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 АЛГОРИДМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

Выполнил: студент учебной группы

ИСПк-204-52-00

Утёмова Ксения Александровна

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

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

**Формулировка задания:**

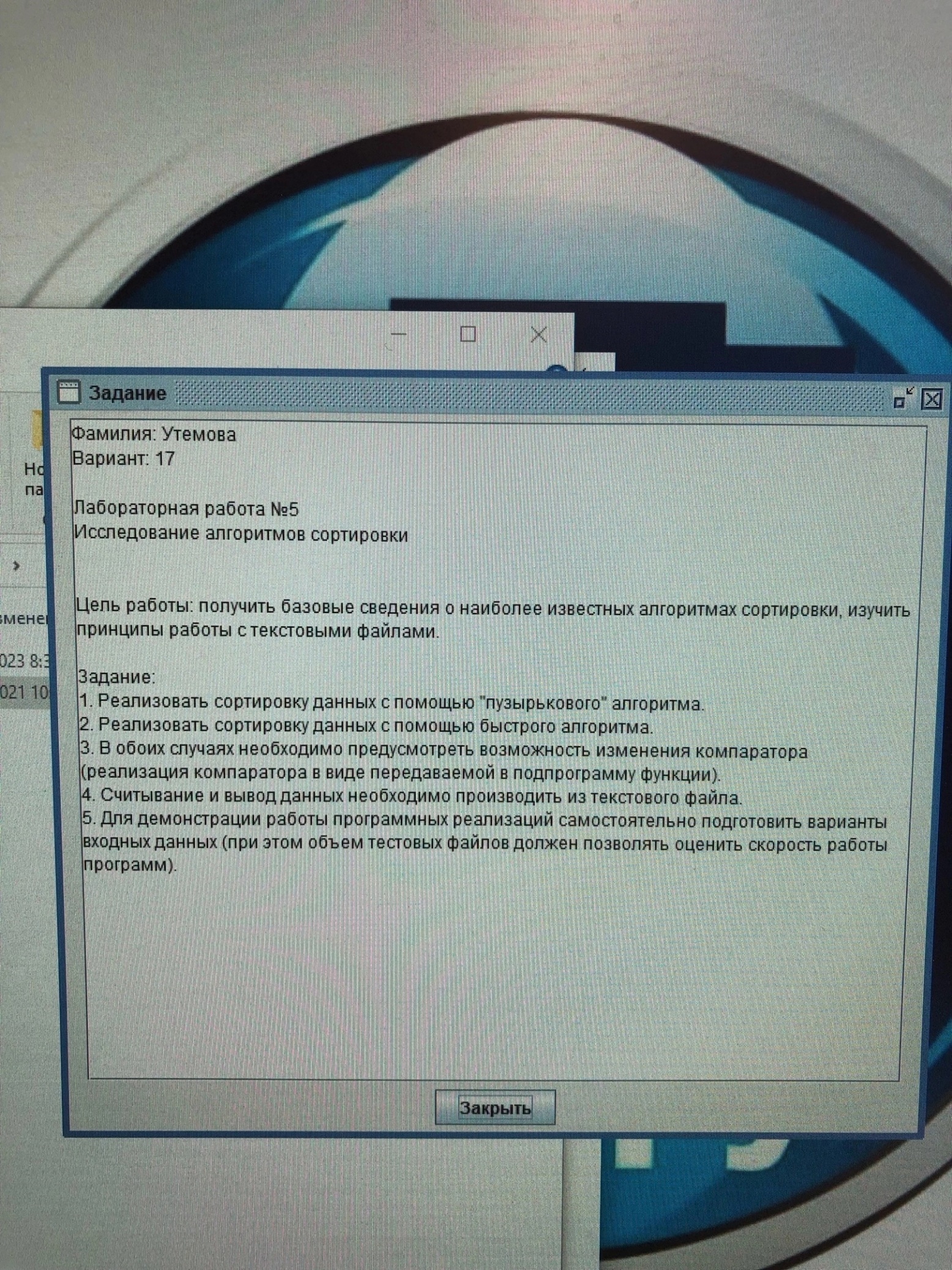
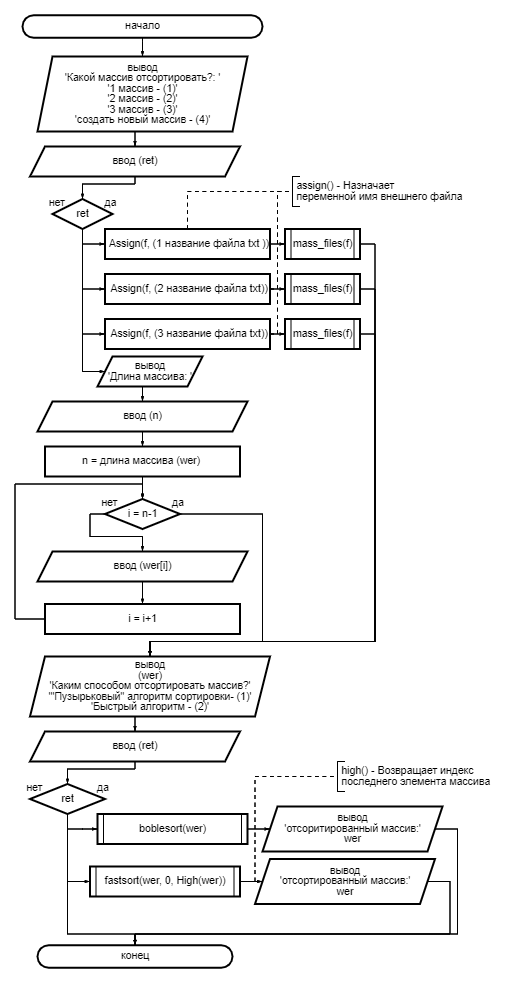


