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-204-52-00

Максимов Антон Николаевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

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

2. Задание:

Вариант: 12.

1. Реализовать сортировку данных с помощью алгоритма выбором.
2. Реализовать сортировку данных с помощью быстрого алгоритма.
3. В обоих случаях необходимо предусмотреть возможность изменения компаратора (реализация компаратора в виде передаваемой в подпрограмму функции).
4. Считывание и вывод данных необходимо производить из текстового файла.
5. Для демонстрации работы программных реализация самостоятельно подготовить варианты входных данных (при этом объем тестовых файлов должен позволять оценить скорость работы программ).

3. Схема алгоритма:

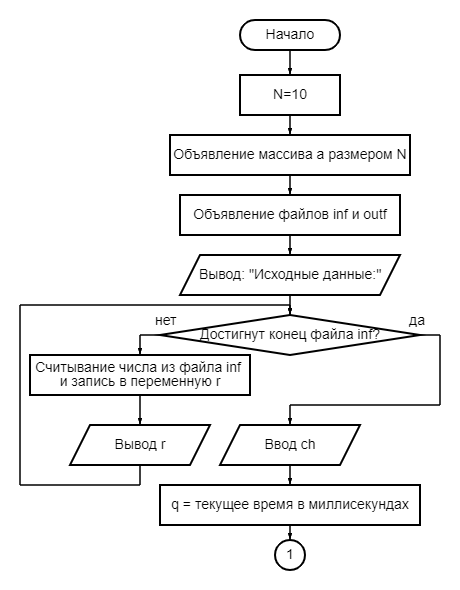


Рисунок 1 – Схема алгоритма (1/6)

