

## Лабораторная работа № 1

**Тема:** «Алгоритмы поиска»

**Цель работы:** Изучение алгоритмов поиска

### Бинарный поиск ( $\text{mid} = (\text{left} + \text{right})/2$ )

В бинарном поиске алгоритм отслеживает наименьший и наибольший индексы элементов массива —  $\text{min}$  и  $\text{max}$ . Изначально они равны первому (нулевому) и последнему индексу соответственно. Затем алгоритм рассчитывает индекс, находящийся между ними, то есть  $\text{mid}$ . Если целевое значение меньше  $\text{mid}$ , алгоритм сбрасывает  $\text{max}$ , чтобы начать новый поиск в левой половине массива; если оно больше — сбрасывается  $\text{min}$  и новый поиск ведется в правой половине. Если же целевое значение равно  $\text{mid}$ , алгоритм возвращает его индекс.

**Линейный поиск**, циклически проходит по массиву, пытаясь отыскать целевой элемент.

В отличие от бинарного поиска, линейный работает не с массивами, а со связными списками, где нельзя запросто перепрыгнуть из одной части в другую. Списки могут быть и несортированными, но с отсортированными алгоритм экономит значительную часть времени: он останавливается, дойдя до элемента со значением, превышающим целевое, и не ведет дальнейший поиск того, чего заведомо не существует.

					АиСД 090000.000 ПР						
Изм	Лист	№ докум.	Подпись	Дата							
Разраб.		Климова Ю.В.			Лабораторная работа №1  «Алгоритмы поиска»			Лит	Лист	Листов	
Провер.		Береза А.Н.							1	8	
								ИСОиП(ф)ДГТУ ИСТ-Тб21			
Н.контр.											
Утв.											

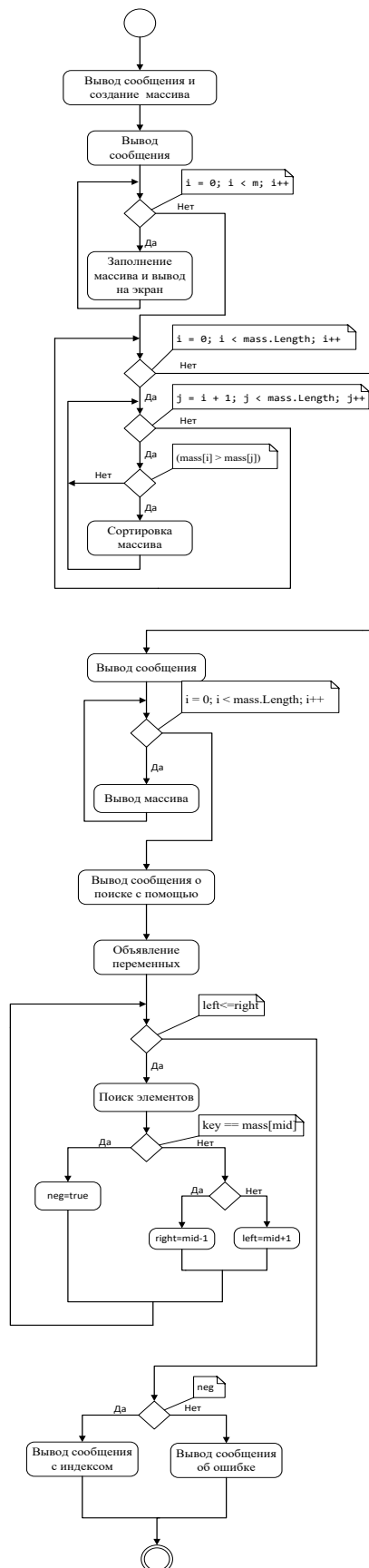


Рисунок 1-Блок-схема алгоритма «Бинарного поиска»

## Программный код алгоритма «Бинарного поиска»

```
#include <iostream>
#include <algorithm>

using namespace std;

int main() {
    setlocale(LC_ALL, "rus");

    int arr[10];
    int key;

    cout << "Введите 10 чисел для заполнения массива: " << endl;

    for (int i = 0; i < 10; i++) {
        cin >> arr[i];
    }

    sort (arr, arr + 10);

    cout << endl << "Введите ключ: ";

    cin >> key;

    bool flag = false;
    int l = 0;
    int r = 9;
    int mid;

    while ((l <= r) && (flag != true)) {
        mid = (l + r) / 2;

        if (arr[mid] == key) flag = true;
        if (arr[mid] > key) r = mid - 1; else l = mid + 1;
    }

    if (flag) cout << "Индекс элемента " << key << " в массиве равен: " << mid;
    else cout << "Извините, но такого элемента в массиве нет";

    system("pause");
    return 0;
}
```

