

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 ЛАБОРАТОРНОЙ РАБОТЕ №5**  
**дисциплины «Алгоритмизация»**

Выполнил:  
Гайчук Дарья Дмитриевна  
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.-доцент кафедры  
инфокоммуникаций

---

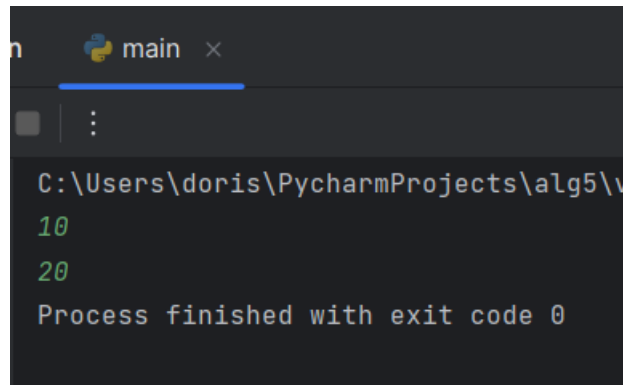
(подпись)

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

Ставрополь, 2023 г.

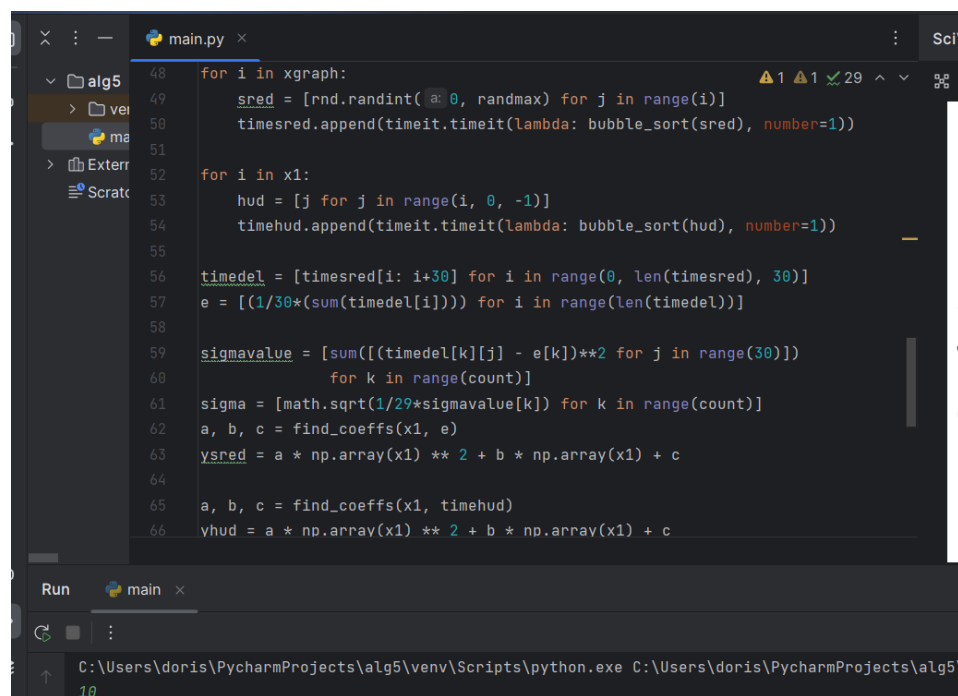
Порядок выполнения работы:

Задание 1. Необходимо проанализировать метод пузырьковой сортировки. Построить график относительно метода наименьшего квадрата.



```
C:\Users\doris\PycharmProjects\alg5\venv\Scripts\python.exe C:\Users\doris\PycharmProjects\alg5\main.py
10
20
Process finished with exit code 0
```

Рисунок 1. Исходные данные



```
48 for i in xgraph:
49     sred = [rnd.randint(a: 0, randmax) for j in range(i)]
50     timesred.append(timeit.timeit(lambda: bubble_sort(sred), number=1))
51
52 for i in x1:
53     hud = [j for j in range(i, 0, -1)]
54     timehud.append(timeit.timeit(lambda: bubble_sort(hud), number=1))
55
56 timedel = [timesred[i: i+30] for i in range(0, len(timesred), 30)]
57 e = [(1/30*(sum(timedel[i]))) for i in range(len(timedel))]
58
59 sigmavalue = [sum([(timedel[k][j] - e[k])**2 for j in range(30)])
60               for k in range(count)]
61 sigma = [math.sqrt(1/29*sigmavalue[k]) for k in range(count)]
62 a, b, c = find_coeffs(x1, e)
63 ypred = a * np.array(x1) ** 2 + b * np.array(x1) + c
64
65 a, b, c = find_coeffs(x1, timehud)
66 yhud = a * np.array(x1) ** 2 + b * np.array(x1) + c
```