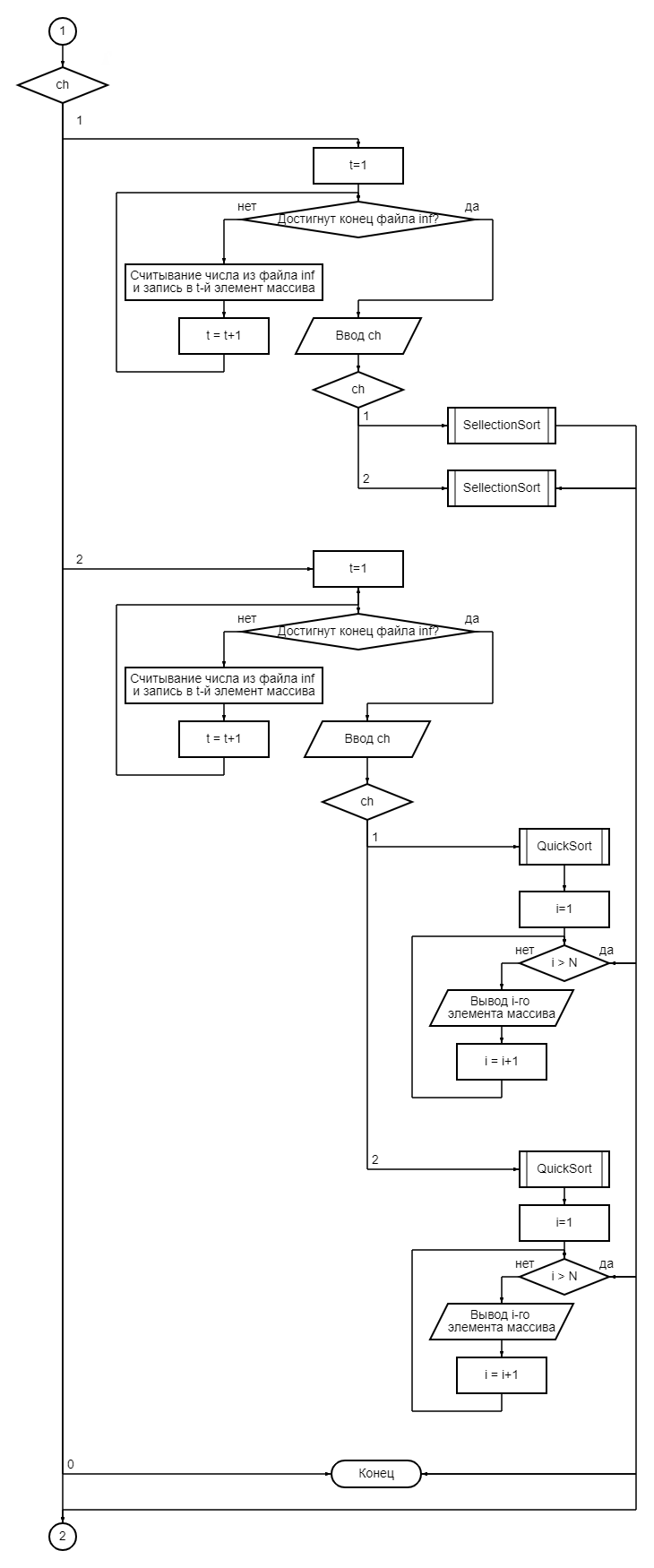


Рисунок 2 – Схема алгоритма (2/6)

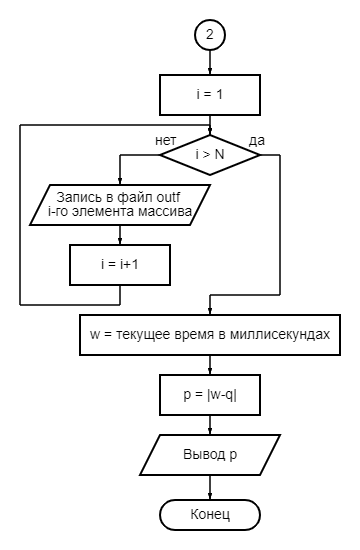


Рисунок 3 – Схема алгоритма (3/6)

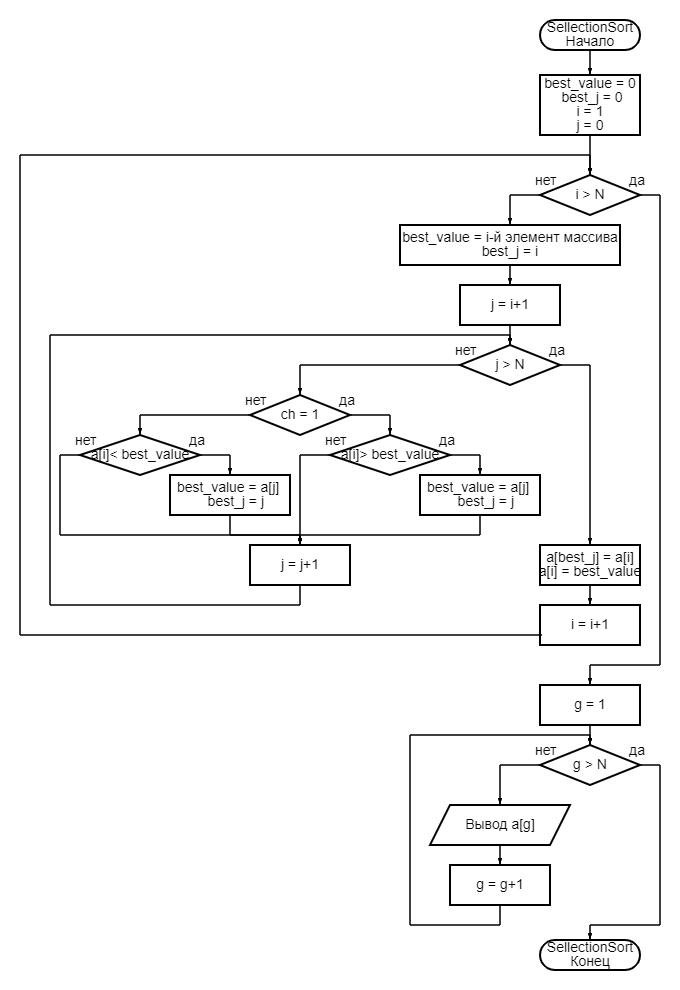


Рисунок 4 – Схема алгоритма (4/6)

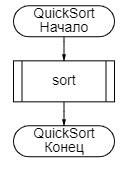


Рисунок 5 – Схема алгоритма (5/6)

