# Учреждение образования

# «Белорусский государственный технологический университет»

Кафедра информационных систем и технологий

Отчёт по лабораторной работе №17

Вариант 2

Выполнил:

студент 1 курса 2 группы Самцевич Алексей

Минск 2020

Задание №1. Шейкерная сортировка, пирамидальная сортировка, сортировка слиянием

Код:

//

// main.cpp

// laba\_17.1

//

// Created by Alex Samtsevich on 09.06.2020.

// Copyright © 2020 Alex Samtsevich. All rights reserved.

//

#include <ctime>

#include <stdlib.h>

#include <iostream>

using namespace std;

// sheykernaya

void bubbleSort(int \*a, int n)

{

for (int i = 0; i < n/2; i++) {

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

if (a[j] > a[j + 1]) {

int t = a[j];

a[j] = a[j + 1];

a[j + 1] = t;

}

for(int j = n-2-i; j>i; j--)

if (a[j] < a[j - 1]) {

int t = a[j];

a[j] = a[j - 1];

a[j - 1] = t;

}

}

}

// piramidalnaya

void heapify(int \*A, int pos, int n)

{

int t, tm;

while (2 \* pos + 1 < n)

{

t = 2 \* pos + 1;

if (2 \* pos + 2 < n && A[2 \* pos + 2] >= A[t])

t = 2 \* pos + 2;

if (A[pos] < A[t])

{

tm = A[pos];

A[pos] = A[t];

A[t] = tm;

pos = t;

}

else break;

}

}

void piramSort(int \*A, int n)

{

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

heapify(A, i, n);

while (n > 0)

{

int tm = A[0];

A[0] = A[n - 1];

A[n - 1] = tm;

n--;

heapify(A, 0, n);

}

}

//sliyanie

void insOrd(int \*m, int sm, int em, int e)

{ // m[]  массив чисел; sm  индекс 1-ого элемента левой части; cm  индекс

//последн. элемента левой части; em  индекс последн. элемента правой части

int buf;

int i = sm;

while (i <= em && m[i] < e)

{

if (i - 1 >= sm)

m[i - 1] = m[i];

i++;

} m[i - 1] = e;

}

int\* merge(int \*m, int sm, int cm, int em)

{

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

{

if (m[i] > m[cm + 1])

{

int buf = m[i];

m[i] = m[cm + 1];

insOrd(m, cm + 1, em, buf);

}

}

return m;

}

int\* sortMerge(int \*m, int lm, int sm = 0)

{

if (lm > 1)

{

sortMerge(m, lm / 2, sm);

sortMerge(m, lm - lm / 2, sm + lm / 2);

merge(m, sm, sm + lm / 2 - 1, sm + lm - 1);

};

return m;

}

int main() {

int\* a = new int[1000000];

for (int i = 0; i < 1000000; i++)

a[i] = rand() % 1000;

for (int n = 1000; n <= 1000000; n += 1000) {// 10 <11

int\* a1 = new int[n];

int\* a2 = new int[n];

int\* a3 = new int[n];

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

a1[i] = a2[i] = a3[i] = a[i];

clock\_t b1 = clock();

bubbleSort(a1, n);

clock\_t e1 = clock();

clock\_t b2 = clock();

piramSort(a2, n);

clock\_t e2 = clock();

clock\_t b3 = clock();

sortMerge(a3, n);

clock\_t e3 = clock();

cout << n << " " << ((double)(e1 - b1) / CLOCKS\_PER\_SEC) <<

" " << ((double)(e2 - b2) / CLOCKS\_PER\_SEC)<<

" " << ((double)(e3 - b3) / CLOCKS\_PER\_SEC) <<endl;

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

// cout << a1[i] << " " << a2[i] << " " << a3[i] << endl;

//}

delete[]a1;

delete[]a2;

delete[]a3;

}

delete[]a;

return 0;

}

Скриншот:



