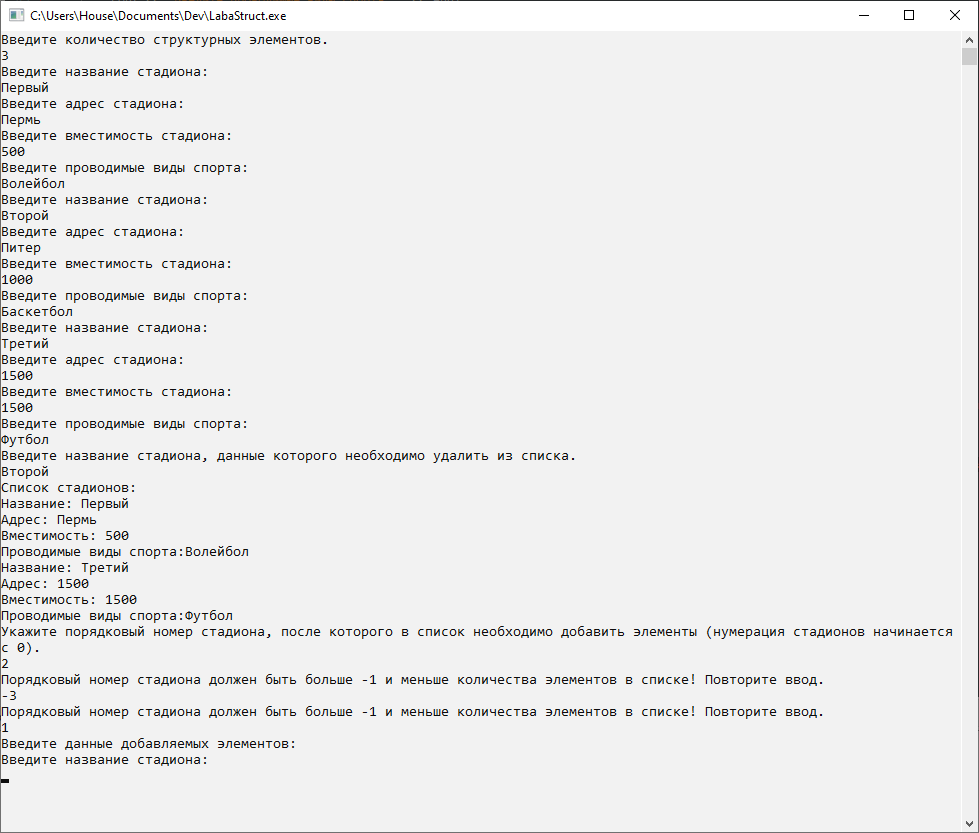
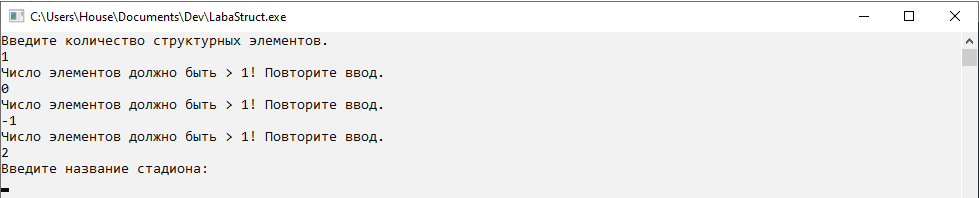
}

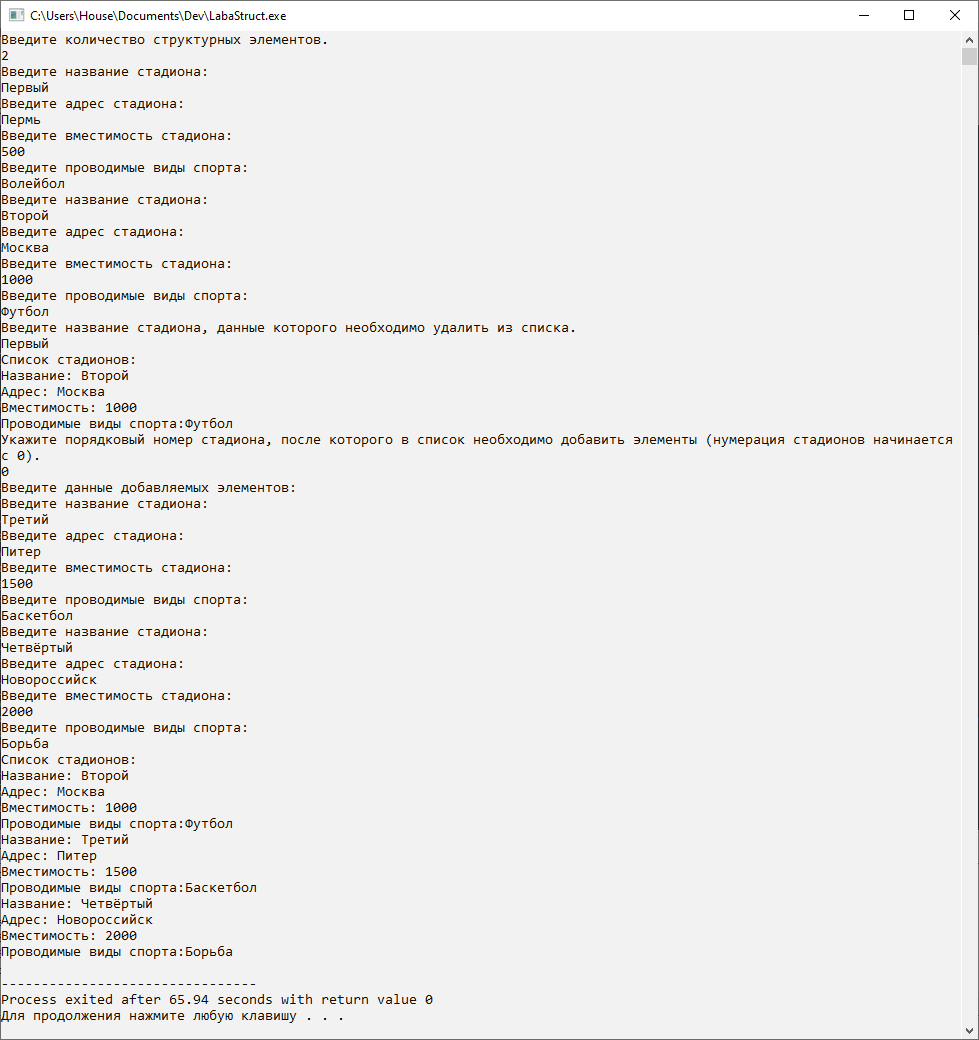
Vivod(n+1);

}

**Блок-схема**



**Результаты тестов**

****