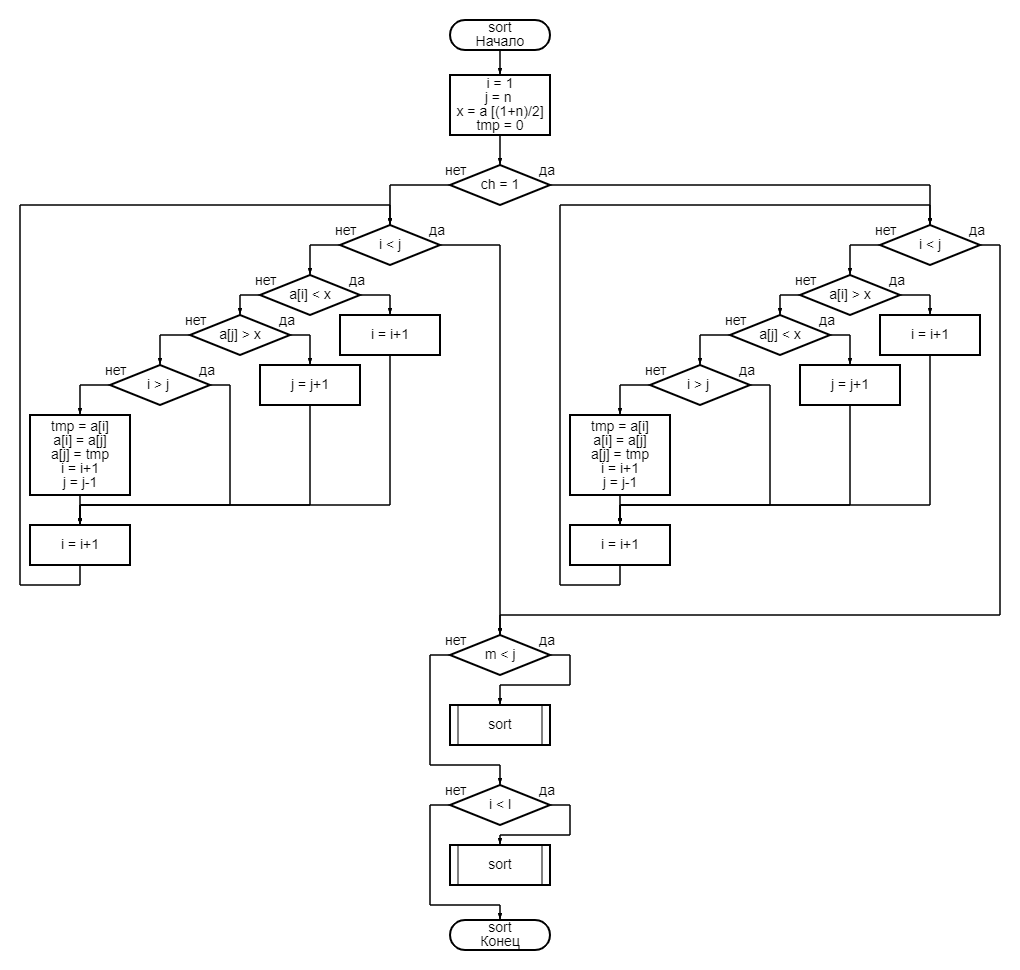


Рисунок 6 – Схема алгоритма (6/6)

4. Код программы:

**uses** CRT;**uses** System;**const** N=10;**type** a = **array** [1..N] **of** integer;

**var** inf,outf:**file of** integer;

**procedure** SellectionSort(**var** a:a;min,max,ch:integer);

**begin**

**var** i, j, best\_value, best\_j: longint;

**for** i:=min **to** max **do begin**

best\_value:=a[i];

best\_j:=i;

**for** j:=i+1 **to** max **do begin**

**if** ch=1 **then begin**

**if** a[j]>best\_value **then begin**

best\_value:=a[j];

best\_j:=j;

**end**;**end**

**else if** ch=2 **then begin**

**if** a[j]<best\_value **then begin**

best\_value:=a[j];

best\_j:=j;

**end**;**end**;

**end**;

a[best\_j]:=a[i];

a[i]:=best\_value;

**end**;TextColor(blue);

**for var** g:=1 **to** high(a) **do** print(a[g]);

**end**;

**procedure** sort(**var** a:a; m, l,k: Integer);

**begin**

**var** i,j,x,tmp:integer;

i := m;

j := l;

x := a[(m+l) **div** 2];

**if** k=1 **then begin**

**repeat**

**while** a[i] > x **Do** Inc(i);

**while** a[j] < x **Do** Dec(j);

**if** i <= j **then begin**

tmp := a[i];

a[i] := a[j];

a[j] := tmp;

Inc(i);

Dec(j)

