МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ПНИПУ)

Факультет: Электротехнический

Кафедра: «Информационные технологии и автоматизированные системы» (ИТАС)

Направление: Разработка информационных систем (РИС)

**О Т Ч Ё Т**   
**о лабораторной работе №1**

Дисциплина: «Основы алгоритмизации и программирования»

Тема: «Методы поиска элементов»

Выполнил:

Студент группы РИС-24-3б

Суяргулов А.О.

Проверил:

Доцент кафедры ИТАС

Полякова О.А.

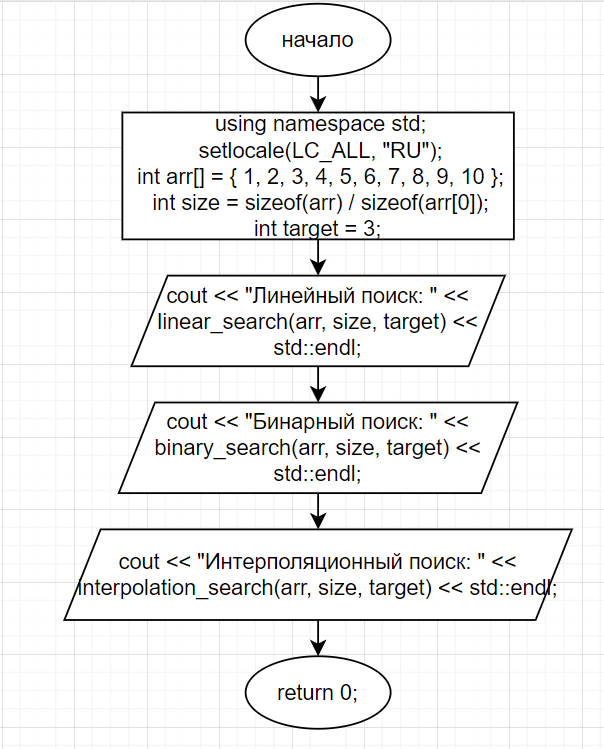
Пермь, 2025

ВВЕДЕНИЕ

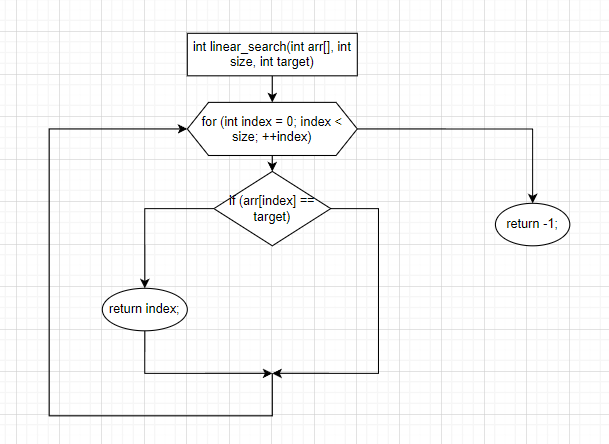
Реализовать методы поиска элемента в массиве(линейный бинарный интерполяционный).

ОСНОВНАЯ ЧАСТЬ

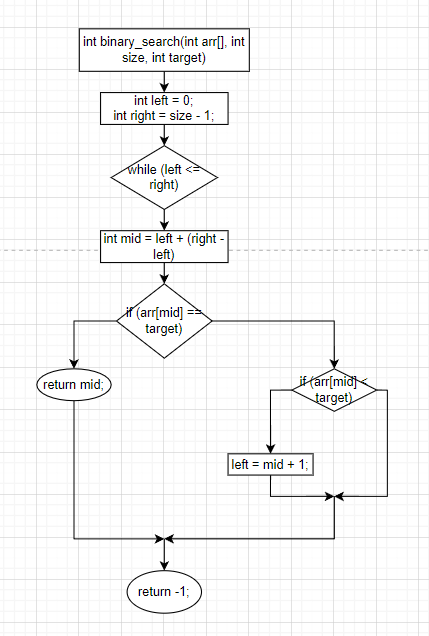
**Блок-схема:**

****

**Линейный поиск:**

****

**Бинарный поиск:**

****

**Код:**

#include <iostream>

int linear\_search(int arr[], int size, int target) {

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

if (arr[index] == target) {

return index; // Возвращаем индекс найденного элемента

}

}

return -1; // Если элемент не найден

}

int binary\_search(int arr[], int size, int target) {

int left = 0;

int right = size - 1;

while (left <= right) {

int mid = left + (right - left) / 2; // Находим средний индекс

if (arr[mid] == target) {

return mid; // Возвращаем индекс найденного элемента

}

else if (arr[mid] < target) {

left = mid + 1; // Ищем в правой половине

}

else {

right = mid - 1; // Ищем в левой половине

}

}

return -1; // Если элемент не найден

}

int interpolation\_search(int arr[], int size, int target) {

int low = 0;

int high = size - 1;

while (low <= high && target >= arr[low] && target <= arr[high]) {

if (low == high) {

if (arr[low] == target) {

return low; // Возвращаем индекс найденного элемента

}

return -1; // Если элемент не найден

}

// Вычисляем позицию

int pos = low + ((high - low) / (arr[high] - arr[low]) \* (target - arr[low]));

if (arr[pos] == target) {

return pos; // Возвращаем индекс найденного элемента

}

if (arr[pos] < target) {

low = pos + 1; // Ищем в правой половине

}

else {

high = pos - 1; // Ищем в левой половине

}

}

return -1; // Если элемент не найден

}

int main() {

using namespace std;

setlocale(LC\_ALL, "RU");

int arr[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

int size = sizeof(arr) / sizeof(arr[0]); // Определяем размер массива

int target = 3;

cout << "Линейный поиск: " << linear\_search(arr, size, target) << std::endl;

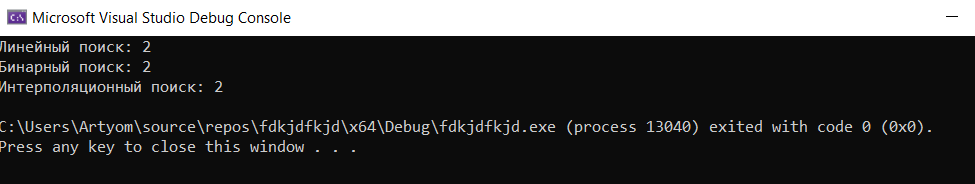
cout << "Бинарный поиск: " << binary\_search(arr, size, target) << std::endl;

cout << "Интерполяционный поиск: " << interpolation\_search(arr, size, target) << std::endl;

return 0;

}

**Результат работы кода:**

****

**Ссылка на GitHub:**

<https://github.com/Drazda1488/search>