Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное учреждение высшего образования  
«Пермский национальный исследовательский политехнический университет»

Электротехнический факультет  
Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т  
по лабораторной работе**

Дисциплина: «Основы алгоритмизации и программирования»

Выполнил  
студент группы КЗИ-21-1б  
Полковников Т.Ю.

Проверил  
доцент кафедры ИТАС  
ст. преп. Яруллин Д.В.

Пермь, 2022

## 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 решения задачи:

Структура “Информация”:

* носитель;
* объем;
* название;
* автор.

Удалить первый элемент с заданным объемом информации, добавить элемент перед элементом с указанным номером.

## Описание переменных:

size – размер массива структур

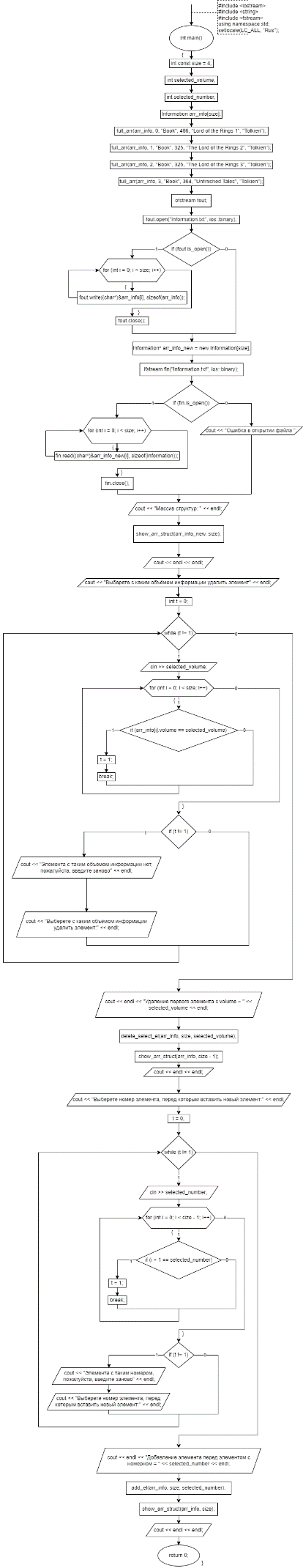
selected\_volume – заданный объем информации

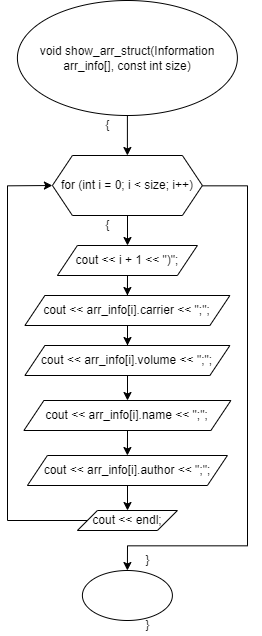
selected\_number – указанный номер элемента

