

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»
Вариант 29

Выполнил:
Саенко Андрей Максимович
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», заочная
форма обучения

(подпись)

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

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____
Ставрополь, 2023 г.

Тема: Метод сортировки пузырьком

Цель: Изучить метод сортировки пузырьком

Порядок выполнения работы:

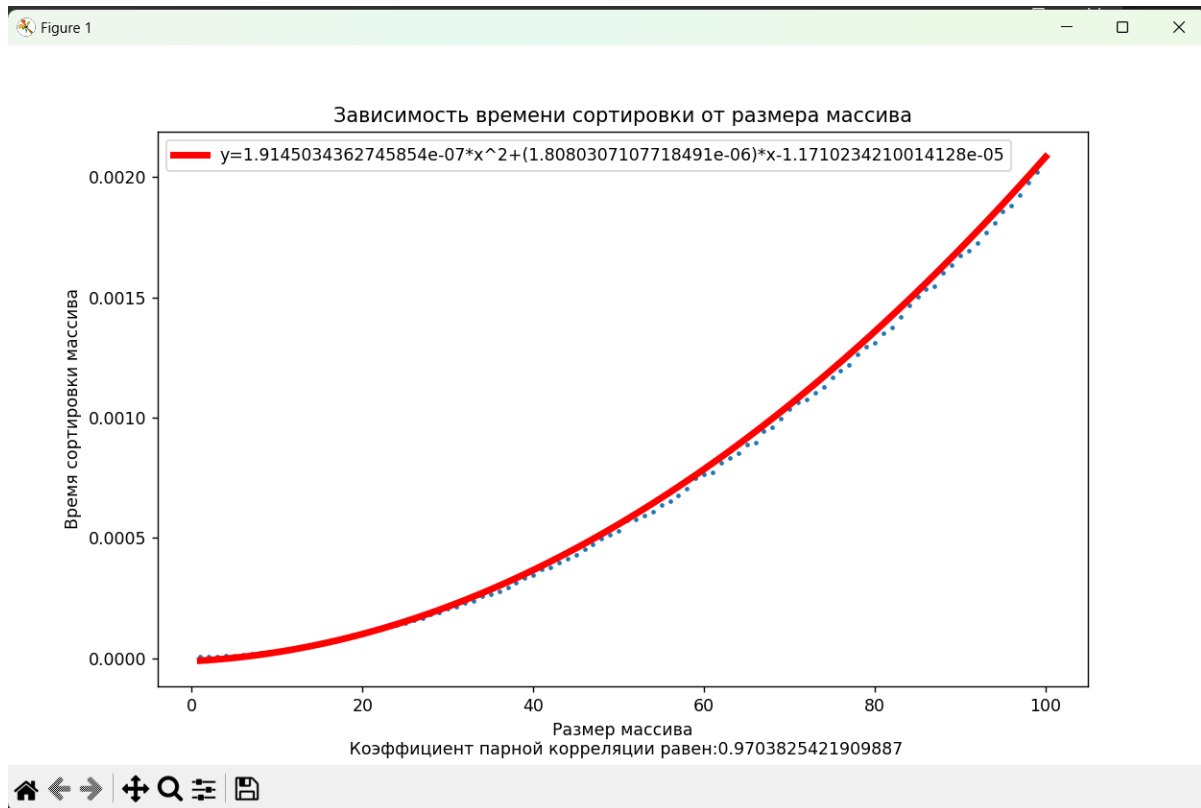


Рисунок 1 – График зависимости времени сортировки от размера массива

Код программы:

```
from random import randint
import random
import numpy as np
import timeit
import matplotlib.pyplot as plt
from math import sqrt

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

def correlation(array_of_values_x, array_of_values_y):
    sigma_first = 0
    sigma_second = 0
    sigma_third = 0
    average_value_x = 0
    average_value_y = 0
    sum_x = 0
    sum_y = 0

    for i in range(len(array_of_values_x)):
        sum_x += array_of_values_x[i]
        sum_y += array_of_values_y[i]
    average_value_x = sum_x/len(array_of_values_x)
    average_value_y = sum_y/len(array_of_values_y)
```

```

        for i in range(len(array_of_values_x)):
            sigma_first += ((array_of_values_x[i]-average_value_x)*
                            (array_of_values_y[i]-average_value_y))
            sigma_second += (array_of_values_x[i]-average_value_x)**2
            sigma_third += (array_of_values_y[i]-average_value_y)**2

        pair_correlation_coefficient = sigma_first/(sqrt(sigma_second)*
                                                    sqrt(sigma_third))

        return pair_correlation_coefficient
time_min = []
x = []
for i in range(1,101):
    arr = [0 for i in range(0,i)]
    x.append(i)
    for j in range(0,len(arr)):
        arr[j] = random.randint(500,1000)
    search_time = (timeit.timeit(lambda: bubble_sort(arr), number=50))/50
    print("Время сортировки массива из ",i," элементов: ",search_time,"\n")
    print(arr,"\n")
    time_min.append(search_time)
sum_y = sum(time_min)
sum_x = sum(x)
sum_x2 = 0
sum_x3 = 0
sum_x4 = 0
sum_yx2 = 0
sum_yx = 0
for i in range(0,len(x)):
    sum_x2 += i*i
    sum_x3 += i*i*i
    sum_x4 += i*i*i*i
    sum_yx2 += time_min[i]*i*i
    sum_yx += time_min[i]*i
an = len(x)
matrix = np.array([[sum_x4, sum_x3,sum_x2],
                   [sum_x3, sum_x2, sum_x],
                   [sum_x2, sum_x, an]])
det = np.linalg.det(matrix)
matrix_x = np.array([[sum_x4, sum_x3,sum_yx2],
                     [sum_x3, sum_x2, sum_yx],
                     [sum_x2, sum_x, sum_y]])
first_det = np.linalg.det(matrix_x)
matrix_y = np.array([[sum_x4, sum_yx2,sum_x2],
                     [sum_x3, sum_yx, sum_x],
                     [sum_x2, sum_y, an]])
second_det = np.linalg.det(matrix_y)
matrix_z = np.array([[sum_yx2, sum_x3,sum_x2],
                     [sum_yx, sum_x2, sum_x],
                     [sum_y, sum_x, an]])
third_det = np.linalg.det(matrix_z)
a = third_det/det
b = second_det/det
c = first_det/det
func = []
for i in x:
    func.append(a*i*i+b*i+c)
plt.figure(figsize=(10,6))
plt.figure(1)
plt.title("Зависимость времени сортировки от размера массива")
plt.plot(x,func,color="red",linewidth=4)
plt.scatter(x, time_min,s=3)
plt.xlabel("Размер массива\n Коэффициент парной корреляции равен:\n"+str(correlation(x,time_min)))
plt.legend(["y="+str(a)+"*x^2+("+str(b)+") *x"+str(c)])
plt.ylabel("Время сортировки массива")
plt.show()

```

Вывод

Время сортировки пузырьком растет квадратично по закону $O(n^2)$, где n - размер массива, поэтому данный метод не подходит для сортировки больших массивов.