

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
Пермский национальный исследовательский политехнический университет  
Электротехнический факультет  
Кафедра информационных технологий и автоматизированных систем

Лабораторная работа №8  
**"Блоковый ввод-вывод"**  
Вариант: 12

Выполнил студент ИВТ-24-26:  
Шишкин Максим Григорьевич

\_\_\_\_\_  
(дата, подпись)

Проверил доцент кафедры ИТАС:  
Полякова Ольга Андреевна

\_\_\_\_\_  
(дата, подпись)

Пермь 2025

## Содержание

1 Постановка задачи.....	3
2 Код на с++.....	4-5
3 Результат работы программы .....	6
3 Блок-схема.....	7-11
Ссылка на github.....	11

## **1 Постановка задачи**

Структура "Музыкальный диск":

- название;
- автор;
- продолжительность;
- цена.

Удалить первый элемент с заданной продолжительностью, добавить 2 элемента после элемента с заданным номером.

## 2 Код на C++

```
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  struct MusicDisk {
7      string title;    // Название
8      string author;   // Автор
9      int duration;    // Продолжительность в минутах
10     double price;    // Цена
11 };
12 // Функция для вывода всех дисков
13 void printDisks(const MusicDisk disks[], int size) {
14     for (int i = 0; i < size; ++i) {
15         cout << i + 1 << ". " << disks[i].title << " by " << disks[i].author
16             << ", Duration: " << disks[i].duration << " min, Price: $" << disks[i].price << endl;
17     }
18 }
19 // Функция для удаления первого элемента с заданной продолжительностью
20 void removeByDuration(MusicDisk disks[], int& size, int targetDuration) {
21     bool found = false; // Флаг для отслеживания удаления
22     for (int i = 0; i < size; ++i) {
23         if (!found && disks[i].duration == targetDuration) {
24             found = true; // Помечаем, что элемент найден
25         }
26         if (found && i < size - 1) {
27             disks[i] = disks[i + 1]; // Сдвиг оставшихся элементов
28         }
29     }
30     if (found) {
31         --size; // Уменьшаем размер массива только после полного прохода
32     }
33 }
34 // Функция для добавления двух новых элементов после заданной позиции
35 void addAfterPosition(MusicDisk disks[], int& size, int maxSize, int position) {
36     MusicDisk newDisk1, newDisk2;
37
38     cout << "Введите данные для первого нового диска:\n";
39     cout << "Название: ";
40     cin >> newDisk1.title;
41     cout << "Автор: ";
42     cin >> newDisk1.author;
43     cout << "Продолжительность (мин): ";
44     cin >> newDisk1.duration;
45     cout << "Цена: ";
```

```

46     cin >> newDisk1.price;
47
48     cout << "Введите данные для второго нового диска:\n";
49     cout << "Название: ";
50     cin >> newDisk2.title;
51     cout << "Автор: ";
52     cin >> newDisk2.author;
53     cout << "Продолжительность (мин): ";
54     cin >> newDisk2.duration;
55     cout << "Цена: ";
56     cin >> newDisk2.price;
57
58     for (int i = size - 1; i > position; --i) {
59         disks[i + 2] = disks[i];
60     }
61
62     disks[position + 1] = newDisk1;
63     disks[position + 2] = newDisk2;
64     size += 2;
65 }
66 int main() {
67     system("chcp 1251");
68     system("cls");
69     int n; // Количество элементов массива
70     const int maxSize = 100; // Максимальный размер массива
71     cout << "Введите количество музыкальных дисков: ";
72     cin >> n;
73     MusicDisk* disks = new MusicDisk[maxSize]; // Создание динамического массива
74     // Ввод данных для каждого элемента массива
75     for (int i = 0; i < n; ++i) {
76         cout << "Введите данные для диска #" << i + 1 << endl;
77         cout << "Название: ";
78         cin >> disks[i].title;
79         cout << "Автор: ";
80         cin >> disks[i].author;
81         cout << "Продолжительность (мин): ";
82         cin >> disks[i].duration;
83         cout << "Цена: ";
84         cin >> disks[i].price;
85     }
86     // Удаление элемента с заданной продолжительностью
87     int targetDuration;
88     cout << "\nВведите продолжительность для удаления: ";
89     cin >> targetDuration;
90     removeByDuration(disks, n, targetDuration);
91     // Добавление двух новых элементов после указанной позиции
92     int positionToAdd;
93     cout << "\nВведите номер элемента (начиная с 1), после которого добавить два новых диска: ";
94     cin >> positionToAdd;
95     addAfterPosition(disks, n, maxSize, positionToAdd - 1); // Преобразование в индекс
96     // Вывод обновлённого списка
97     cout << "\nОбновлённый список музыкальных дисков:\n";
98     printDisks(disks, n);
99     // Освобождение памяти
100    delete[] disks;
101
102    return 0;
103 }

