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-203-52-00

Дворников Владислав Максимович

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 вставок.

2)Реализовать сортировку данных с помощью быстрого алгоритма.

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:

**program** DKR5;

**uses** crt, system;

**var**

input: **file of** char;

output: text;

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

start, start1, stop, stop1, n: integer;

s: string;

**function** vstavsort: integer;

**var**

i, j: integer;

a: char;

**begin**

**for** i := 2 **to** n **do**

**begin**

a := arr[i];

j := i - 1;

**while** (j >= 1) **and** (arr[j] > a) **do**

**begin**

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

j := j - 1;

**end**;

arr[j + 1] := a;

**end**;

**end**;

**function** vstavsortub: integer;

**var**

i, j: integer;

a: char;

**begin**

**for** i := 2 **to** n **do**

**begin**

a := arr[i];

j := i - 1;

**while** (j >= 1) **and** (arr[j] < a) **do**

**begin**

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

j := j - 1;

**end**;

arr[j + 1] := a;

**end**;

**end**;

**procedure** fastsort(l, r: integer);

**var**

i, j, m: integer;

arr1, y1: char;

**begin**

i := l;

j := r;

m := round((l + r) / 2);

arr1 := arr[m];

**repeat**

**while** arr[i] < arr1 **do** inc(i);

**while** arr[j] > arr1 **do** dec(j);

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

**begin**

y1 := arr[i];

arr[i] := arr[j];

arr[j] := y1;

inc(i);

dec(j);

**end**;

**until** i > j;

**if** l < j **then** fastsort(l, j);

**if** i < r **then** fastsort(i, r);

**end**;

**procedure** fastsortub(first, last: integer);

**var**

f, l: integer; mid, count: char;

**begin**

f := first;

l := last;

mid := arr[(f + l) **div** 2];

**repeat**

**while** arr[f] > mid **do** inc(f);

**while** arr[l] < mid **do** dec(l);

**if** f <= l **then**

**begin**

count := arr[f];

arr[f] := arr[l];

arr[l] := count;

inc(f);

dec(l);

**end**;

**until** f > l;

**if** first < l **then** fastsortub(first, l);

**if** f < last **then** fastsortub(f, last);

**end**;

**begin**

assign(input, 'j:input.txt'); assign(output, 'j:output.txt');

rewrite(input); rewrite(output);

writeln('Введите длинну массива');

readln(n);

writeln('Введите массив');

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

**begin**

readln(arr[i]);

write(input, arr[i]);

**end**;

**var** st: integer;

st := 1;

**while** st <> 0 **do**

**begin**

clrscr;

writeln('Выберите сортировку');

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

writeln('Сортировка быстрым алгоритимом по возрастанию - 2');

writeln('Сортировка вставкой по убыванию - 3');

writeln('Сортировка быстрым алгоритимом по убыванию - 4');

writeln('Выход из программы - 0');

readln(st);

start := DateTime.Now.Second;

start1 := DateTime.Now.MilliSecond;

**case** st **of**

1: vstavsort;

2: fastsort(1, n);

3: vstavsortub;

4: fastsortub(1, n);

0: **exit**;

**end**;

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

print(output, arr[i]);

stop := DateTime.Now.Second;

stop1 := DateTime.Now.MilliSecond;

close(output);

reset(output);

read(output, s);

close(output);

rewrite(output);

write(s);

writeln;

writeln('Время выполнения: ', stop - start, ' секнуд, ', abs(stop1 - start1), ' милисекунд');

writeln('Для продолжения дважды нажмите enter');

readln;

readln;

**end**;

**end**.

1. Результат выполнения программы:

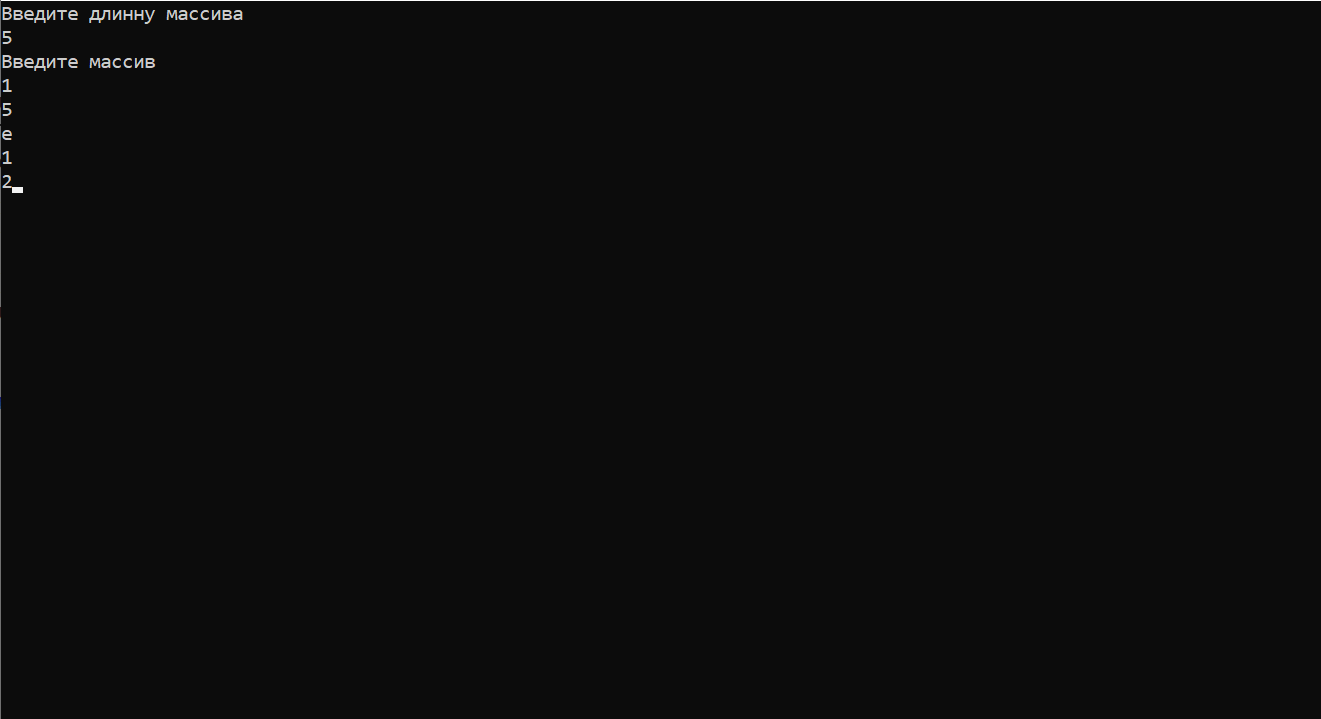


Рисунок 1- Результаты выполнения программы (1/3)

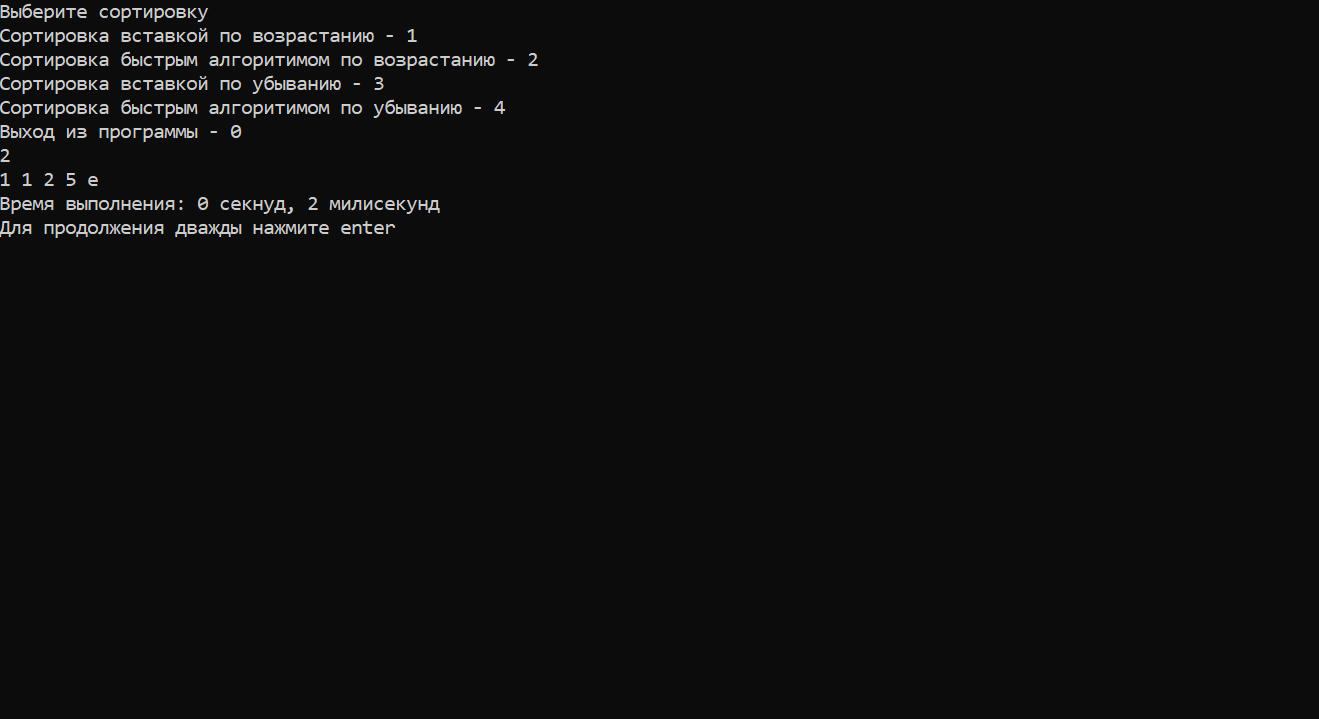


Рисунок 2- Результаты выполнения программы (2/7)