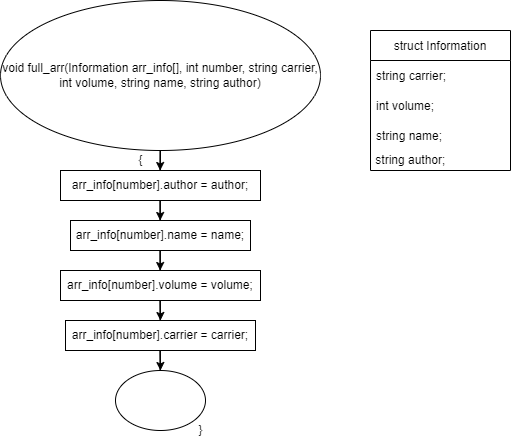
arr\_info – массив структур

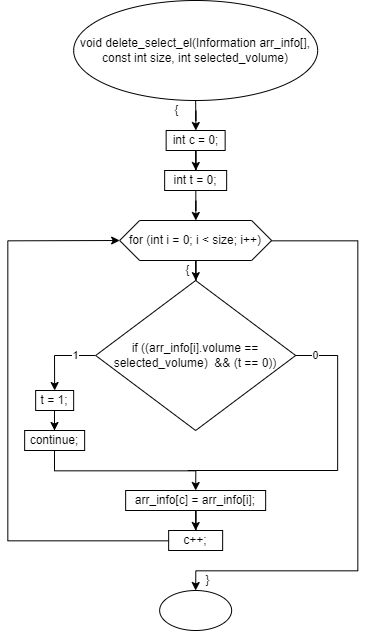
arr\_info\_new – новый массив структур

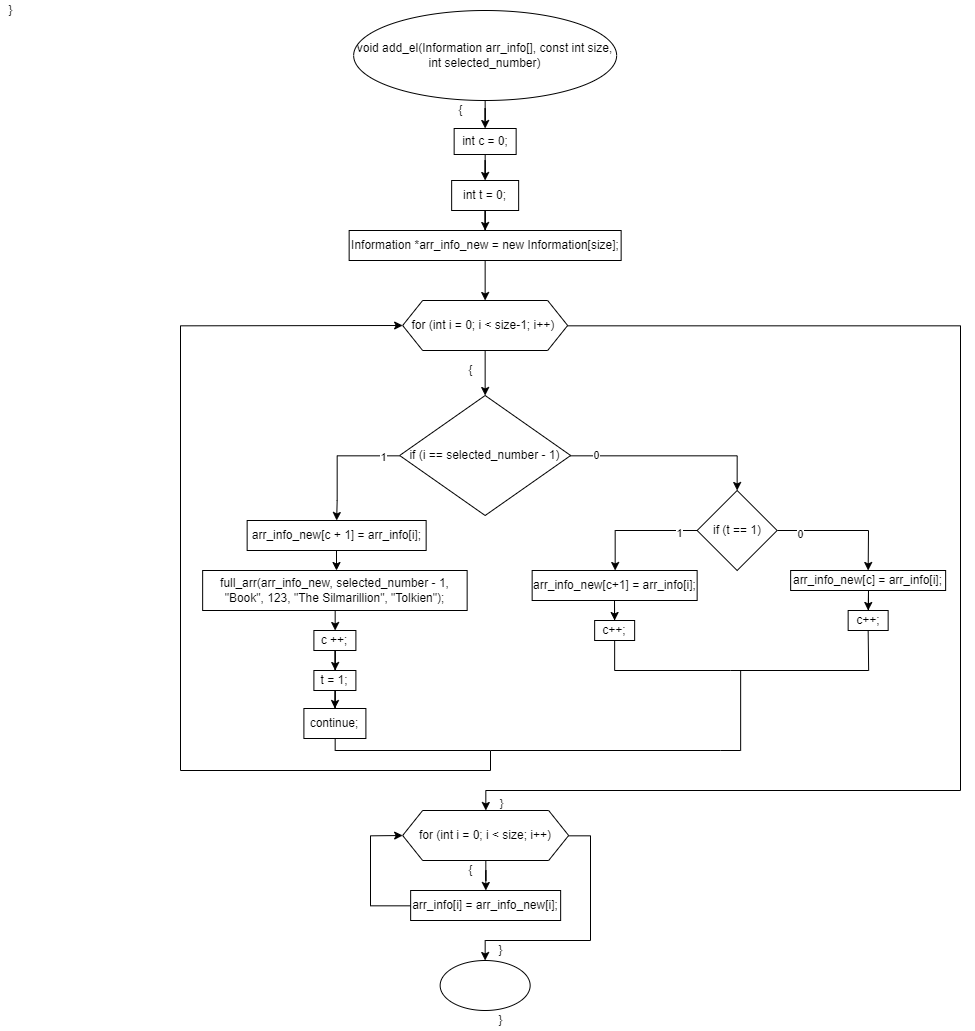
**Блок схема**











## Код программы

#include <iostream>

#include <string>

#include <fstream>

using namespace std;

struct Information {

string carrier;

int volume;

string name;

string author;