```



### 3 Результат работы программы

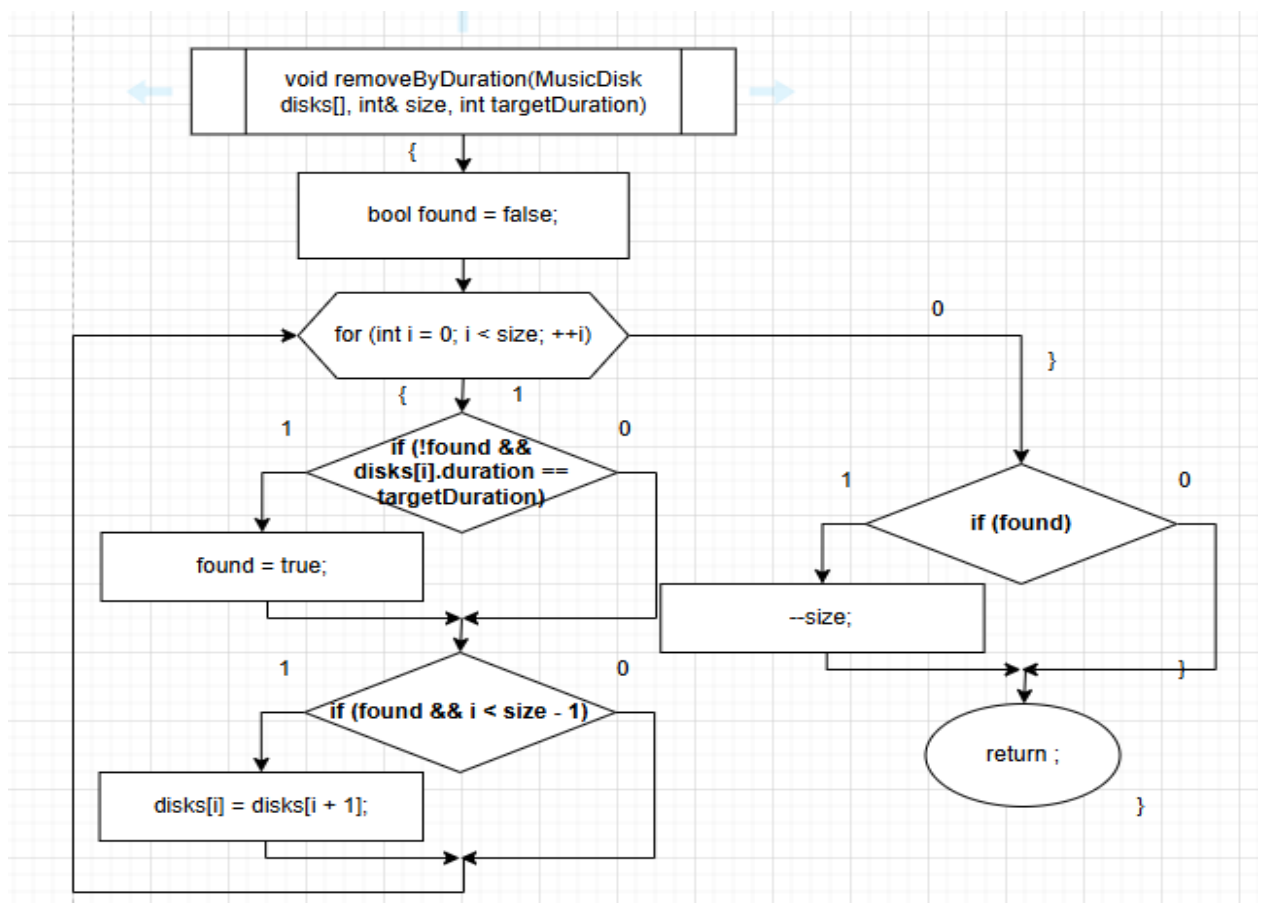
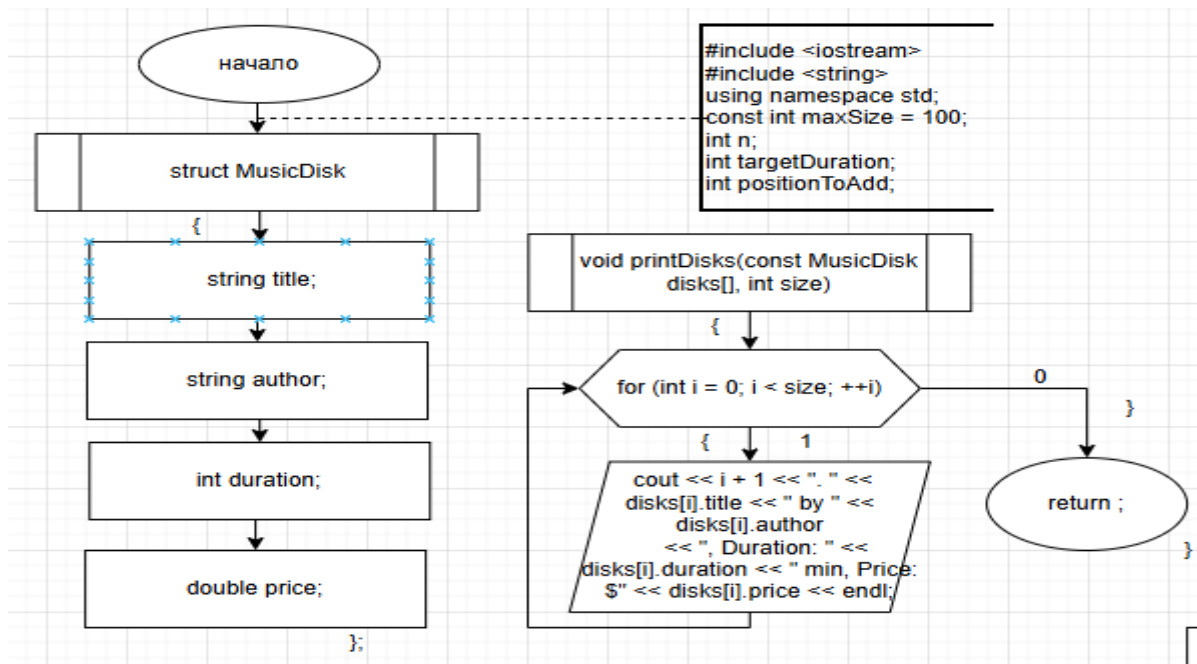
```
Введите количество музыкальных дисков: 2
Введите данные для диска #1
Название: HH
Автор: qw
Продолжительность (мин): 3
Цена: 2
Введите данные для диска #2
Название: KK
Автор: qw
Продолжительность (мин): 7
