

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»

Выполнила:
Беседина Инга Олеговна
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

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

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Порядок выполнения работы:

```
import timeit as ti
import numpy as np
import statistics
import matplotlib.pyplot as plt

mysetup = """
from __main__ import n
from __main__ import a
from __main__ import bubble_sort
"""

def bubble_sort(n, arr):
    for i in range(n - 1):
        for j in range(n - i - 1):
            if arr[j] > arr[j + 1]:
                arr[j], arr[j + 1] = arr[j + 1], arr[j]

N = 400
x = np.array(range(10, N + 1, 10))
y = []
for n in range(10, N+1, 10):
    a = np.array(np.random.randint(-100, 100, n))
    a[::-1].sort()
    y.append(ti.timeit(setup=mysetup, stmt="bubble_sort(n, a)", number=10))
print("Коэффициент корреляции:", statistics.correlation(x, y))
model = np.poly1d(np.polyfit(x, y, 2))
x_new = np.linspace(x.min(), x.max(), 200)
y_new = model(x_new)
print("Уравнение кривой:")
print(model)
plt.scatter(x, y, s=7, color='b')
plt.plot(x_new, y_new, color='r')
plt.xlabel('Размер массива')
plt.ylabel('Время работы функции')
plt.title('Худший случай')
plt.show()
```

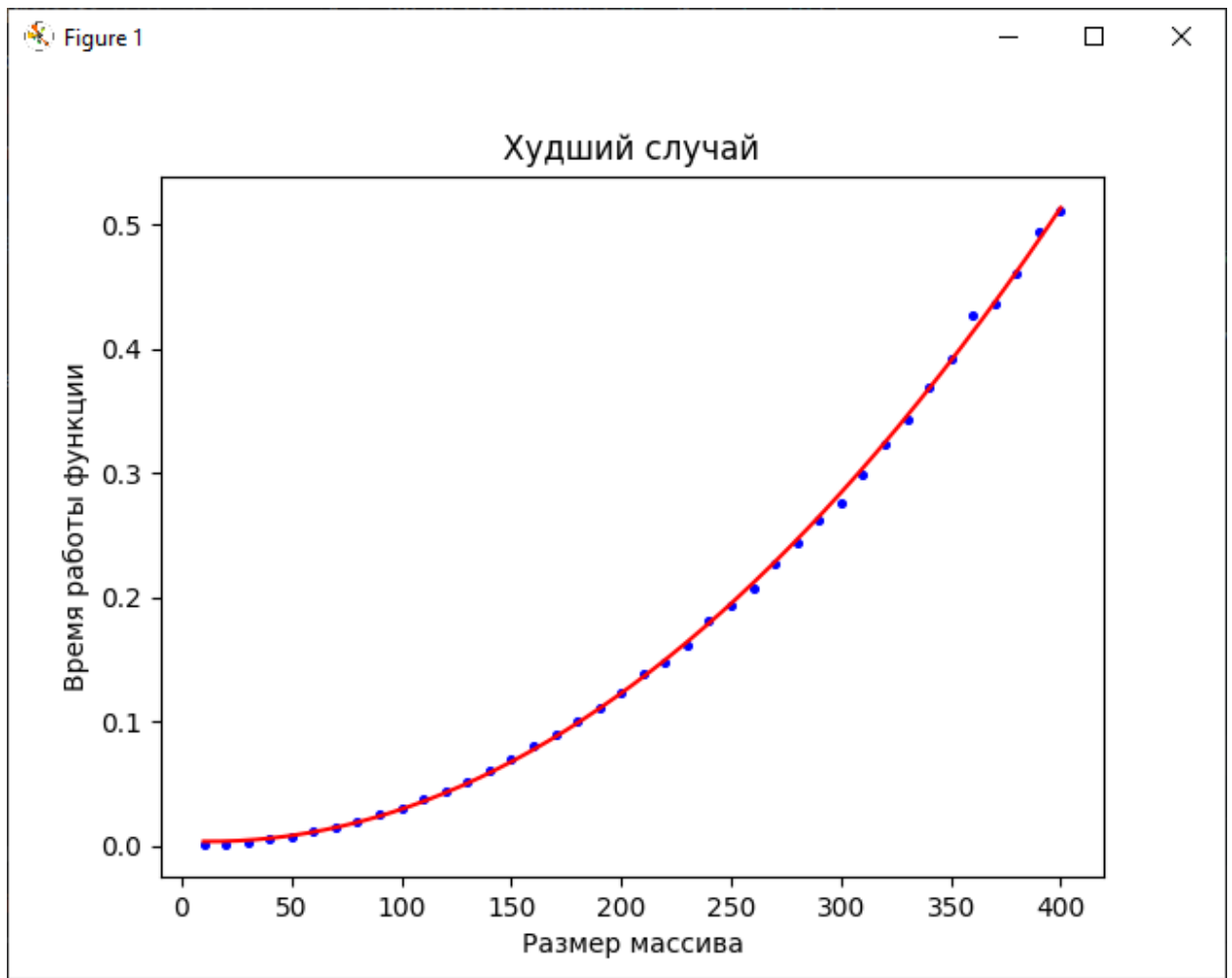


Рисунок 1. Зависимость времени работы функции от количества элементов в массиве в худшем случае

Коэффициент корреляции: 0.9656730401117204
Уравнение кривой:
$$3.401e-06 x^2 - 8.594e-05 x + 0.003916$$

