

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

"Простые поиски"

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 Линейный поиск	4
2.1 Код на с++.....	4
2.2 Результат работы программы	4
2.3 Блок-схема.....	5
3 Бинарный поиск	6
3.1 Код на с++.....	6
3.2 Результат работы программы	7
3.3 Блок-схема.....	7-8
4 Интерполяционный поиск.....	9
4.1 Код на с++.....	9
4.2 Результат работы программы	10
4.3 Блок-схема.....	10-11
5 Ссылка на github.....	11

1 Постановка задачи

Найти в массиве целых чисел заданное число используя три метода поиска: Линейный, Бинарный и Интерполяционный.

2 Линейный поиск

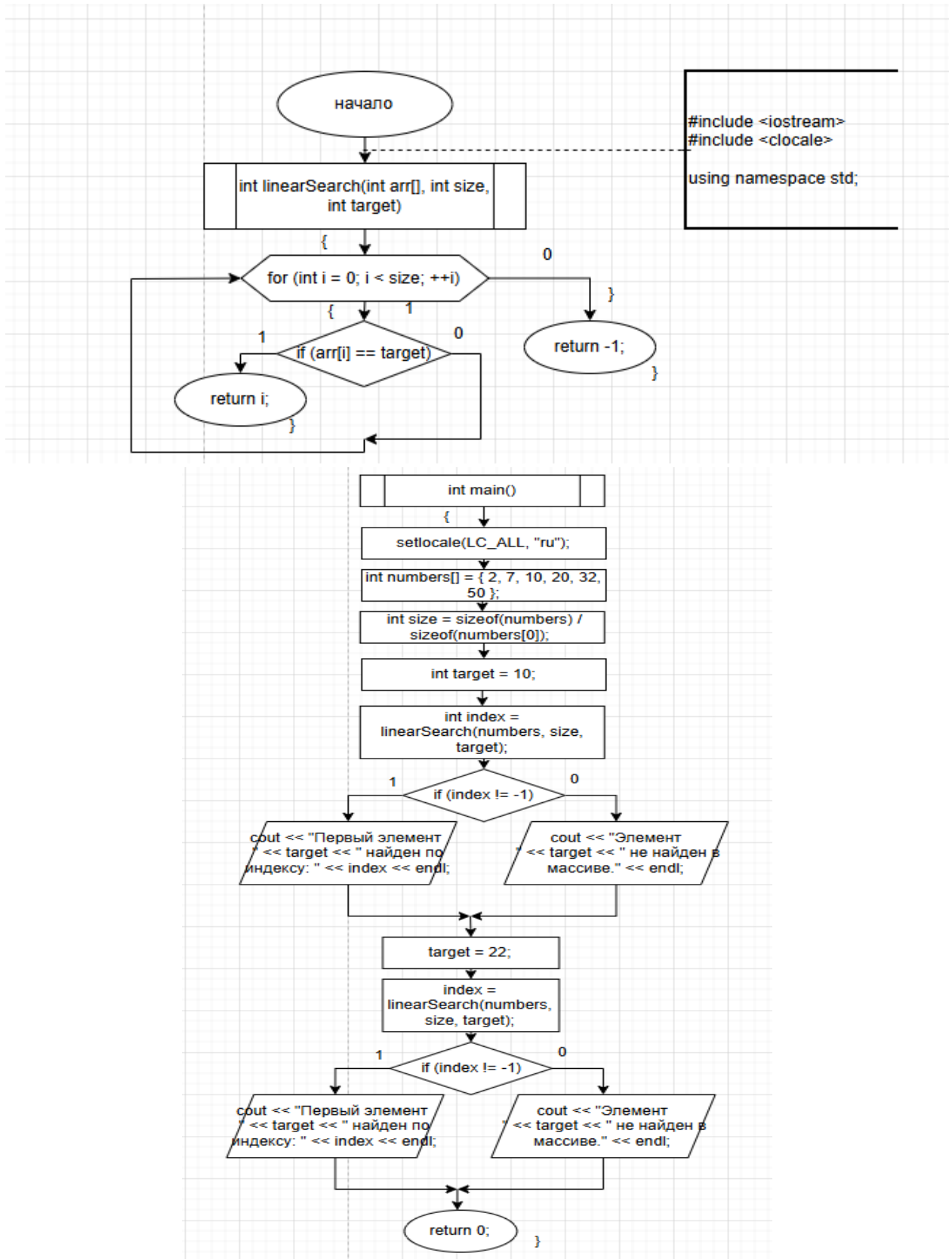
2.1 Код на C++

```
1  #include <iostream>
2  #include <locale>
3
4  using namespace std;
5
6  int linearSearch(int arr[], int size, int target) {
7      for (int i = 0; i < size; ++i) {
8          if (arr[i] == target) {
9              return i;
10         }
11     }
12     return -1; // Элемент не найден
13 }
14
15 int main() {
16     setlocale(LC_ALL, "Russian");
17
18     int numbers[] = { 2, 7, 10, 20, 32, 50 };
19     int size = sizeof(numbers) / sizeof(numbers[0]); // Вычисляем размер массива
20     int target = 10;
21
22     int index = linearSearch(numbers, size, target);
23
24     if (index != -1) {
25         cout << "Элемент " << target << " найден по индексу: " << index << endl;
26     }
27     else {
28         cout << "Элемент " << target << " не найден в массиве." << endl;
29     }
30
31     target = 22;
32     index = linearSearch(numbers, size, target);
33
34     if (index != -1) {
35         cout << "Первый элемент " << target << " найден по индексу: " << index << endl;
36     }
37     else {
38         cout << "Элемент " << target << " не найден в массиве." << endl;
39     }
40
41     return 0;
42 }
```

