

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития  
Кафедра инфокоммуникаций

**ОТЧЕТ**  
**ПО ПРАКТИЧЕСКОЙ РАБОТЕ №5**  
**дисциплины «Алгоритмизация»**  
**Вариант \_\_\_\_**

Выполнил:  
Репкин Александр Павлович  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение средств  
вычислительной  
техники и автоматизированных  
систем», очная форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Р.А., канд. техн. наук,  
доцент кафедры инфокоммуникаций

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

## Порядок выполнения работы:

1. Создана программа на основе поставленного задания – необходимо проанализировать, почему метод пузырьковой сортировки плох, на основе полученных данных о затраченном времени необходимо построить график, используя метод наименьших квадратов.

```
public static void swapping(int first, int second, List<Integer> numbers) {
    int a = numbers.get(first);
    numbers.set(first, numbers.get(second));
    numbers.set(second, a);
}

no usages
public static void main(String[] args) {
    while (length <= 45000) {
        List<Integer> numbers = new ArrayList<>();
        results.append(length).append(";");
        Random value = new Random();
        for (int i = 0; i < length; i++) numbers.add(value.nextInt());
        long startTime = System.nanoTime();
        for (int out = length - 1; out >= 1; out--) {
            for (int in = 0; in < out; in++) {
                if (numbers.get(in) > numbers.get(in + 1))
                    swapping(in, second: in + 1, numbers);
            }
        }
        long endTime = System.nanoTime();
        long spent_time = (endTime - startTime); //divide by 1000000 to get milliseconds.
        results.append(spent_time / 1000000).append(",") .append((spent_time / 10000) % 100).append("\n");
        length += 250;
        File outputFile = new File( pathname: "Results.txt");
        try (OutputStream outputStream = new FileOutputStream(outputFile)) {
            outputStream.write(results.toString().getBytes(StandardCharsets.UTF_8));
            outputStream.flush();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

Рисунок 1. Полученный код программы.

2. Составлен график относительно полученных данных – получен экспоненциальный рост затрат времени на сортировку.

Длина массива	Время (мс), ушедшее на сортировку пузырьком
500	40,49
750	17,1
1000	49,27
1250	53,11
1500	20,11
1750	26,51
2000	29,97
2250	39,1
2500	32,86
2750	60,87
3000	73,58
3250	92,65
3500	69,55
3750	98,39
4000	107,98
4250	102,3
4500	109,5
4750	123,6
5000	193,67
5250	149,55
5500	222,17



Рисунок 2. Полученный график.

**Вывод:** в ходе выполнения практической работы был рассмотрен метод пузырьковой сортировки. Из полученных данных выявлен экспоненциальный рост времени, требуемого на сортировку массива, что недопустимо при постоянной работе с сортировкой массивов.