Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №5**

**«ИССЛЕДОВАНИЕ АЛГОРИТМОВ СОРТИРОВКИ»**

**ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

Выполнил: студент учебной группы

ИСПк-204-52-00

Сокованов Игорь Николаевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

1. Цель работы: получить базовые сведения о наиболее известных алгоритмах сортировки, изучить принципы работы с текстовыми.
2. Задание:

1)Реализовать сортировку данных с помощью алгоритма выбором.

2)Реализовать сортировку данных с помощью алгоритма слияния.

3)В обоих случаях необходимо предусмотреть возможность изменения компаратора (реализация компаратора необходимо передавать в подпрограмму функции).

4)Считывание и вывод данных необходимо производить из текстового файла.

5)Для демонстрации работы программных реализаций самостоятельно подготовить варианты работы выходных данных (при этом объем тестовых файлов должен позволять оценить скорость работы программ).

1. Код программы:

**uses** crt, System;

**var**

in1text: **file of** char;

in2text: **file of** char;

out1text, out2text: text;

arr1, arr2: **array**[0..10000] **of** char; //Сам массив

n1, n2, star,stop,star1,stop1: integer; //Размер массива

//Печать Гошандрийского Фонда

//Начало сортировки выборки

**function** sortVIBot: integer;

**var**

i, j, min: integer;

buf: char;

**begin**

reset(in1text);

**for** i := 1 **to** n2 **do**

read(in1text, arr1[i]);

**for** i := 1 **to** n2 **do**

**begin**

min := i;

**for** j := i + 1 **to** n2 **do**

**if** arr1[j] < arr1[min] **then**

min := j;

buf := arr1[i];

arr1[i] := arr1[min];

arr1[min] := buf;

**end**;

writeln('Сортировка с помощью выборки:');

**for** i := 1 **to** n2 **do**

**begin**

write(arr1[i], ' ');

**end**;

**end**;

**function** sortVIBdo: integer;

**var**

i, j, min: integer;

buf: char;

**begin**

reset(in1text);

**for** i := 1 **to** n2 **do**

read(in1text, arr1[i]);

**for** i := 1 **to** n2 **do**

**begin**

min := i;

**for** j := i + 1 **to** n2 **do**

**if** arr1[j] > arr1[min] **then**

min := j;

buf := arr1[i];

arr1[i] := arr1[min];

arr1[min] := buf;

**end**;

writeln('Сортировка с помощью выборки:');

**for** i := 1 **to** n2 **do**

**begin**

print(arr1[i]);

**end**;

**end**;

//Конец сортировки выборки

//Начало сортировки слияния

**procedure** Slivot(**var** p, q: integer);

**var**

r, i, j, k: integer;

b: **array**[1..10000] **of** char;

**begin**

r := (p + q) **div** 2;

i := p;

j := r + 1;

**for** k := p **to** q **do**

**if** (i <= r) **and** ((j > q) **or** (arr2[i] < arr2[j])) **then**

**begin**

b[k] := arr2[i];

i := i + 1;

**end**

**else**

**begin**

b[k] := arr2[j];

j := j + 1;

**end**;

**for** k := p **to** q **do**

arr2[k] := b[k];

**end**;

**procedure** SortSLIot(**var** p, q: integer);

**begin**

**if** p < q **then**

**begin**

**var** ss, sss: integer;

ss := (p + q) **div** 2;

SortSLIot(p, ss);

sss := (p + q) **div** 2 + 1;

SortSLIot(sss, q);

Slivot(p, q);

**end**;

**end**;

**procedure** Slivdo(**var** p, q: integer);

**var**

r, i, j, k: integer;

b: **array**[1..10000] **of** char;

**begin**

r := (p + q) **div** 2;

i := p;

j := r + 1;

**for** k := p **to** q **do**

**if** (i <= r) **and** ((j > q) **or** (arr2[i] > arr2[j])) **then**

**begin**

b[k] := arr2[i];

i := i + 1;

**end**

**else**

**begin**

b[k] := arr2[j];

j := j + 1;

**end**;

**for** k := p **to** q **do**

arr2[k] := b[k];

**end**;

**procedure** SortSLIdo(**var** p, q: integer);

**begin**

**if** p < q **then**

**begin**

**var** ss, sss: integer;

ss := ((p + q) **div** 2);

SortSLIdo(p, ss);

sss := (p + q) **div** 2 + 1;

SortSLIdo(sss, q);

Slivdo(p, q);