2.2 Результат работы

```
Элемент 10 найден по индексу: 2
Элемент 22 не найден в массиве.
```

2.3 Блок-схема



3 Бинарный поиск

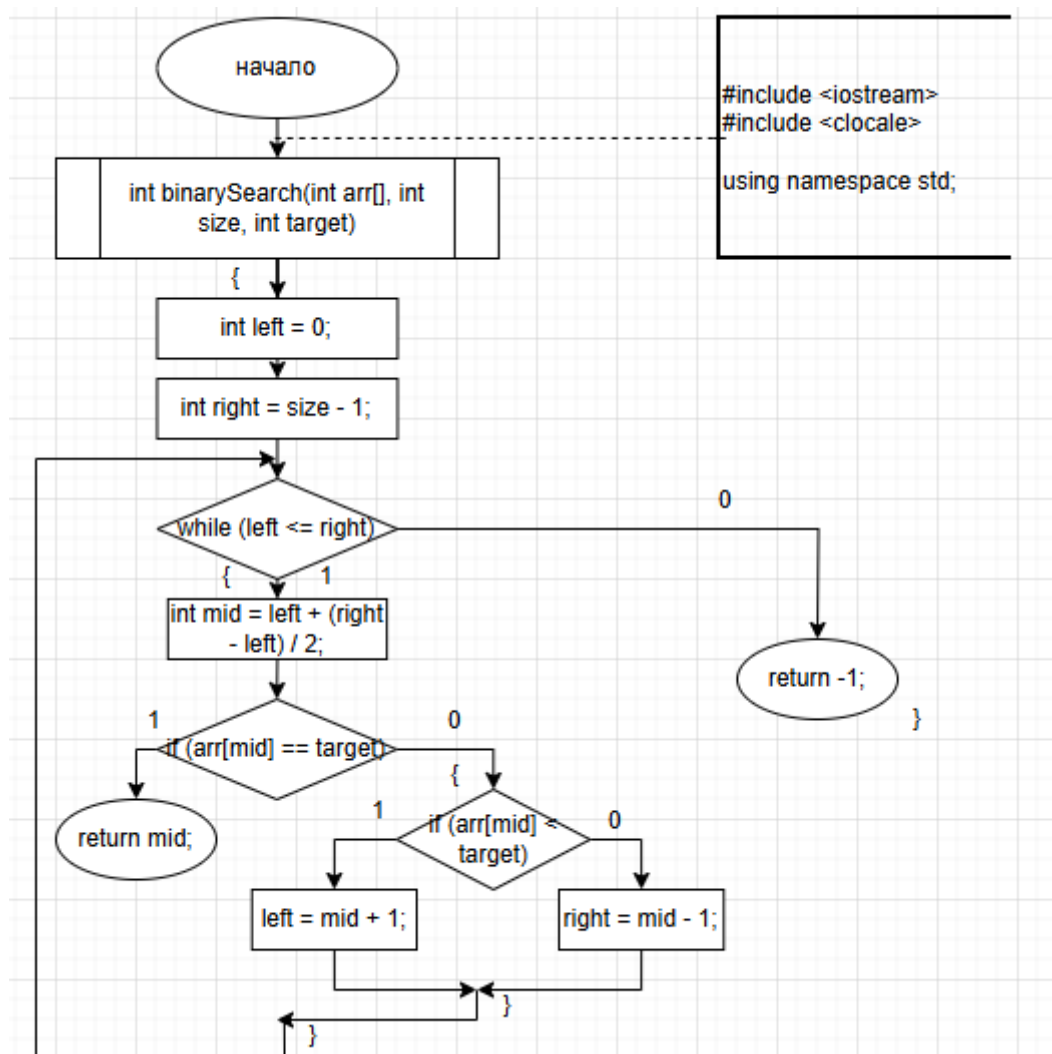
3.1 Код на C++

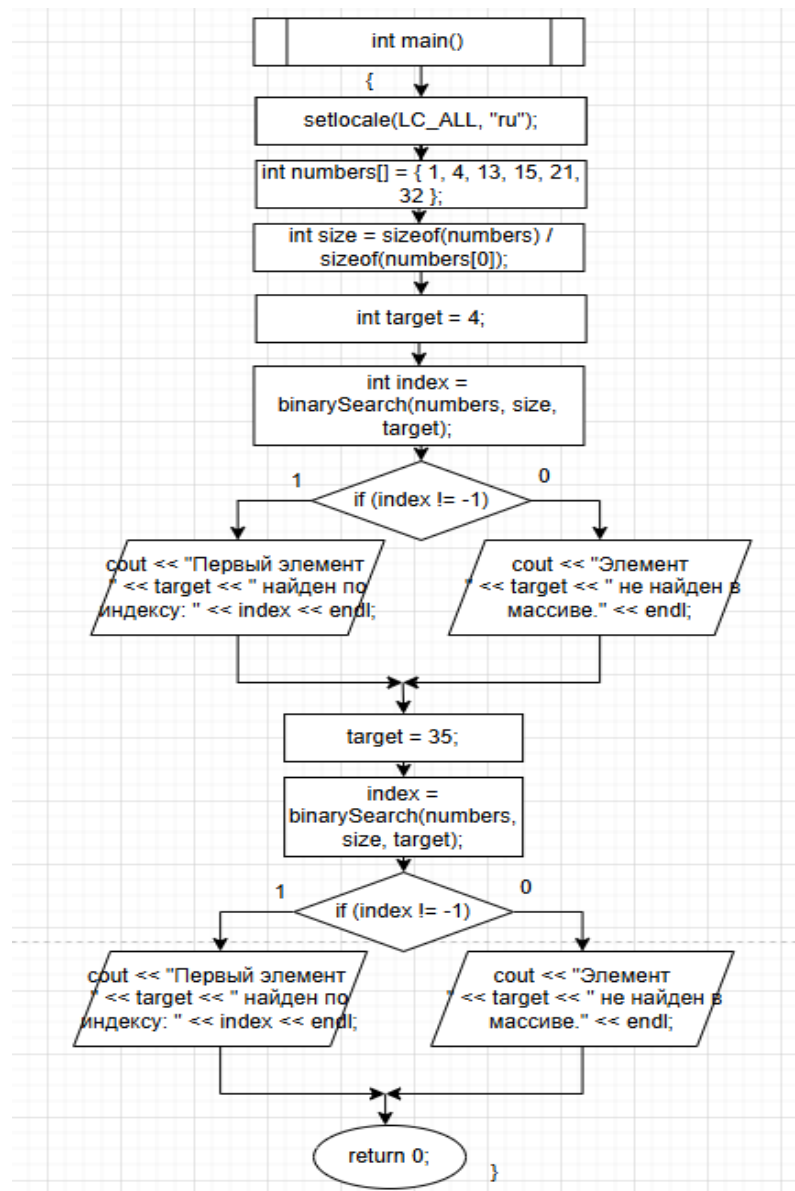
```
1  #include <iostream>
2  #include <locale>
3
4  using namespace std;
5
6
7  int binarySearch(int arr[], int size, int target) {
8      int left = 0;
9      int right = size - 1;
10
11      while (left <= right) {
12          int mid = left + (right - left) / 2; // Предотвращает переполнение
13
14          if (arr[mid] == target) {
15              return mid; // Элемент найден
16          }
17          else if (arr[mid] < target) {
18              left = mid + 1; // Ищем в правой половине
19          }
20          else {
21              right = mid - 1; // Ищем в левой половине
22          }
23      }
24
25      return -1; // Элемент не найден
26  }
27
28  int main() {
29      setlocale(LC_ALL, "Russian");
30      // Важно: Массив должен быть отсортирован для бинарного поиска!
31      int numbers[] = { 1, 4, 13, 15, 21, 32 };
32      int size = sizeof(numbers) / sizeof(numbers[0]); // Вычисляем размер массива
33      int target = 4;
34
35      int index = binarySearch(numbers, size, target);
36
37      if (index != -1) {
38          cout << "Первый элемент " << target << " найден по индексу: " << index << endl;
39      }
40      else {
41          cout << "Элемент " << target << " не найден в массиве." << endl;
42      }
43
44      target = 35;
45      index = binarySearch(numbers, size, target);
46
47      if (index != -1) {
48          cout << "Первый элемент " << target << " найден по индексу: " << index << endl;
49      }
50      else {
51          cout << "Элемент " << target << " не найден в массиве." << endl;
52      }
53
54      return 0;
55  }
```