Цена: 2

Введите продолжительность для удаления: 7

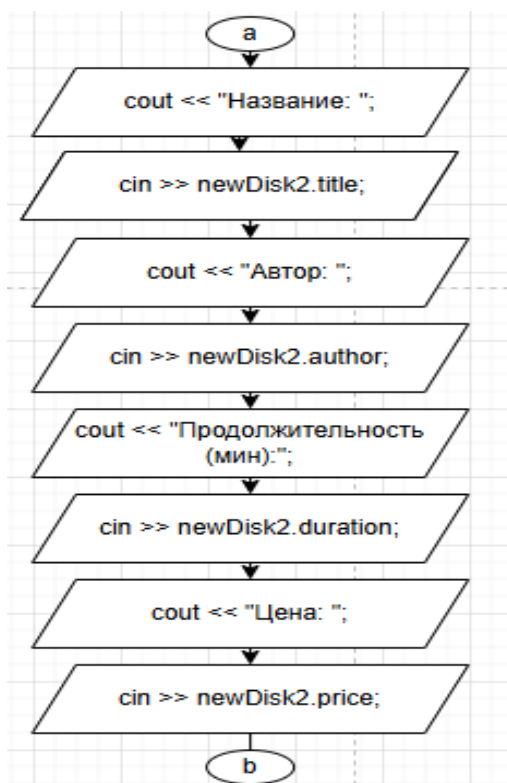
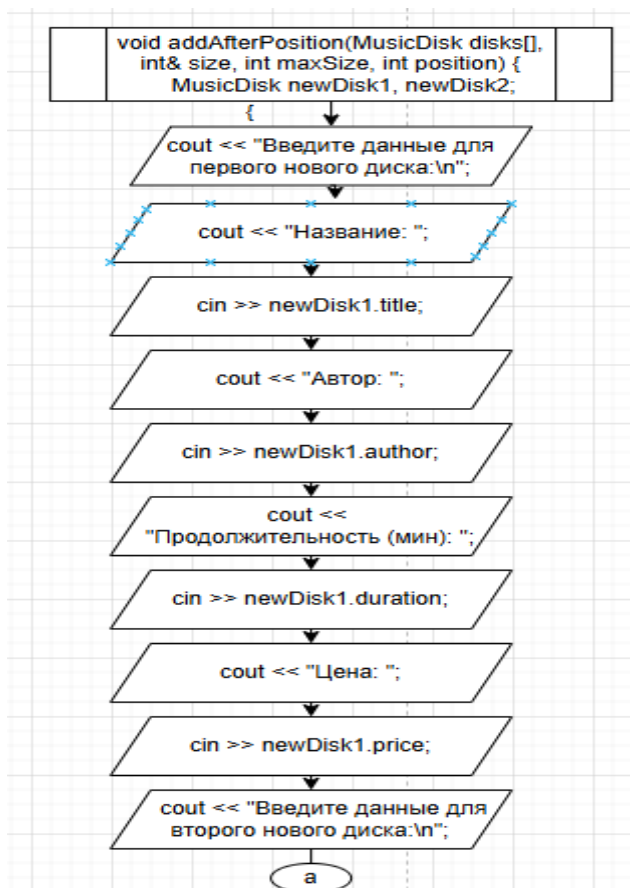
Введите номер элемента (начиная с 1), после которого добавить два новых диска: 1
Введите данные для первого нового диска:
Название: WW
Автор: er
Продолжительность (мин): 4
Цена: 2