Рисунок 2. Код программы

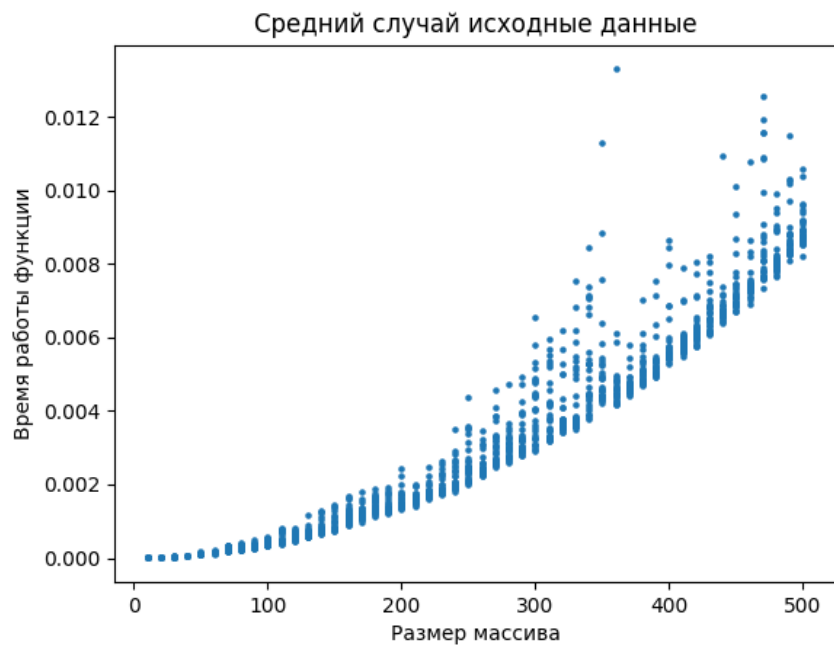


Рисунок 3. Исходные данные на графике

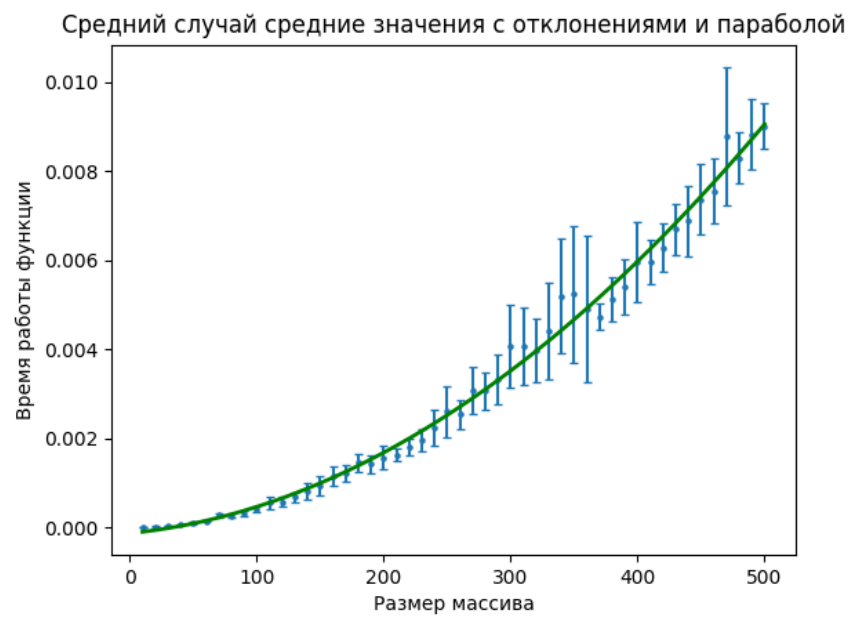


Рисунок 4. График среднего случая

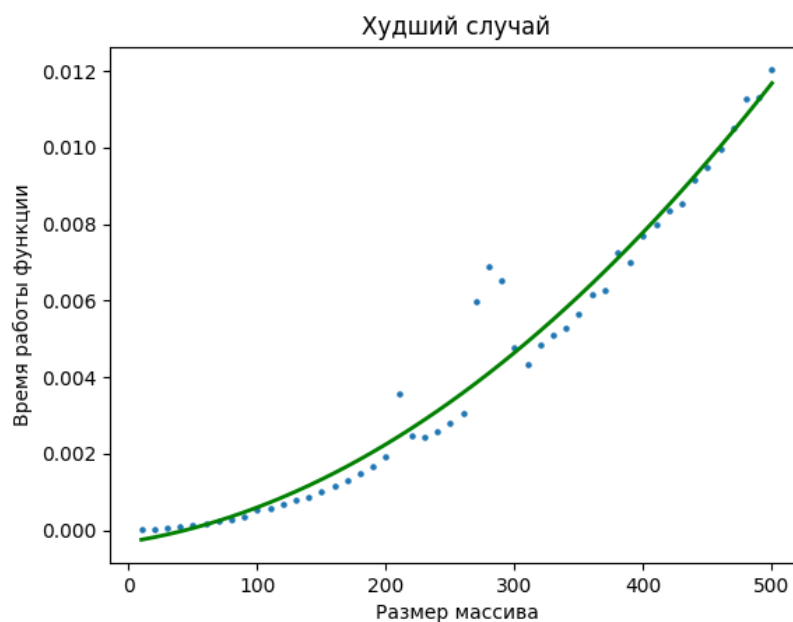


Рисунок 5. График худшего случая

Вывод: время выполнения пузырьковой сортировки имеет квадратичную зависимость от размера входного массива. Это означает, что при увеличении размера массива вдвое, время выполнения сортировки будет увеличиваться в четыре раза. Метод пузырьковой сортировки не является эффективным для больших массивов данных из-за его высокой временной сложности.