**end**;

**end**;

//Конец сортировки слияния

**function** kopr: integer;

**begin**

ClrScr;

writeln('Хотители изменить компоратор сортировок');

writeln('1-Сортировка по возрастанию выбором');

writeln('2-Сортировка по убыванию выбором');

writeln('3-Сортировка по возрастанию слиянием');

writeln('4-Сортировка по убыванию слиянием');

writeln('0-Выход');

**var** a: integer;

read(a);

**case** a **of**

1:

**begin**

star1:=DateTime.Now.Second;

star:=DateTime.Now.MilliSecond;

sortVIBot;

stop1:=DateTime.Now.Second;

stop:=DateTime.Now.MilliSecond;

writeln('Программа выполнялась ',stop1-star1,' секунд ',abs(stop-star),' милисекунду');

readln;

readln;

kopr;

**end**;

2:

**begin**

star1:=DateTime.Now.Second;

star:=DateTime.Now.MilliSecond;

sortVIBdo;

stop:=DateTime.Now.Second;

stop1:=DateTime.Now.Second;

writeln('Программа выполнялась ',stop1-star1,' секунд ',abs(stop-star),' милисекунду');

readln;

readln;

kopr;

**end**;

3:

**begin**

star1:=DateTime.Now.Second;

star:=DateTime.Now.MilliSecond;

n1 := 1;

reset(in2text);

**for var** i := 1 **to** n2 **do**

**begin**

read(in2text, arr2[i]);

**end**;

sortSLIot(n1, n2);

writeln('Сортировка с помощью слияния:');

**for var** i := 1 **to** n2 **do**

**begin**

write(arr2[i], ' ');

**end**;

stop1:=DateTime.Now.Second;

stop:=DateTime.Now.MilliSecond;

writeln('Программа выполнялась ',stop1-star1,' секунд ',abs(stop-star),' милисекунду');

readln;

readln;

kopr;

**end**;

4:

**begin**

star1:=DateTime.Now.Second;

star:=DateTime.Now.MilliSecond;

n1 := 1;

reset(in2text);

**for var** i := 1 **to** n2 **do**

**begin**

read(in2text, arr2[i]);

**end**;

sortSLIdo(n1, n2);

writeln('Сортировка с помощью слияния:');

**for var** i := 1 **to** n2 **do**

**begin**

write(arr2[i], ' ');

**end**;

stop1:=DateTime.Now.Second;

stop:=DateTime.Now.MilliSecond;

writeln('Программа выполнялась ',stop1-star1,' секунд ',abs(stop-star),' милисекунду');

readln;

readln;

kopr;

**end**;

**end**;

**end**;

**function** slove: integer;

**begin**

ClrScr;

write('Введите размер массива ');

readln(n2);

Assign(in1text, '1text.txt');

Assign(in2text, '2text.txt');

Assign(out1text, '1otvet.txt');

Assign(out2text, '2otvet.txt');

rewrite(in1text);

**var** a: char;

writeln('Содержимое первого массива(файла)');

**for var** i := 1 **to** n2 **do**

**begin**

readln(a);

write(in1text, a);

**end**;

rewrite(in2text);

writeln('Содержимое второго массива(файла)');

**for var** i := 1 **to** n2 **do**

**begin**

readln(a);

write(in2text, a);

**end**;

kopr;

rewrite(out1text);

writeln(out1text, 'Сортировка с помощью выборки:');

**for var** i := 1 **to** n2 **do**

**begin**

write(out1text, arr1[i]);

write(out1text, ' ');

**end**;

rewrite(out2text);

writeln(out2text, 'Сортировка с помощью слияния:');

**for var** i := 1 **to** n2 **do**

**begin**

write(out2text, arr2[i]);

write(out2text, ' ');

**end**;

**end**;

**begin**

slove;

close(in1text);

close(in2text);

close(out1text);

close(out2text);

**end**.

1. Результат выполнения программы:

