Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

Кафедра вычислительной техники

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №5**

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

|  |  |  |
| --- | --- | --- |
| № | Первый метод | Второй метод |
| 21 | Сортировка вставками | Сортировка кучей |

1. Исходный код

Программа с двумя методами сортировки (к программе прикреплен файл с массивом чисел):

#include <clocale>

#include <iostream>

using namespace std;

void printArray(int arr[], int n)

{

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

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

cout << endl;

}

void InsertionSort(int arr[], int n)

{

int temp; // временная переменная для хранения значения элемента сортируемого массива

int item; // индекс предыдущего элемента

for (int i = 1; i < n; i++)

{

temp = arr[i]; // инициализируем временную переменную текущим значением элемента массива

item = i - 1; // запоминаем индекс предыдущего элемента массива

while (item >= 0 && arr[item] > temp) // пока индекс не равен 0 и предыдущий элемент массива больше текущего

{

// перестановка элементов массива

arr[item + 1] = arr[item];

arr[item] = temp;

item--;

}

}

}

// Процедура для преобразования в двоичную кучу поддерева с корневым узлом i, что является

// индексом в arr[]. n - размер кучи

void heapify(int arr[], int n, int i)

{

int largest = i;

// Инициализируем наибольший элемент как корень

int l = 2 \* i + 1; // левый = 2\*i + 1

int r = 2 \* i + 2; // правый = 2\*i + 2

// Если левый дочерний элемент больше корня

if (l < n && arr[l] > arr[largest])

largest = l;

// Если правый дочерний элемент больше, чем самый большой элемент на данный момент

if (r < n && arr[r] > arr[largest])

largest = r;

// Если самый большой элемент не корень

if (largest != i)

{

swap(arr[i], arr[largest]);

// Рекурсивно преобразуем в двоичную кучу затронутое поддерево

heapify(arr, n, largest);

}

}

// Основная функция, выполняющая пирамидальную сортировку

void heapSort(int arr[], int n)

{

// Построение кучи (перегруппируем массив)

for (int i = n / 2 - 1; i >= 0; i--)

heapify(arr, n, i);

// Один за другим извлекаем элементы из кучи

for (int i = n - 1; i >= 0; i--)

{

// Перемещаем текущий корень в конец

swap(arr[0], arr[i]);

// вызываем процедуру heapify на уменьшенной куче

heapify(arr, i, 0);

}

}

int main()

{

setlocale(LC\_ALL, "rus");

int n, choose;

cout << "Введите размер массива:";

cin >> n;

int \*arr = new int[n];

cout << "Введите " << n << " элементов массива" << endl;

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

cin >> arr[i];

}

cout << "Какой способ сортировки вы выберете?" << endl;

cout << "1 - сортировка вставками, 2 - пирамидальная сортировка" << endl;

Label:

cin >> choose;

if (choose == 1)

InsertionSort(arr, n);

else if (choose == 2) heapSort(arr, n);

else {

cout << "Выберите способ сортировки!" << endl;

goto Label;

}

cout << "Отсортированный массив" << endl;

printArray(arr, n);

return 0;

}

1. Теоретические оценки сложности

Теоретическая сложность программы при использовании сортировки вставками составляет , (где n – массив чисел) так как мы считываем массив с консоли, сортируем его и выводим.

Теоретическая сложность программы при использовании сортировки кучей составляет , (где n – массив чисел) так как мы считываем массив с консоли, сортируем его и выводим.

1. Результаты

Результаты выполнения алгоритма приведены в таблице 1

Таблица 1

|  |  |  |
| --- | --- | --- |
| Количество элементов | Сортировка вставками, время в мс. | Сортировка кучей, время в мс. |
| 10 | 0,12 | 0,043 |
| 1000 | 1002 | 8,91 |
| 10000 | 100020 | 112,1 |