**end**

**until** i > j;

**end**

**else begin**

**repeat**

**while** a[i] < x **Do** Inc(i);

**while** a[j] > x **Do** Dec(j);

**if** i <= j **then begin**

tmp := a[i];

a[i] := a[j];

a[j] := tmp;

Inc(i);

Dec(j)

**end**

**until** i > j;

**end**;

**if** m < j **then**

sort(a, m, j,k);

**if** i < l **then**

sort(a, i, l,k)

**end**;

**procedure** QuickSort(**var** a:a;k:integer);

**begin**

sort(a,1,n,k)

**end**;

**begin**

**var** q:integer;

**var** b:a;

assign(inf,'input.txt');reset(inf);

TextColor(blue);

println('Исходные данные:');

TextColor(Magenta);

**var** r:integer;

**while not** eof(inf) **do begin**

read(inf,r);

print(r);

**end**;println();close(inf);

**var** ch:byte;

TextColor(blue);

println('Введите 1 для сортировки данных с помощью алгоритма выбором');

println('Введите 2 для сортировки данных с помощью быстрого алгоритма');TextColor(Magenta);

println('Введите 0 для завершения программы');readln(ch);

q:=DateTime.Now.MilliSecond;

reset(inf);

**case** ch **of**

1:**begin**

**var** t:=1;

**while not** Eof(inf) **do begin**

read(inf, b[t]);

inc(t);**end**;TextColor(Magenta);

println('Введите 1 для сортировки данных по убыванию');

println('Введите 2 для сортировки данных по возрастанию');

readln(ch);TextColor(blue);

**case** ch **of**

1:SellectionSort(b, 1, N,1);

2:SellectionSort(b, 1, N,2);

**end**;

**end**;

2:**begin**

**var** t:=1;

**while not** Eof(inf) **do begin**

read(inf, b[t]);

inc(t);**end**;TextColor(Magenta);

println('Введите 1 для сортировки данных по убыванию');

println('Введите 2 для сортировки данных по возрастанию');

readln(ch);TextColor(blue);

**case** ch **of**

1:**begin** QuickSort(b,1);**for var** i:= 1 **to** N **do**

Write(b[i], ' ');**end**;

2:**begin** QuickSort(b,2);**for var** i:= 1 **to** N **do**

Write(b[i], ' ');**end**;

**end**;

**end**;

0:**begin** println('Завершение программы');**exit**();**end**;

**end**;close(inf);

assign(outf,'outfile.txt');

rewrite(outf);

**for var** i:=1 **to** N **do begin**

write(outf,b[i]);

**end**;

**var** w:integer;

w:=DateTime.Now.MilliSecond;

**var** p:=abs(w-q);

println('Время выполнения программы в миллисекундах:',p);

close(outf);

**end**.

5. Результат выполнения программы:

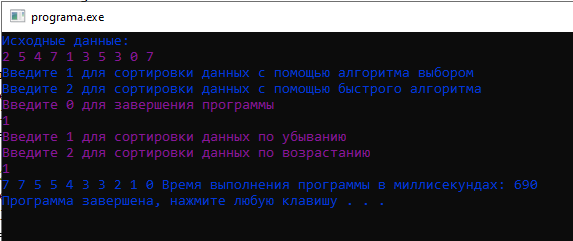


Рисунок 7 – Результат выполнения программы (1/2)

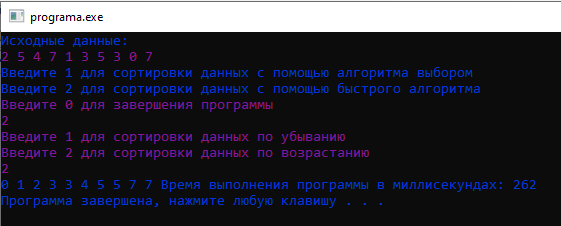


Рисунок 8 – Результат выполнения программы (2/2)

1. Вывод:

Для выполнения задания были использованы функции библиотек CRT и System языка Pascal.

Для сортировки данных были изучены и использованы методы сортировки алгоритма выбором и быстрого алгоритма.

В программе реализована возможность изменения компаратора для выбора метода сортировки (по возрастанию либо по убыванию).

Для вычисления времени выполнения программы были использованы функции DateTime.Now.MilliSecond библиотеки System.

Ко всему можно добавить использование операторов if, циклов for, внутриблочное описание переменных, инициализации переменных при описании, расширенное присваивание, для вывода используется функция Print.

Благодаря всему вышеперечисленному задание было успешно выполнено.