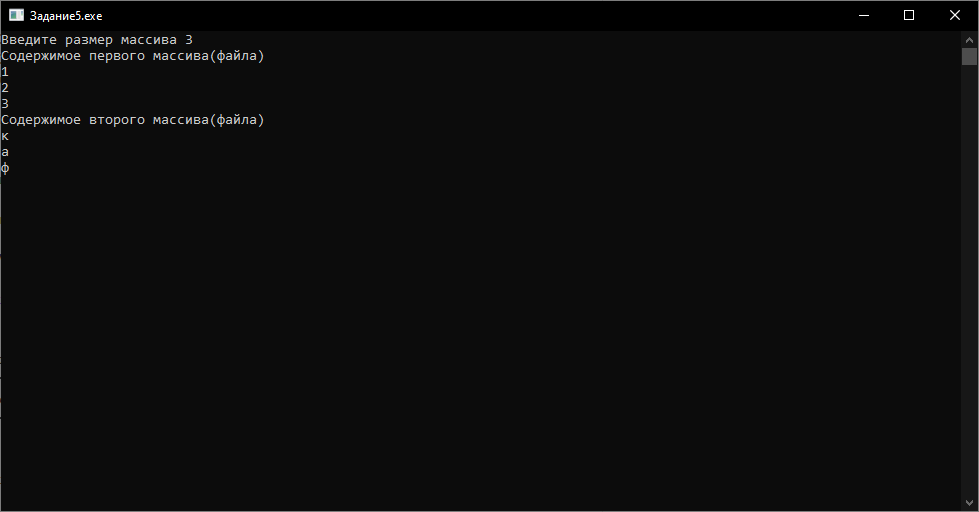


Рисунок 1- Результаты выполнения программы (1/7)

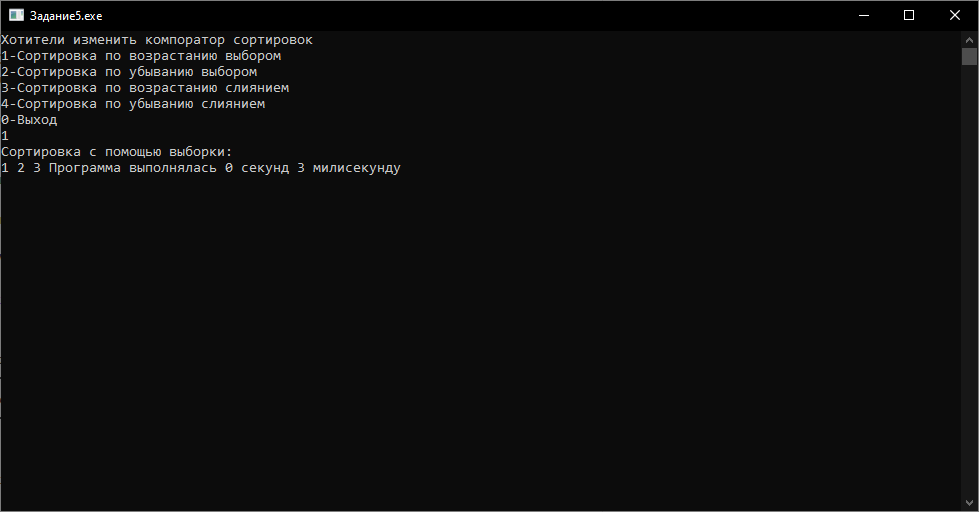


Рисунок 2- Результаты выполнения программы (2/7)

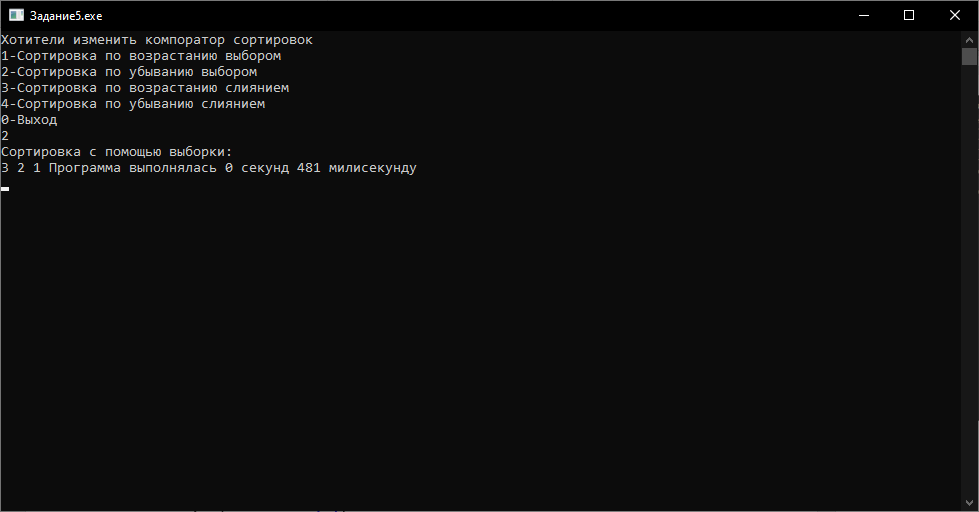


Рисунок 3- Результаты выполнения программы (3/7)