};

void full\_arr(Information arr\_info[], int number, string carrier, int volume, string name, string author) {

arr\_info[number].author = author;

arr\_info[number].name = name;

arr\_info[number].volume = volume;

arr\_info[number].carrier = carrier;

}

void show\_arr\_struct(Information arr\_info[], const int size) {

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

cout << i + 1 << ")";

cout << arr\_info[i].carrier << ";";

cout << arr\_info[i].volume << ";";

cout << arr\_info[i].name << ";";

cout << arr\_info[i].author << ";";

cout << endl;

}

}

void delete\_select\_el(Information arr\_info[], const int size, int selected\_volume) {

int c = 0;

int t = 0;

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

if ((arr\_info[i].volume == selected\_volume) && (t == 0)) {

t = 1;

continue;

}

arr\_info[c] = arr\_info[i];

c++;

}

}

void add\_el(Information arr\_info[], const int size, int selected\_number) {

int c = 0;

int t = 0;

Information\* arr\_info\_new = new Information[size];

for (int i = 0; i < size - 1; i++) {

if (i == selected\_number - 1) {

arr\_info\_new[c + 1] = arr\_info[i];

full\_arr(arr\_info\_new, selected\_number - 1, "Book", 123, "The Silmarillion", "Tolkien");

c++;

t = 1;

continue;

}

else {

if (t == 1) {

arr\_info\_new[c + 1] = arr\_info[i];

c++;

}

else {

arr\_info\_new[c] = arr\_info[i];

c++;

}

}

}

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

arr\_info[i] = arr\_info\_new[i];

}

}

int main()

{

setlocale(LC\_ALL, "Rus");

int const size = 4;

int selected\_volume;

int selected\_number;

Information arr\_info[size];

full\_arr(arr\_info, 0, "Book", 466, "Lord of the Rings 1", "Tolkien");

full\_arr(arr\_info, 1, "Book", 325, "The Lord of the Rings 2", "Tolkien");

full\_arr(arr\_info, 2, "Book", 325, "The Lord of the Rings 3", "Tolkien");

full\_arr(arr\_info, 3, "Book", 364, "Unfinished Tales", "Tolkien");

ofstream fout;

fout.open("Information.txt", ios::binary);

if (fout.is\_open())

{

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

{

fout.write((char\*)&arr\_info[i], sizeof(arr\_info));

}

fout.close();

}

Information\* arr\_info\_new = new Information[size];

ifstream fin("Information.txt", ios::binary);

if (fin.is\_open())

{

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

fin.read((char\*)&arr\_info\_new[i], sizeof(Information));

}

fin.close();

}

else {

cout << "Ошибка в открытии файла ";

}

cout << "Массив структур: " << endl;

show\_arr\_struct(arr\_info\_new, size);

cout << endl;

cout << "Выберете с каким объёмом информации удалить элемент" << endl;

int t = 0;

while (t != 1) {

cin >> selected\_volume;

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

if (arr\_info[i].volume == selected\_volume) {

t = 1;

break;

}

}

if (t != 1) {

cout << "Элемента с таким объёмом информации нет, пожалуйста, введите заново" << endl;

cout << "Выберете с каким объёмом информации удалить элемент:" << endl;

}

}

cout << endl << "Удаление первого элемента с volume = " << selected\_volume << endl;

delete\_select\_el(arr\_info, size, selected\_volume);

show\_arr\_struct(arr\_info, size - 1);

cout << endl << endl;

cout << "Выберете номер элемента, перед которым вставить новый элемент:" << endl;

t = 0;

while (t != 1) {

cin >> selected\_number;

for (int i = 0; i < size - 1; i++) {

if (i + 1 == selected\_number) {

t = 1;

break;

}

}

if (t != 1) {

cout << "Элемента с таким номером, пожалуйста, введите заново" << endl;

cout << "Выберете номер элемента, перед которым вставить новый элемент:" << endl;

}

}

cout << endl << "Добавление элемента перед элементом с номерном = " << selected\_number << endl;

add\_el(arr\_info, size, selected\_number);