Введите данные для второго нового диска:
Название: LL
Автор: re
Продолжительность (мин): 8
Цена: 2

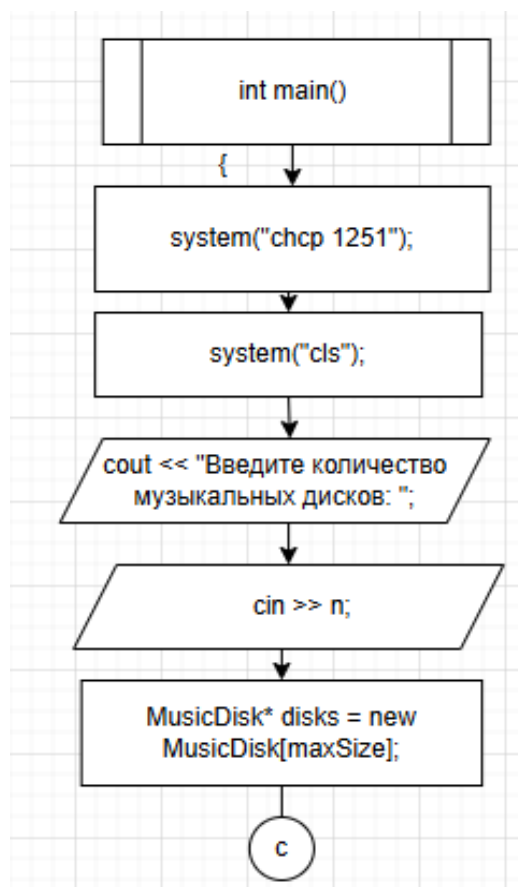
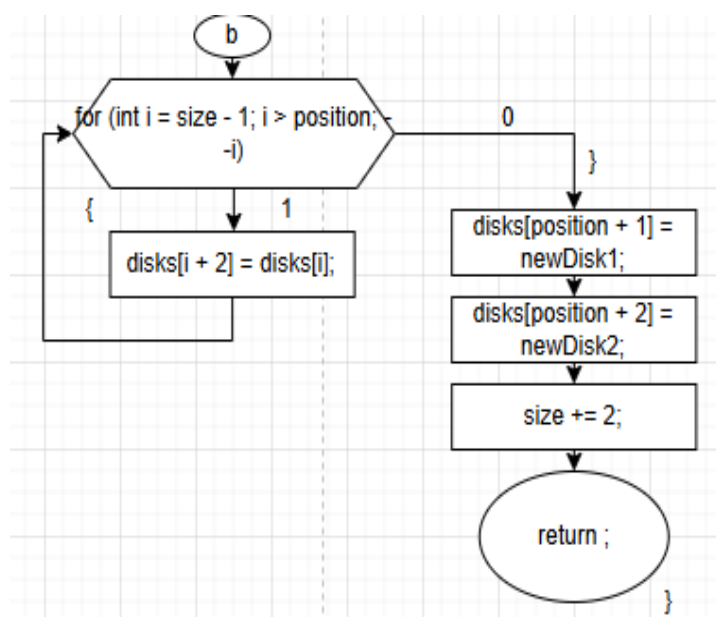
Обновлённый список музыкальных дисков:
1. HH by qw, Duration: 3 min, Price: $2
2. WW by er, Duration: 4 min, Price: $2
3. LL by re, Duration: 8 min, Price: $2
```