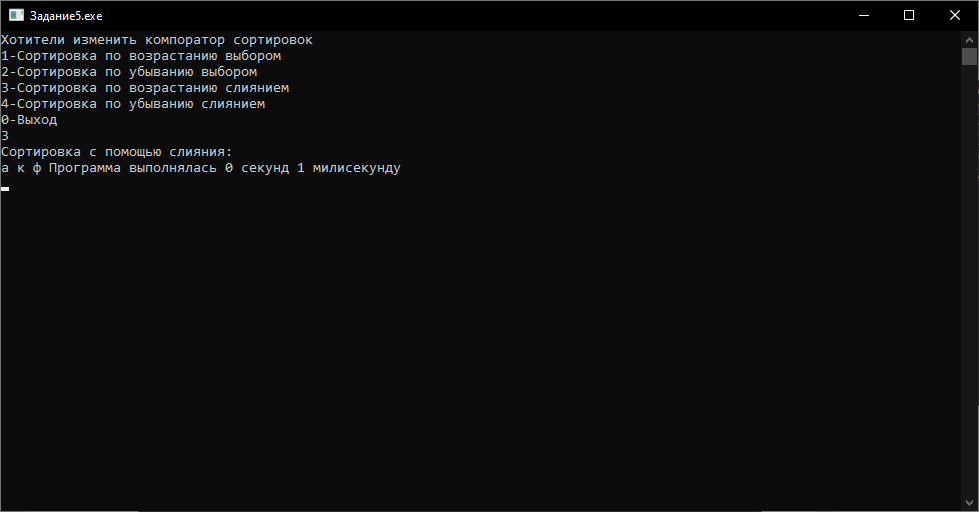


Рисунок 4- Результаты выполнения программы (4/7)

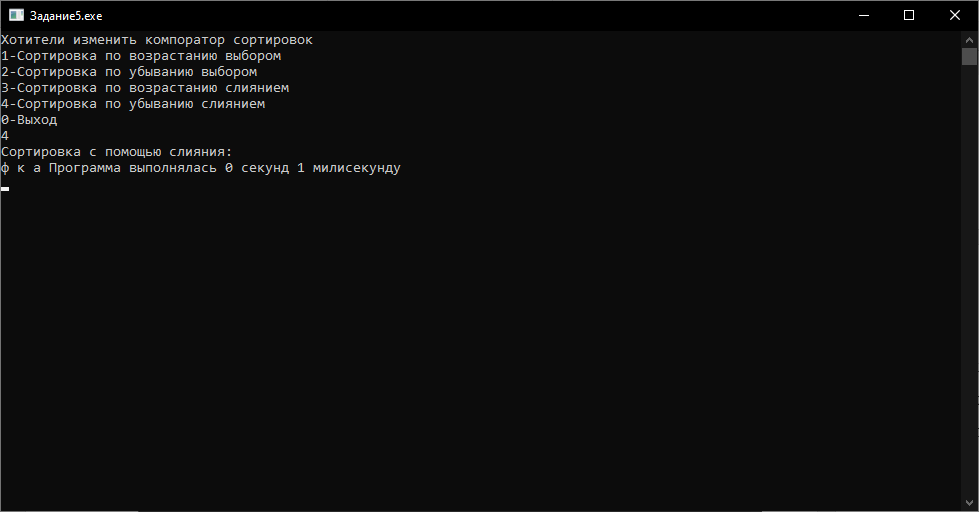


Рисунок 5- Результаты выполнения программы (5/7)