3.2 Результат работы

Первый элемент 4 найден по индексу: 1
Элемент 35 не найден в массиве.

3.3 Блок-схема





4 Интерполяционный поиск

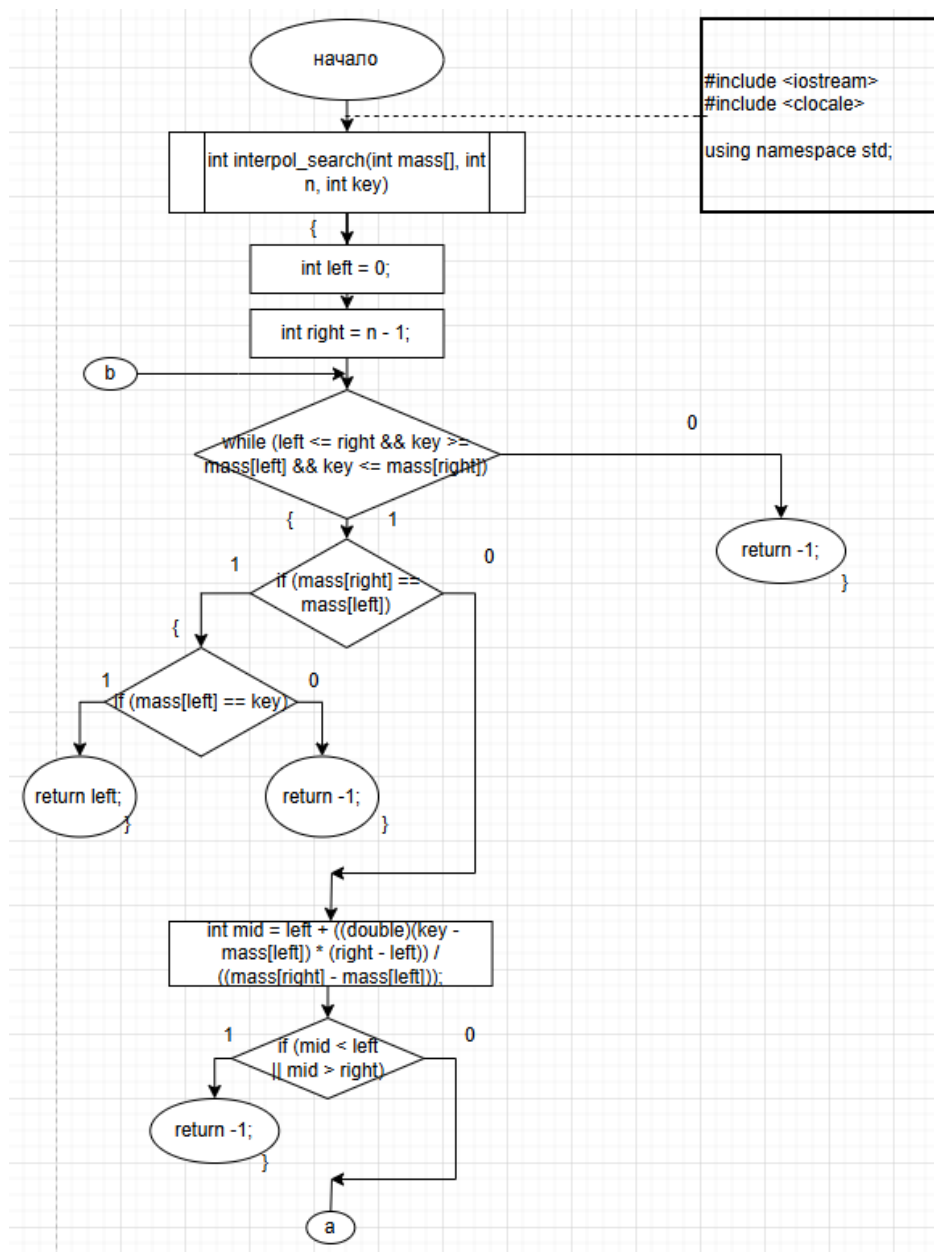
4.1 Код на C++

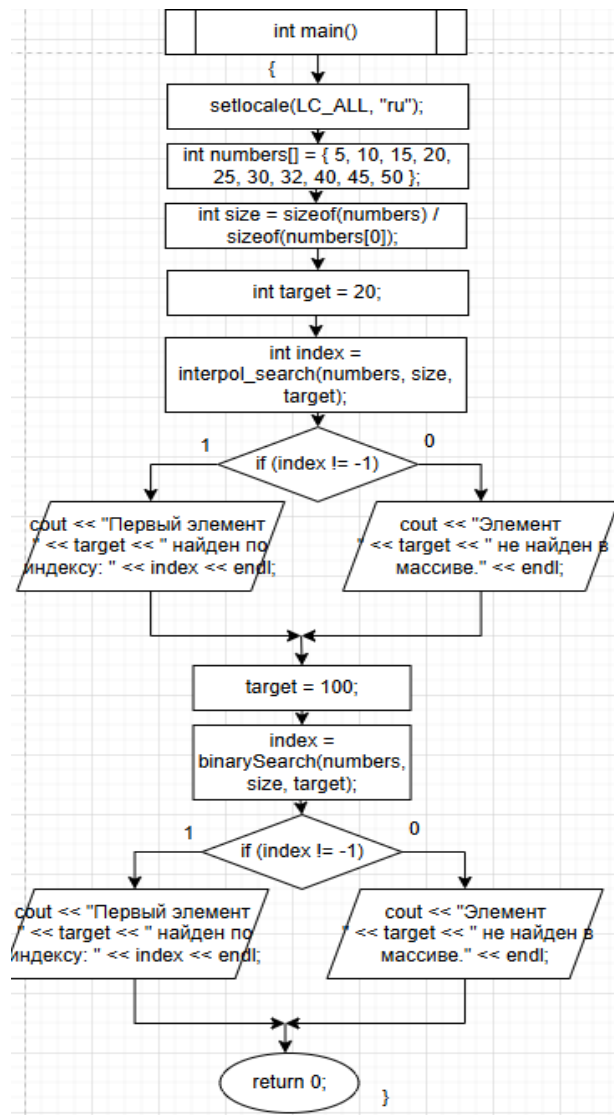
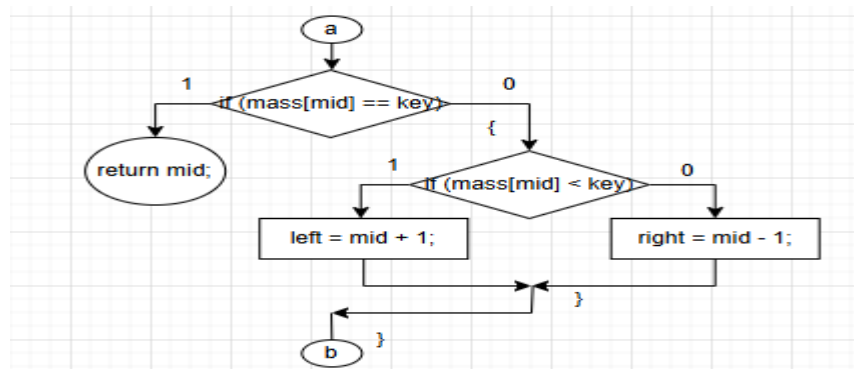
```
1  #include <iostream>
2  #include <locale>
3
4  using namespace std;
5
6  {}
7  int interpol_search(int mass[], int n, int key) {
8      int left = 0;
9      int right = n - 1;
10
11     while (left <= right && key >= mass[left] && key <= mass[right]) {
12         if (mass[right] == mass[left]) {
13             if (mass[left] == key) return left;
14             else return -1;
15         }
16
17         int mid = left + ((double)(key - mass[left]) * (right - left)) / ((mass[right] - mass[left]));
18
19         // Проверяем, что mid находится в пределах границ массива
20         if (mid < left || mid > right) {
21             return -1; // Индекс за пределами массива, элемент не найден
22         }
23
24         if (mass[mid] == key) {
25             return mid; // Если нашли key
26         }
27         else {
28             if (mass[mid] < key) {
29                 left = mid + 1; // Ищем в правой половине массива
30             }
31             else {
32                 right = mid - 1; // Ищем в левой половине массива
33             }
34         }
35     }
36
37     return -1; // Если key не найдем, возвращаем -1
38 }
39
40 int main() {
41     setlocale(LC_ALL, "Russian");
42
43     // Массив должен быть отсортирован для интерполяционного поиска
44     int numbers[] = { 5, 10, 15, 20, 25, 30, 32, 40, 45, 50 };
45     int size = sizeof(numbers) / sizeof(numbers[0]);
46     int target = 20;
47
48     int index = interpol_search(numbers, size, target);
49
50     if (index != -1) {
51         cout << "Первый элемент " << target << " найден по индексу: " << index << endl;
52     }
53     else {
54         cout << "Элемент " << target << " не найден в массиве." << endl;
55     }
56
57     target = 100; // Попробуем найти элемент, которого нет в массиве
58     index = interpol_search(numbers, size, target);
59
60     if (index != -1) {
61         cout << "Первый элемент " << target << " найден по индексу: " << index << endl;
62     }
63     else {
64         cout << "Элемент " << target << " не найден в массиве." << endl;
65     }
66
67     return 0;
68 }
```

4.2 Результат работы программы

Первый элемент 20 найден по индексу: 3
Элемент 100 не найден в массиве.

4.3 Блок-схема





5 Ссылка на github

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