Рисунок 1 - Скриншот задания.

**Схемы алгоритмов с комментариями:**

1. **Алгоритм для всей задачи:**



1

2

4

3

2

1



Рисунок 2 - Схема алгоритма для заданий.

1. **Алгоритмы для подзадач (процедур):**

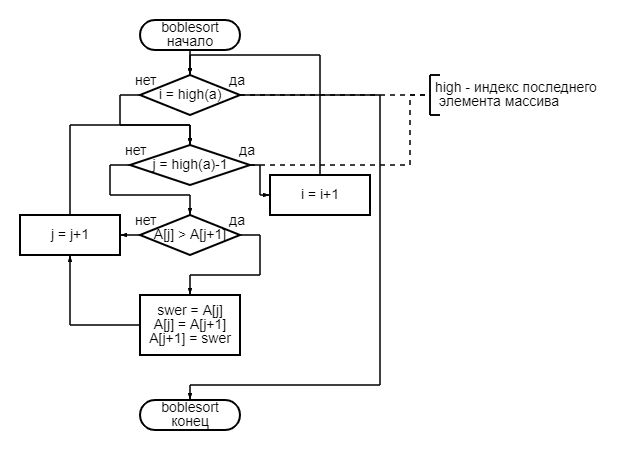


Рисунок 3 – Алгоритм подзадачи (boblesort)

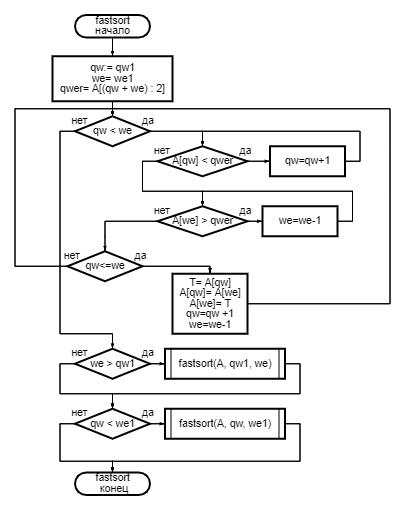


Рисунок 4 - Алгоритм функции (fastsort)

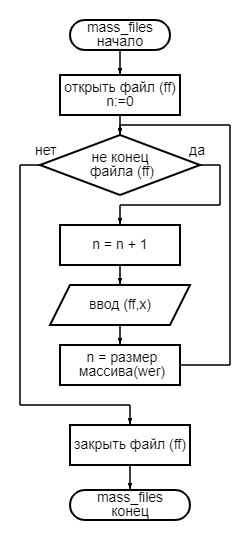


Рисунок 5 - Алгоритм для функции (mass\_files)

**Код программы:**

**var** ret,n,i,j,x: integer;

wer: **array of** integer;

f: text;

**procedure** boblesort(**var** A: **array of** integer);

**var** i, j, swer : integer;

**begin**

**for** i := 0 **to** High(A) **do**

**begin**

**for** j := 0 **to** High(A) - 1 **do**

**begin**

**if** A[j] > A[j+1] **then**

**begin**

swer := A[j];

A[j] := A[j+1];

A[j+1] := swer;

**end**;

**end**;

**end**;

**end**;

**procedure** fastsort(**var** A: **array of** integer; qw1, we1: integer);

**var** qw, we, qwer, T: Integer;

**begin**

qw:= qw1;

we:= we1;

qwer:= A[(qw + we) **div** 2];

