Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №5**

**«ИССЛЕДОВАНИЕ АЛГОРИТМОВ СОРТИРОВКИ»**

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

Выполнил: студент учебной группы

ИСПк-205-52-00

Митягин Даниил Сергеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

**1.ЦЕЛЬ ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЫ**

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

**2.ФОРМУЛИРОВКА ЗАДАНИЯ**

Задание и решение делится на несколько этапов:

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

**3.КОД ПРОГРАММ**

**uses** crt, System;

**var**

in1text: **file of** char;

in2text: **file of** char;

out1text, out2text: text;

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

n1, n2, star,stop,star1,stop1: integer;

**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 алгоритмом 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 алгоритмом выборки:');

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

**begin**

print(arr1[i]);

**end**;

**end**;

**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**;

**function** kopr: integer;

**begin**

ClrScr;

writeln('Хотители изменить компоратор сортировок');

writeln('1-Сортировка по возрастанию алгоритмом выбора');

writeln('2-Сортировка по убыванию алгоритмом выбора');

writeln('3-Сортировка по возрастанию алгоритмом слияния');

writeln('4-Сортировка по убыванию алгоритмом слияния');

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('Время работы программы: ',stop1-star1,' сек ',abs(stop-star),' мсек');

readln;

readln;

kopr;

**end**;

2:

**begin**

star1:=DateTime.Now.Second;

star:=DateTime.Now.MilliSecond;

sortVIBdo;

stop:=DateTime.Now.Second;

stop1:=DateTime.Now.Second;

writeln('Время работы программы: ',stop1-star1,' сек ',abs(stop-star),' мсек');

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 алгоритмом слияния:');

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

**begin**

write(arr2[i], ' ');

**end**;

stop1:=DateTime.Now.Second;

stop:=DateTime.Now.MilliSecond;

writeln('Время работы программы: ',stop1-star1,' сек ',abs(stop-star),' мсек');

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 алгоритмом слияния слияния:');

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

**begin**

write(arr2[i], ' ');

**end**;

stop1:=DateTime.Now.Second;

stop:=DateTime.Now.MilliSecond;

writeln('Время работы программы: ',stop1-star1,' сек ',abs(stop-star),' мсек');

readln;

readln;

kopr;

**end**;

**end**;

**end**;

**function** slove: integer;

**begin**

ClrScr;

write('Введите размер массива - ');

readln(n2);

Assign(in1text, 'D:\LR 14\DKR5textin.1.txt');

Assign(in2text, 'D:\LR 14\DKR5textin.2.txt');

Assign(out1text,'D:\LR 14\DKR5textout.1.txt');

Assign(out2text,'D:\LR 14\DKR5textout.2.txt');

rewrite(in1text);

**var** a: char;

writeln('Первый массив - ');

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

**begin**

readln(a);

write(in1text, a);

**end**;

rewrite(in2text);

writeln('Второй массив - ');

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

**begin**

readln(a);

write(in2text, a);

**end**;

kopr;

rewrite(out1text);

writeln(out1text, 'Сортировка алгоритмом выборки:');

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

**begin**

write(out1text, arr1[i]);

write(out1text, ' ');

**end**;

rewrite(out2text);

writeln(out2text, 'Сортировка алгоритмом 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 – Вывод программы

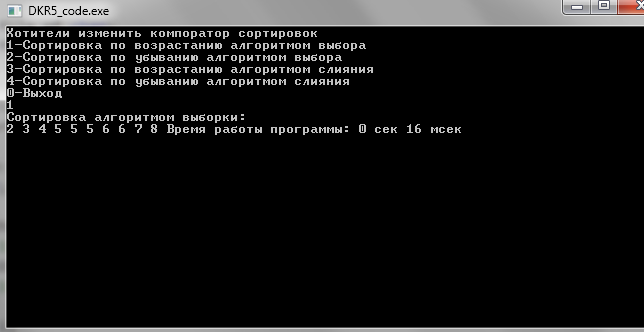


Рисунок 2 – Вывод программы

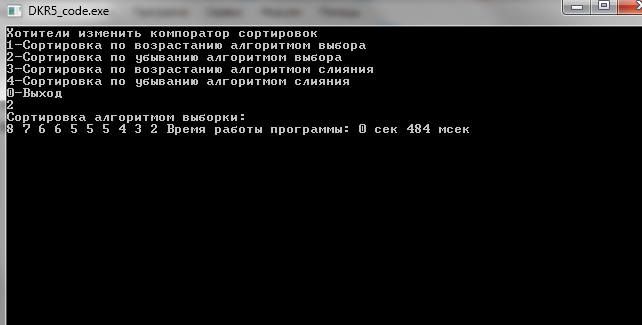


Рисунок 3 – Вывод программы

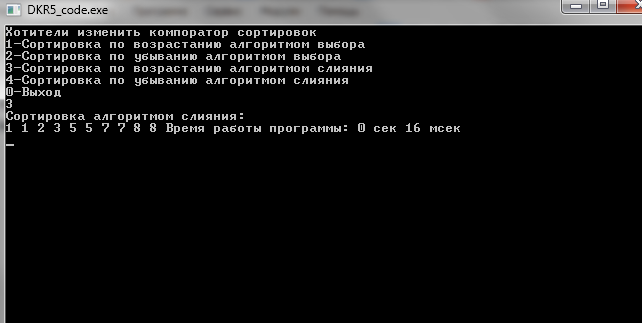


Рисунок 4 – Вывод программы

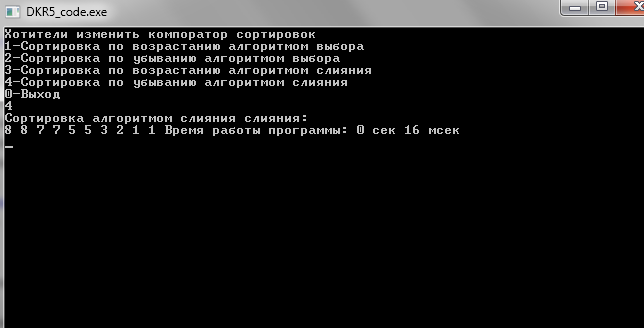


Рисунок 5 – Вывод программы

**6. ВЫВОД**

На данной домашней контрольной работе были получены базовые сведения о наиболее известных алгоритмах сортировки, изучены принципы работы с текстовыми файлами языка программирования Pascal. Была создана программа сортирующая двумя алгоритмами. Алгоритмом выбора и алгоритмом слияния.

Были использованы подпрограммы (функции и процедуры) для вычисления значений функции, данной в задачи, и выполнения основной части программы. Также были применены функции библиотеки CRT для организации взаимодействия с пользователем посредством case-меню

После того как мы создали программу и схему нам поручили написать отчёт, с отчётом трудностей возникнуть у нас не должно, самое главное оформить все по плану.