Рисунок 2. Коэффициент корреляции и уравнение кривой

```
import timeit as ti
import numpy as np
import statistics
import matplotlib.pyplot as plt

def bubble_sort(n, arr):
    for i in range(n - 1):
        for j in range(n - i - 1):
            if arr[j] > arr[j + 1]:
```

```

arr[j], arr[j + 1] = arr[j + 1], arr[j]

N = 400
k = 30
x = np.array(range(10, N + 1, 10))
E = []
q = []
t = []
for n in range(10, N + 1, 10):
    a = np.array(np.random.randint(-100, 100, n))
    for i in range(30):
        start_time = ti.default_timer()
        bubble_sort(n, a)
        end_time = ti.default_timer() - start_time
        t.append(end_time)
    E.append(1/k*np.sum(t))
    q.append(np.std(t))
    t.clear()
print("Коэффициент корреляции:", statistics.correlation(x, E))
model = np.poly1d(np.polyfit(x, E, 2))
x_new = np.linspace(x.min(), x.max(), 200)
y_new = model(x_new)
print("Уравнение кривой:")
print(model)
plt.plot(x_new, y_new, color='r')
plt.errorbar(x, E, yerr=q, fmt='o', markersize=2, capsize=2)
plt.xlabel('Размер массива')
plt.ylabel('Время работы функции')
plt.title('Средний случай')
plt.show()

```

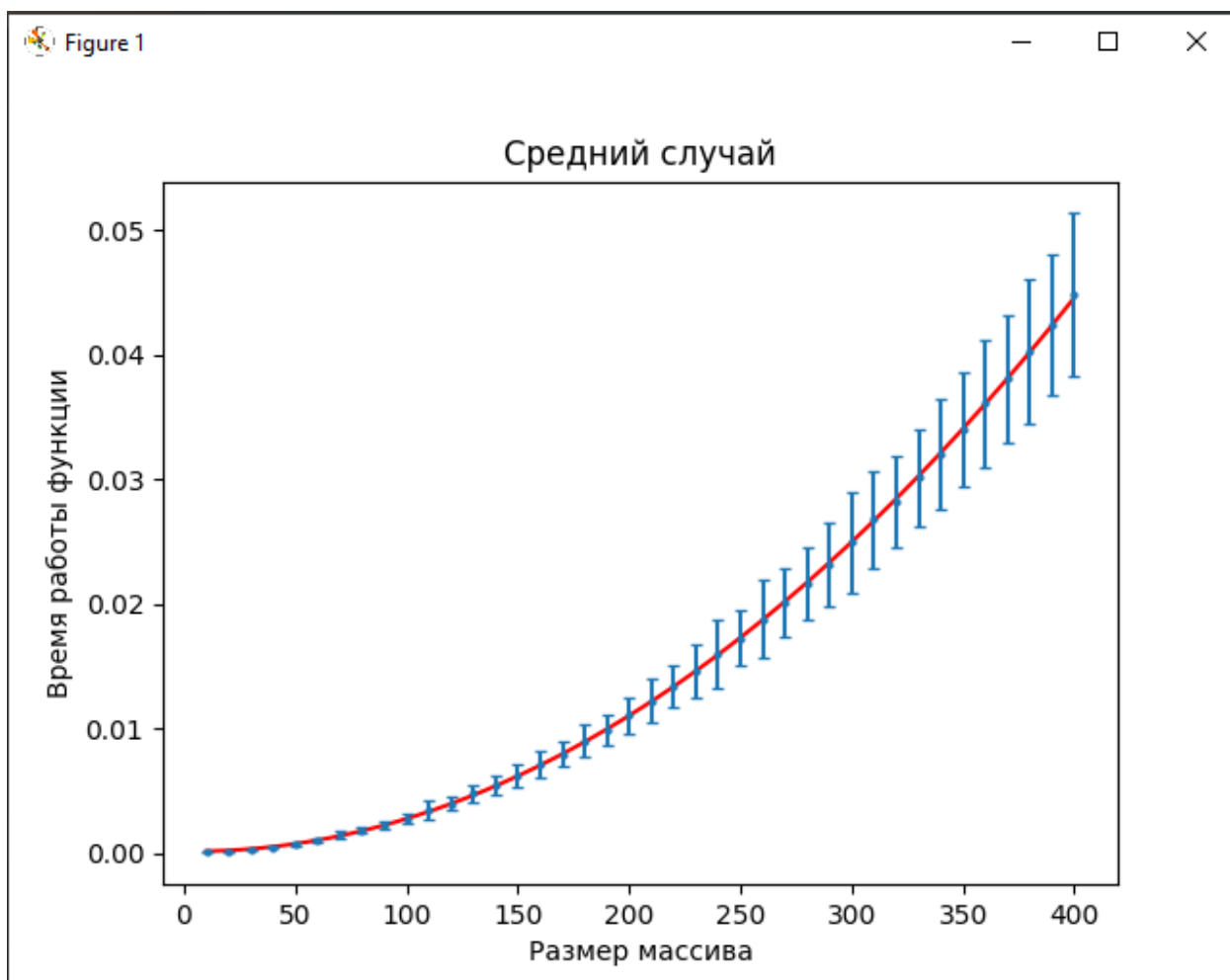


Рисунок 3. стандартное отклонение

```
Коэффициент корреляции: 0.9686267631751508  
Уравнение кривой:  
2  
2.831e-07 x - 2.214e-06 x + 0.0001299
```

Рисунок 4. Коэффициент корреляции и уравнение кривой