Результат выполнения алгоритма :

```
Введите 10 чисел для заполнения массива:
10 54 74 34 95 37 45 64 28 94

Введите ключ: 95
sh: 1: pause: not found
Индекс элемента 95 в массиве равен: 9
```

					100000.000 ПР	Лист
						3
Изм	Лист	№ докум.	Подпись	Дата		

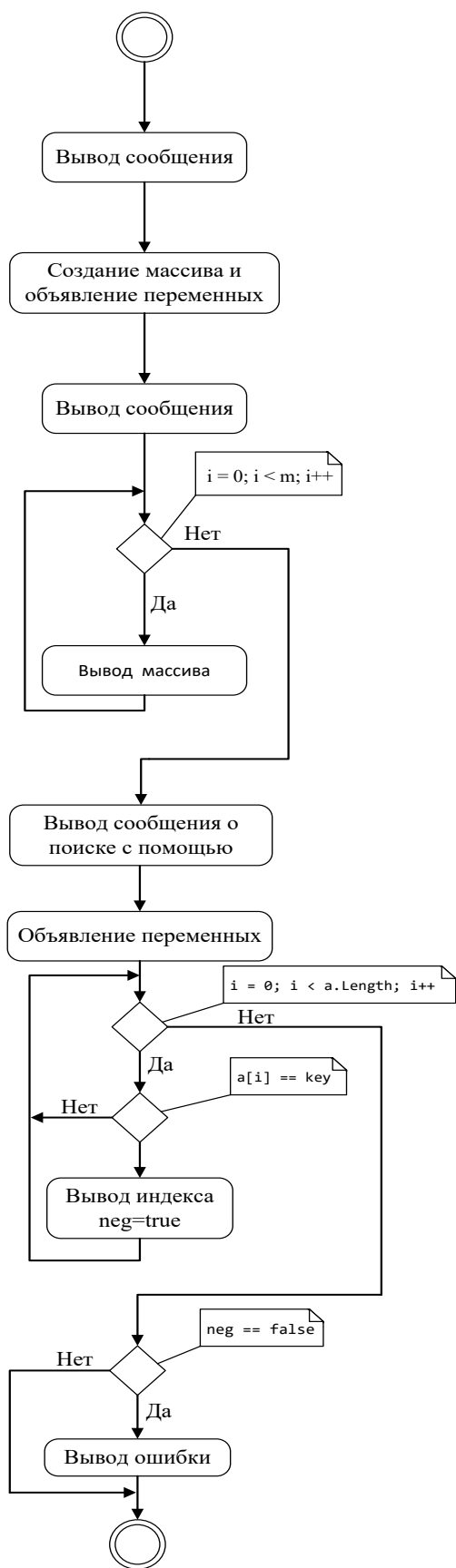


Рисунок 2- Блок-схема алгоритма «Линейного поиска»

## Программный код Линейный поиск

```
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;
int main() {
    setlocale(LC_ALL, "rus");

    int ans[10]; // создали массив для записи всех индексов
    int h = 0;
    int arr[10]; // создали массив на 20 элементов
    int key; // создали переменную в которой будет находиться ключ
    srand ( time(NULL) );

    for (int i = 0; i < 10; i++) {
        arr[i] = 1 + rand() % 10; // заполняем случайными числами ячейки

        cout << arr[i] << " "; // выводим все ячейки массива

        if (i == 9) {
            cout << endl;
        }
    }

    cout << endl << endl << "Введите ключ : "; cin >> key; // считываем ключ

    for (int i = 0; i < 10; i++) {
        if (arr[i] == key) { // проверяем равен ли arr[i] ключу
            ans[h++] = i;
        }
    }

    if (h != 0) { // проверяем были ли совпадения
        for (int i = 0; i < h; i++) {
            cout << "Ключ " << key << " находится в ячейке " << ans[i] << endl; //выводим все индексы
        }
    }
    else {
        cout << "Мы не нашли ключ " << key << " в массиве";
    }

    system("pause");
    return 0;
}
```

Результат выполнения алгоритма :

```
11 11 18 10 19 3 11 6 15 17
4 7 8 18 13 14 14 8 9 5

Введите ключ : 5
Ключ 5 находится в ячейке 19
sh: 1: pause: not found
```

					100000.000 ПР	Лист
						5
Изм	Лист	№ докум.	Подпись	Дата		