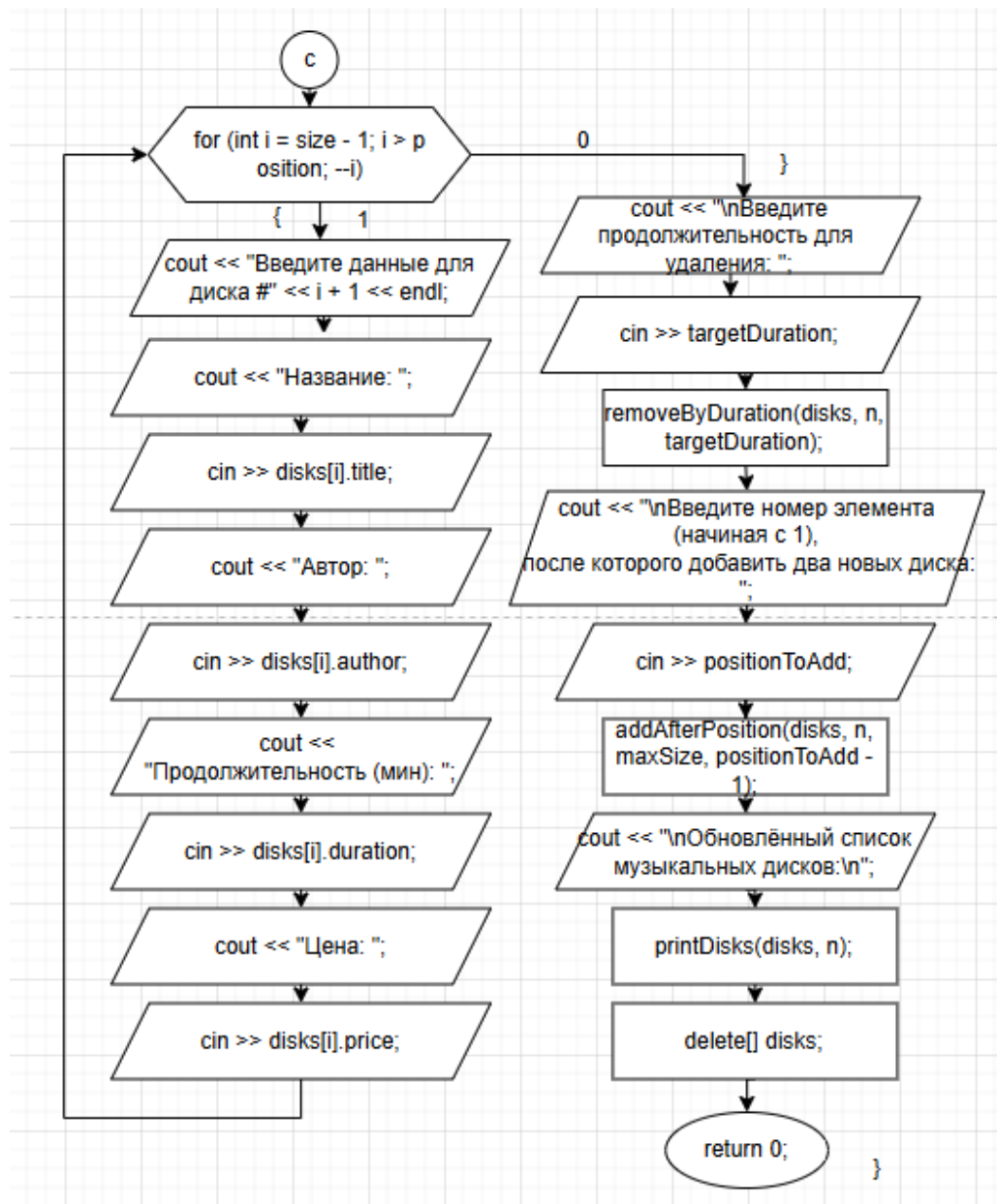
## 4 Блок-схема











## 5 Ссылка на github

ссылка на github - <https://github.com/MAKSPOWERO/mas1>