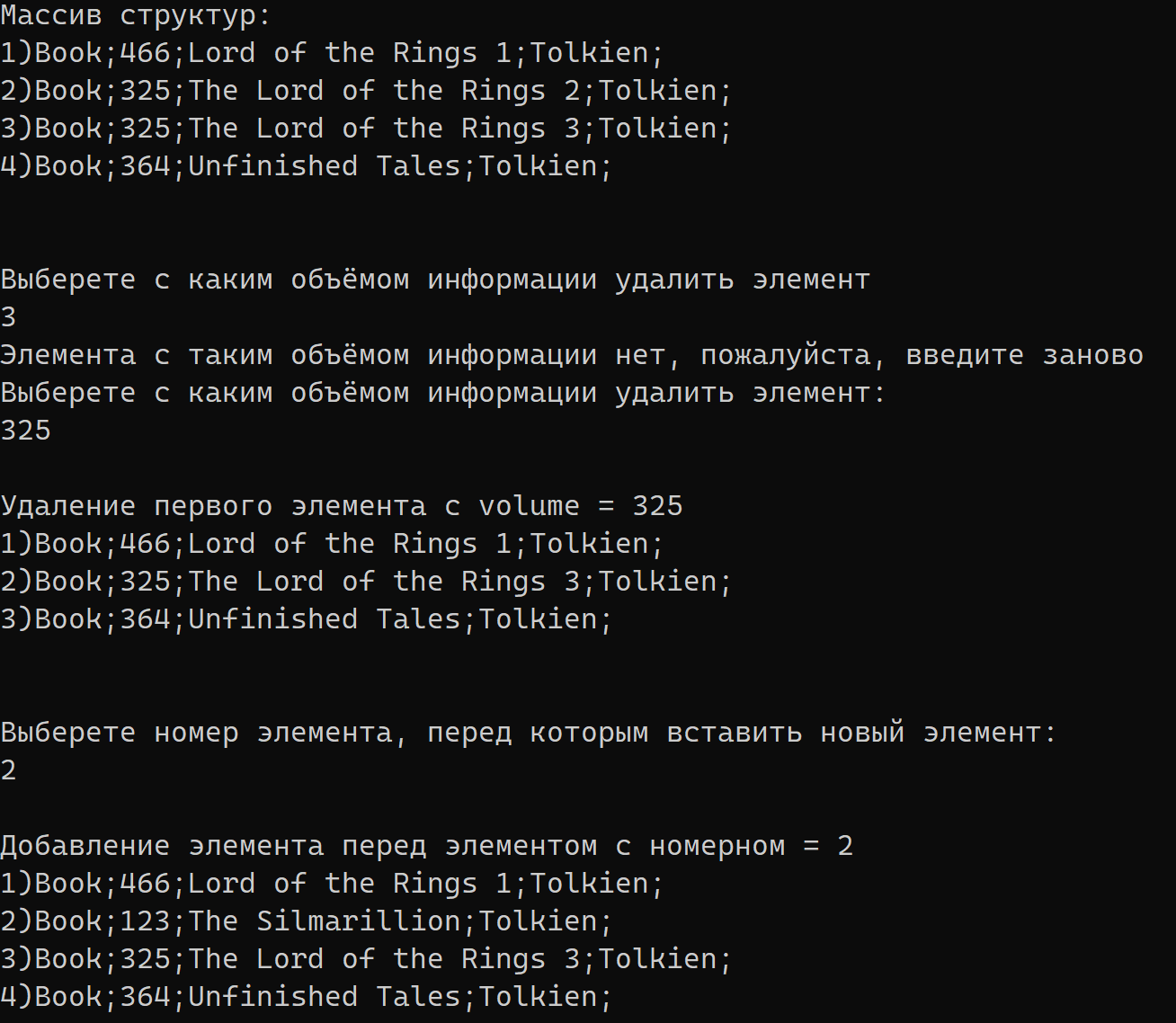
show\_arr\_struct(arr\_info, size);

