

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 №4
дисциплины «Алгоритмизация»
Вариант 9

Выполнил:
Дудкин Константин Александрович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»
направление «Программное
обеспечение средств вычислительной
техники и автоматизированных
систем»,
очная форма обучения

(подпись)

Руководитель практики:
доцент кафедры инфокоммуникаций,
кандидат технических наук
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Порядок выполнения работы

Провел анализ алгоритма поиска максимума и минимума, написал программу для его исследования, составил график линейной зависимости для данного алгоритма и нашел коэффициент корреляции:

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import timeit
5  import random
6  import matplotlib.pyplot as plt
7  import numpy as np
8
9
10 # Функции
11 usage new *
12 def MinSearch_experiment(arr_size):
13     arr = random.sample(range(1, arr_size + 1), arr_size)
14     execution_time = timeit.timeit(lambda: min(arr), number=1000)
15     return execution_time
16
17 usage new *
18 def MaxSearch_experiment(arr_size):
19     arr = random.sample(range(1, arr_size + 1), arr_size)
20     execution_time = timeit.timeit(lambda: max(arr), number=1000)
21     return execution_time
```

Рисунок 1. Код программы исследования (часть 1)

```

22  ▶ if __name__ == '__main__':
23
24      # Параметры для исследования
25      array_sizes = [100, 200, 300, 400, 500, 600, 700, 800, 900, 1000,
26                     2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000]
27      num_experiments = 5
28      min_execution_times = []
29      max_execution_times = []
30
31      for size in array_sizes:
32          min_times = []
33          max_times = []
34
35          for _ in range(num_experiments):
36              min_time = MinSearch_experiment(size)
37              max_time = MaxSearch_experiment(size)
38
39              min_times.append(min_time)
40              max_times.append(max_time)
41
42          # Подсчет среднего времени
43          avg_min_time = sum(min_times) / num_experiments
44          avg_max_time = sum(max_times) / num_experiments
45
46          min_execution_times.append(avg_min_time)
47          max_execution_times.append(avg_max_time)
48

```

Рисунок 2. Код программы исследования (часть 2)

```

49     # Выполнение линейной регрессии и расчет коэффициента корреляции для обоих случаев.
50     x = np.array(array_sizes)
51     y_min = np.array(min_execution_times)
52     y_max = np.array(max_execution_times)
53
54     A_min = np.vstack([x, np.ones(len(x))]).T
55     a_min, b_min = np.linalg.lstsq(A_min, y_min, rcond=None)[0]
56
57     A_max = np.vstack([x, np.ones(len(x))]).T
58     a_max, b_max = np.linalg.lstsq(A_max, y_max, rcond=None)[0]
59
60     correlation_coefficient_min = np.corrcoef(x, y_min)[0, 1] ** 2
61     correlation_coefficient_max = np.corrcoef(x, y_max)[0, 1] ** 2
62
63     # Визуализация
64     plt.figure(figsize=(12, 6))
65
66     plt.subplot(*args: 1, 2, 1)
67     plt.plot(*args: x, y_min, 'o', label='Поиск минимума')
68     plt.plot(*args: x, a_min * x + b_min, 'r', label=f'Коэффициент корреляции (R^2={correlation_coefficient_min:.5f})')
69     plt.xlabel('Размер массива')
70     plt.ylabel('Время выполнения (секунды)')
71     plt.title('Аналитика поиска минимума')
72     plt.legend()
73
74     plt.subplot(*args: 1, 2, 2)
75     plt.plot(*args: x, y_max, 'o', label='Поиск максимума')
76     plt.plot(*args: x, a_max * x + b_max, 'g', label=f'Коэффициент корреляции (R^2={correlation_coefficient_max:.5f})')
77     plt.xlabel('Размер массива')
78     plt.ylabel('Время выполнения (секунды)')
79     plt.title('Аналитика поиска максимума')
80     plt.legend()
81
82     plt.tight_layout()
83     plt.show()

```

Рисунок 3. Код программы исследования (часть 3)

