Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

Кафедра вычислительной техники

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №3**

По дисциплине “Алгоритмы и структуры данных”

Вариант 21

|  |  |  |
| --- | --- | --- |
| Преподаватель | \_\_\_\_\_\_\_\_\_\_ | Матковский И.В. |
|  | подпись, дата |  |
| Студент КИ20-07Б, 032052463 | \_\_\_\_\_\_\_\_\_\_ | Базаров А.С. |
|  | подпись, дата |  |
|  |  |  |
|  |  |  |

Красноярск 2021

**СОДЕРЖАНИЕ**

[1 Задание 4](#_Toc81912682)

[2 Исходный код 4](#_Toc81912683)

[3 Результаты 5](#_Toc81912684)

1. Задание

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Задание | Тип поиска | Содержимое массива | Массив отсортирован? |
| 21 | Найти минимальный из элементов больше заданного | Интерполирующий | Целые числа | Да |

1. Исходный код

Программа с интерполирующим поиском (к программе прикреплен файл с массивом чисел):

#include <stdio.h>

#include <iostream>

#include <fstream>

using namespace std;

// Если x присутствует в arr [0..n-1], то возвращает

// его индекс, иначе возвращает -1.

int interpolationSearch(int arr[], int lo, int hi, int x)

{

int mid;

// Поскольку массив отсортирован, элемент присутствует

// в массиве должен находиться в диапазоне, определяемом углом

if (lo < hi) {

// Исследование позиции с сохранением

// с учетом равномерного распределения.

mid = lo + (hi - lo) / 2;

// Состояние цели найдено

if (arr[mid] == x)

return mid + 1;

// Если x больше, x находится в правом подмассиве

if (arr[mid] < x)

if (arr[mid + 1] > x) return mid + 1;

else return interpolationSearch(arr, mid + 1, hi, x);

// Если x меньше, x находится в левом подмассиве

if (arr[mid] > x) {

if (arr[mid - 1] <= x)

return mid;

if (arr[mid - 1] > x) return interpolationSearch(arr, lo, mid - 1, x);

}

}

return -1;

}

int main()

{

setlocale(LC\_ALL, "rus");

fstream in;

in.open("S\_Array.txt");

int n = 0;

in >> n;

int\* arr = new int[n];

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

{

in >> arr[i];

}

cout << "Массив из " << n << " элементов";

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

{

cout << arr[i] << " ";

}

cout << endl;

// Element to be searched

int x;

cout << "Введите элемент, больше которого надо найти:";

cin >> x;

if (x > arr[n - 1]) {

cout << "Число больше самого максимального числа!";

exit(1);

}

if (x == arr[n - 1]) {

cout << "Число равно самому максимальному числу!";

exit(1);

}

int index = interpolationSearch(arr, 0, n - 1, x);

// If element was found

if (index != -1)

cout << "Наименьший элемент больше заданного - " << arr[index];

return 0;

}

1. Теоретические оценки сложности

Теоретическая сложность данного алгоритма составляет составляет , (где n – массив чисел) так как мы использовали интерполирующий поиск с условием.

1. Результаты

Результаты выполнения алгоритма приведены в таблице 1

Таблица 1

|  |  |
| --- | --- |
| Размер элемента | Простой алгоритм, время в мс. |
| 10 | 0.0011 |
| 1000 | 0,1001 |
| 100000 | 10.0001 |