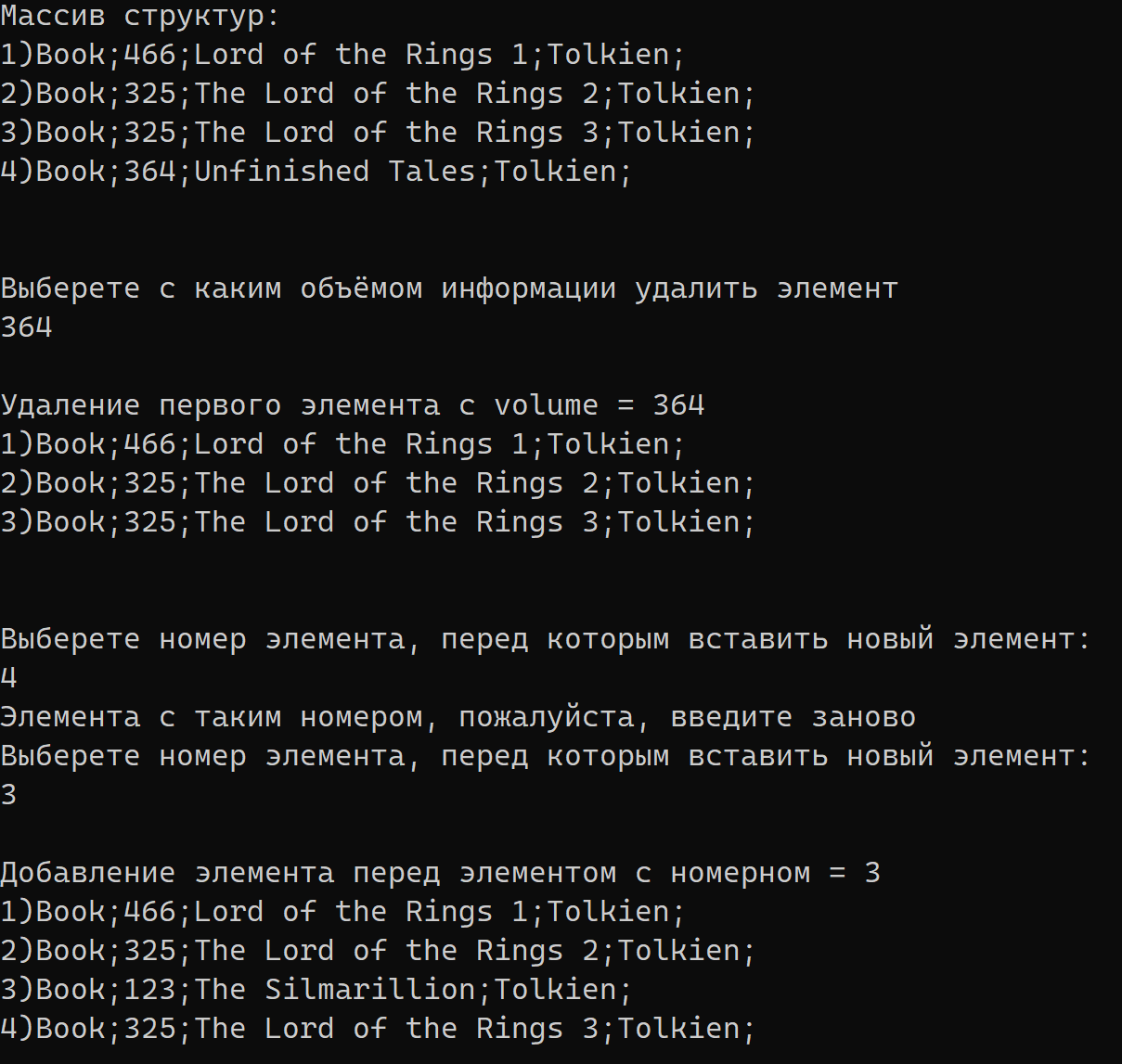
cout << endl << endl;

return 0;

}

Скриншоты результатов:





Анализ результатов:

Программа работает правильно. Выводится: массив структур, массив структур после удаления элемента, массив структур после добавления элемента.