```

85     # Вывод в консоль
86     print(f"Поиск минимума: Линейная зависимость:  $y = \{a\_min:.5f\} * x + \{b\_min:.5f\}$ ")
87     print(f"Поиск минимума: Коэффициент линейной корреляции: {correlation_coefficient_min:.5f}")
88
89     print(f"Поиск максимума: Линейная зависимость:  $y = \{a\_max:.5f\} * x + \{b\_max:.5f\}$ ")
90     print(f"Поиск максимума: Коэффициент линейной корреляции: {correlation_coefficient_max:.5f}")

```

Рисунок 4. Код программы исследования (часть 4)

```

/usr/bin/python3.11 /home/code_raider/git/Algorithm_PR4/Task.py
Поиск минимума: Линейная зависимость:  $y = 0.00001 * x + -0.00053$ 
Поиск минимума: Коэффициент линейной корреляции: 0.99946
Поиск максимума: Линейная зависимость:  $y = 0.00001 * x + -0.00054$ 
Поиск максимума: Коэффициент линейной корреляции: 0.99945

Process finished with exit code 0

```

Рисунок 5. Вывод результатов в консоль

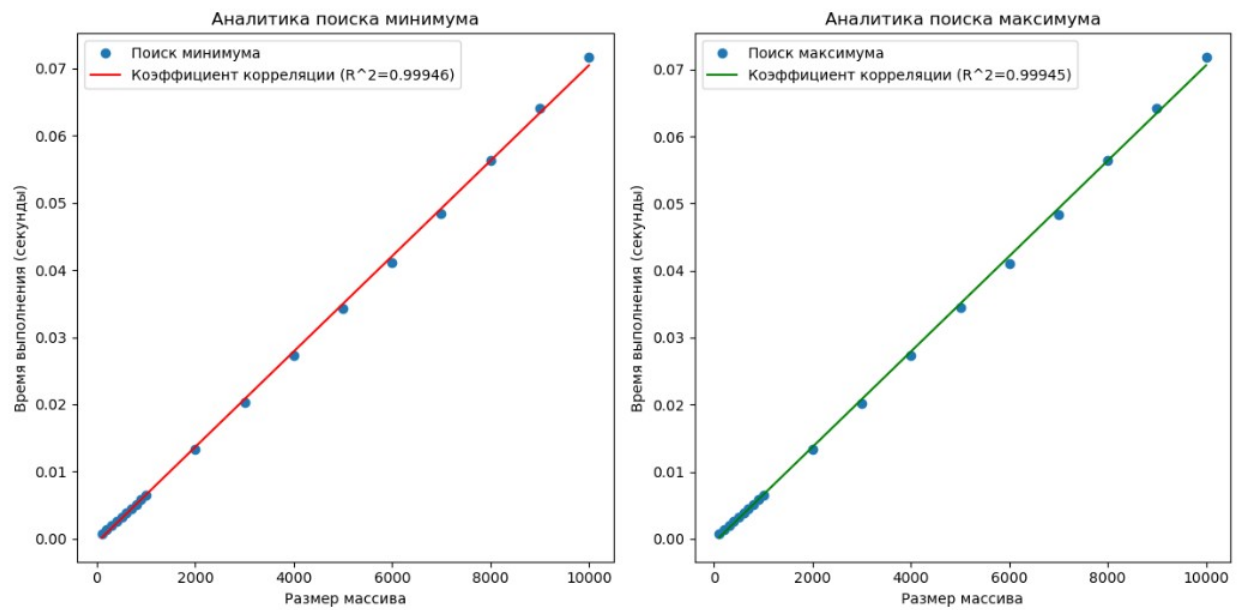


Рисунок 6. Графики линейной зависимости

Вывод: В ходе работы было проведено исследования алгоритма поиска максимума и минимума, исследована линейная зависимость и найдена корреляция для каждого из случаев