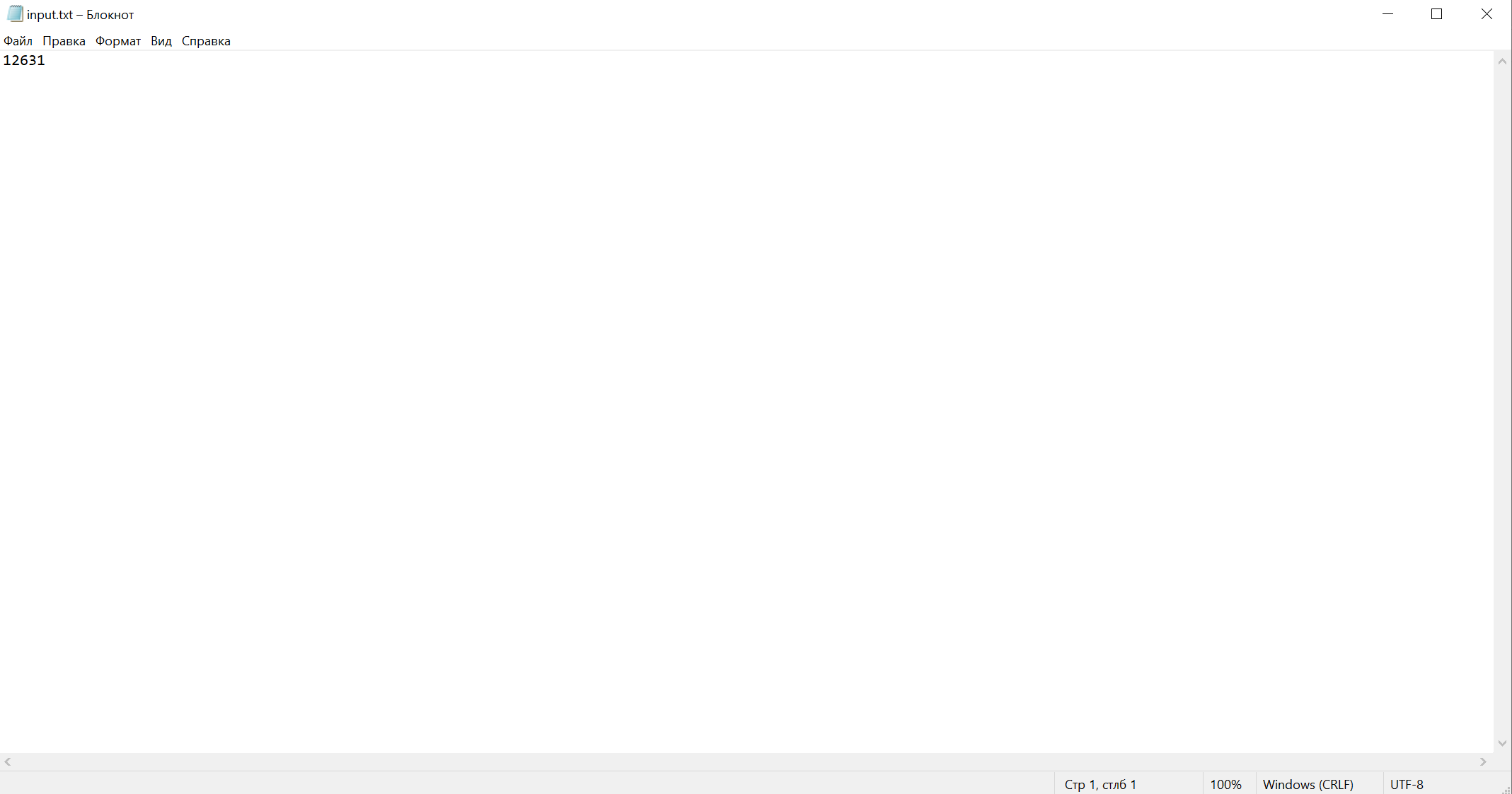


Рисунок 3- Результаты выполнения программы (3/3)

1. Вывод:

В этой домашней контрольной работе были изучены сортировка быстрым алгоритмом и сортировка вставкой. Была изучена библиотека System в языке программирования PascalABC.NET, которая была необходима для вычисления длительности работы сортировки.

Возникли небольшие трудности с сортировкой быстрым алгоритмом по убыванию, потому что её реализация немного отличалась от сортировки быстрым алгоритмом по возрастанию. Так же не сразу стало ясно как должен осуществляться подсчёт времени, которое программа тратит на сортировку данных. В конечном итоге, после некоторых размышлений работа была успешно завершена.