**while** qw < we **do**

**begin**

**while** A[qw] < qwer **do** Inc(qw);

**while** A[we] > qwer **do** Dec(we);

**if** qw<=we **then**

**begin**

T:= A[qw];

A[qw]:= A[we];

A[we]:= T;

qw:=qw +1;

we:=we-1;

**end**;

**end**;

**if** we > qw1 **then**

fastsort(A, qw1, we);

**if** qw < we1 **then**

fastsort(A, qw, we1);

**end**;

**procedure** mass\_files( ff : text);

**begin**

reset(ff);

n:=0;

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

n := n + 1;

readln(ff,x);

Setlength(wer, n);

wer[n-1] := x;

**end**;

CloseFile(ff);

**end**;

**begin**

Writeln('Какой массив отсортировать?: ');

Writeln('1 массив - (1)');

Writeln('2 массив - (2)');

Writeln('3 массив - (3)');

Writeln('создать новый массив - (4)');

readln(ret);

**case** ret **of**

1 : **begin**

Assign(f, '1массив.txt');

mass\_files(f);

**end**;

2 : **begin**

Assign(f, '2массив.txt');

mass\_files(f);

**end**;

3 : **begin**

Assign(f, '3массив.txt');

mass\_files(f);

**end**;

4 : **begin**

writeln('Длина массива: ');

readln(n);

setlength(wer, n);

writeln('Введите начения для массива: ');

**for** i := 0 **to** n-1 **do**

readln(wer[i]);

**end**;

**end**;

Writeln(wer);

writeln('Каким способом отсортировать массив?');

writeln('"Пузырьковый" алгоритм сортировки- (1)');

writeln('Быстрый алгоритм - (2)');

readln(ret);

**case** ret **of**

1 : **begin**

boblesort(wer);

writeln('отсоритированный массив:');

writeln(wer)

**end**;

2 : **begin**

fastsort(wer, 0, High(wer));

writeln('отсортированный массив:');

Writeln(wer)

**end**;

**end**;

**end**.

**Результаты выполнения программ:**

1. **Результат задания сортировкой ‘пузырьком’.**

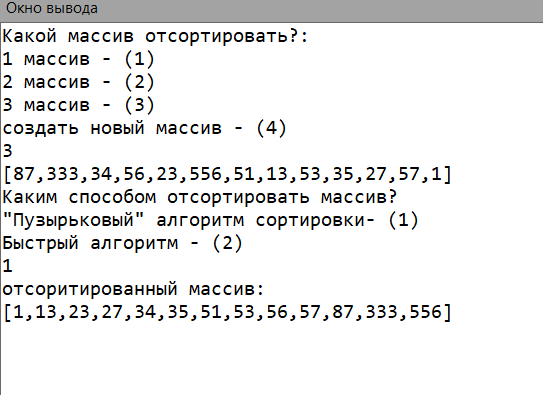


Рисунок 6 - Результат №1 для задания.

1. **Результат задания быстрой сортировкой**.

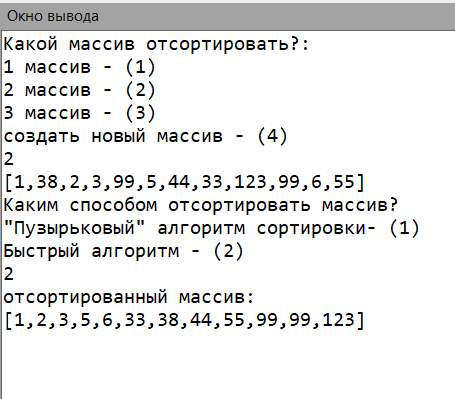
****

Рисунок 7 - Результат №2 для задания.

Результаты программ мы получили правильные, соответствующие ожидаемым ответам. Полностью выполняют все условия задания. Использованные текстовые файлы видны программе и использованы в виде носителя данных.

**Вывод:**

Во время выполнения пятой домашней контрольной работы мы смогли написать программные коды для решения поставленных задач по теме “Исследование алгоритмов сортировки.” Получили базовые сведения о наиболее известных алгоритмах сортировки и изучили принципы работы с текстовыми файлами.

Результатом нашей работы стало решение поставленной задачи на языке программирования Pascal, в программе PascalABC.Net.

Изучить способы передачи данных с помощью использования внешних текстовых файлов. Закрепили умение создавать схемы алгоритма и разобрались, как писать и оформлять отчеты по учебной дисциплине “Основы алгоритмизации и программирования.”