МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Методы сортировки

(наименование темы проекта или работы)

ОТЧЕТ

к лабораторной работе №4

по дисциплине

Технологии программирования

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Багиров М.Б.

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Толстова М.Г.

(подпись) (фамилия, и.,о.)

18-ИСТ-4

(шифр группы)

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород 2020

Вариант 19

Задание:

Реализовать пирамидальную сортировку. Листинг

Laba5.cpp

#include "stdafx.h"

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

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

// индексом в arr[]. size - размер кучи

void heapify(int \*arr, int size, 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 < size && arr[l] > arr[largest])

largest = l;

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

if (r < size && 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, size, largest);

}

}

// Основная функция, выполняющая пирамидальную сортировку

void heapSort(int \*arr, int size)

{

// Построение кучи (перегруппируем массив)

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

heapify(arr, size, i);

// Один за другим извлекаем элементы из кучи

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

{

// Перемещаем текущий корень в конец

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

// вызываем процедуру heapify на уменьшенной куче

heapify(arr, i, 0);

}

}

/\* Вспомогательная функция для вывода на экран массива размера n\*/

void printArray(int \*arr, int size)

{

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

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

cout << "\n";

}

int main()

{

int size;

cout << "Number of elements: ";

cin >> size;

int \*arr = new int[size];

srand(time(0));

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

{

arr[i] = rand() % 100;

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

}

heapSort(arr, size);

cout << endl;

cout << "Sorted array is \n";

printArray(arr, size);

}