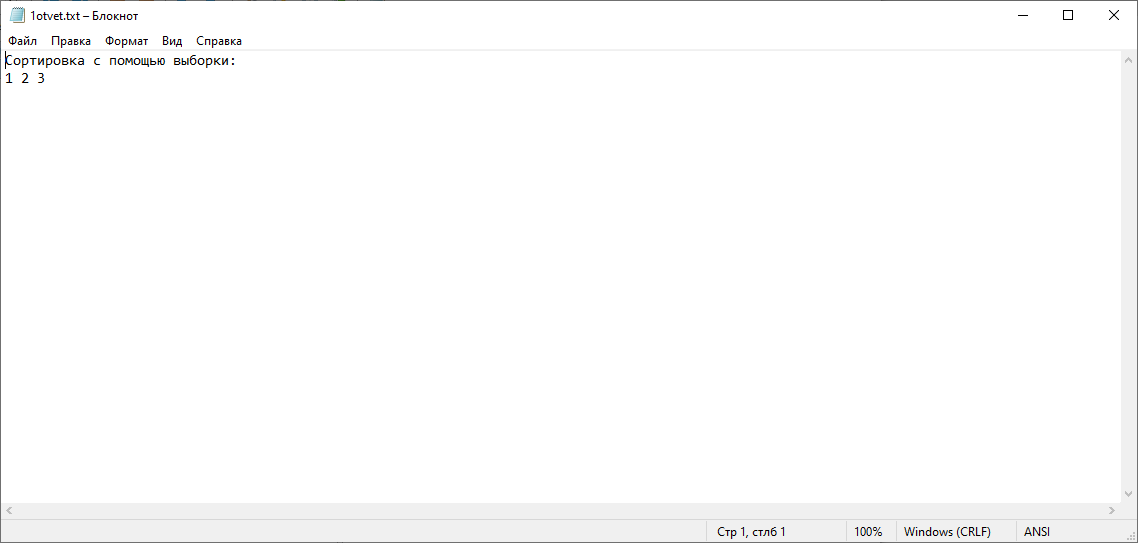


Рисунок 6- Результаты выполнения программы (6/7)

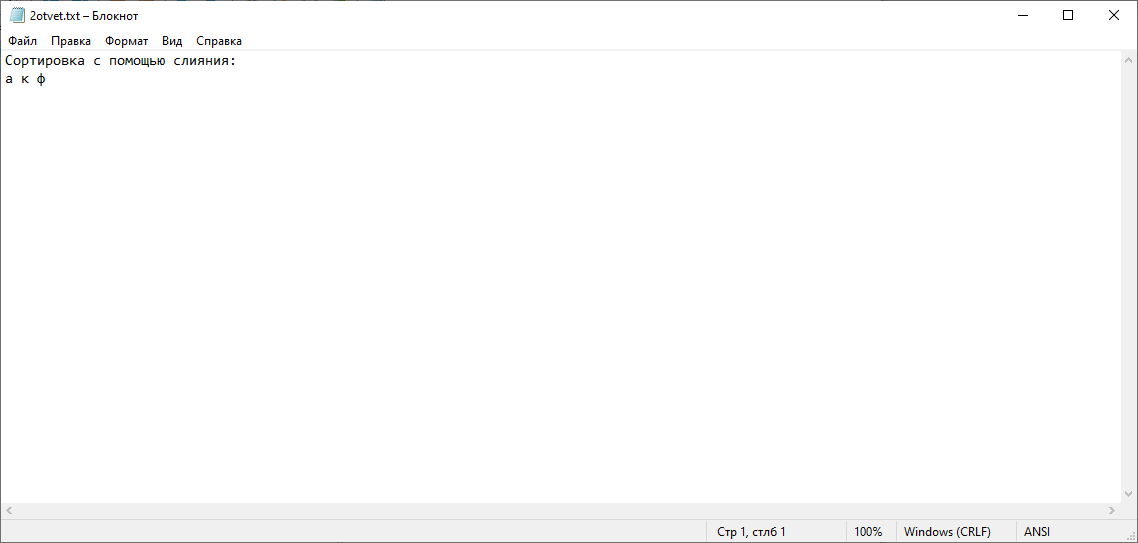


Рисунок 7- Результаты выполнения программы (7/7)

1. Вывод:

В этой лабораторной работе я узнал, как как составлять сортировку с помощью выбора и слияния. Узнал, что в паскале существует модуль System который мне пригодился для вычисления времени работы сортировок, но из-за этого модуля перестала работать команда random.

Но возникли трудности с:

С сортировки массива с помощью выборки. По неизвестным причинам программа не хотела выполнятся так как я хочу. Немного посидев в коде не заметил ошибок. Тогда принялся изучать файлы, в которые заносились данные и было замечено что перед первым символом стоял «Enter» из-за него программа работала не корректно. После этого вновь решил проверить код и было замечено что перед заполнением файла стоял не readln а read. Одна неправильная команда сломала весь код.

С нахождением команды секундомера которая мне нужна была для замирения времени работы сортировок.

С построением компаратора так как вообще не представлял, как это сделать из-за того, что не знал, что это слово обозначает.