

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:

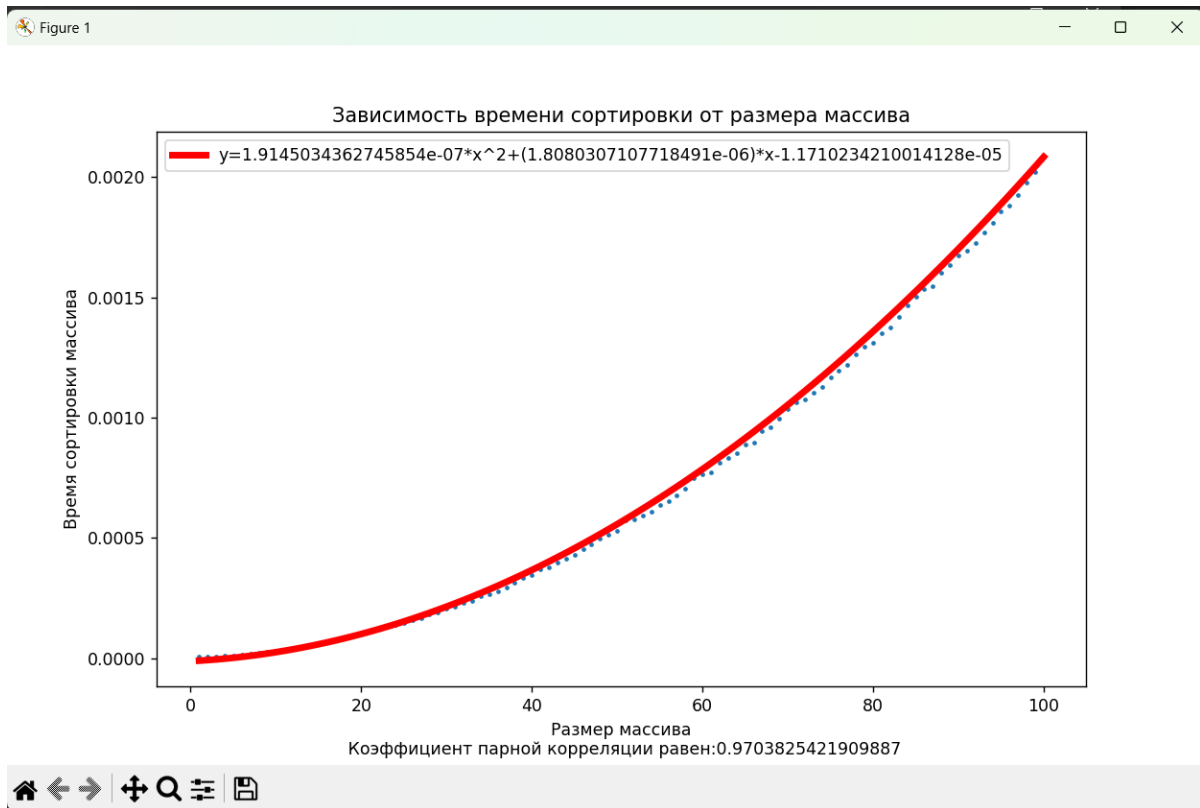


Рисунок 1 – График зависимости времени сортировки от размера массива

Код программы:

```
from random import randint
import random
import numpy as np
import timeit
import matplotlib.pyplot as plt
from math import sqrt
def BubbleSort(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 corr(arrX, arrY):
    sigma1=0
    sigma2=0
    sigma3=0
    xSred=0
    ySred=0
    sumX=0
    sumY=0
    for i in range(len(arrX)):
        sumX+=arrX[i]
        sumY+=arrY[i]
    xSred=sumX/len(arrX)
    ySred=sumY/len(arrY)
    for i in range(len(arrX)):
        sigma1+=(arrX[i]-xSred)*(arrY[i]-ySred)
        sigma2+=(arrX[i]-xSred)**2
```

```

        sigma3+=(arrY[i]-ySred)**2
    return sigma1/(sqrt(sigma2)*sqrt(sigma3))
def minElement(arr):
    temp=arr[0]
    for i in range(0,len(arr)):
        if arr[i]<temp:
            temp=arr[i]
    return temp
arrTimeMin=[]
x=[]
for i in range(1,101):
    arr2=[0 for i in range(0,i)]
    x.append(i)
    for j in range(0,len(arr2)):
        arr2[j]= random.randint(500,1000)
    timePoisk=(timeit.timeit(lambda: BubbleSort(arr2), number=50))/50
    print()
    print("Время сортировки массива из ",i," элементов: ",timePoisk)
    print(arr2)
    arrTimeMin.append(timePoisk)
    print()
sumY=sum(arrTimeMin)
sumX=sum(x)
sumX2=0
sumX3=0
sumX4=0
sumYX2=0
sumYX=0
for i in range(0,len(x)):
    sumX2+=i*i
    sumX3+=i*i*i
    sumX4+=i*i*i*i
    sumYX2+=arrTimeMin[i]*i*i
    sumYX+=arrTimeMin[i]*i
an=len(x)
matrix = np.array([[sumX4, sumX3,sumX2],
                   [sumX3, sumX2, sumX],
                   [sumX2, sumX, an]])
det = np.linalg.det(matrix)
matrixXkramer=np.array([[sumX4, sumX3,sumYX2],
                        [sumX3, sumX2, sumYX],
                        [sumX2, sumX, sumY]])
det1=np.linalg.det(matrixXkramer)
matrixYkramer=np.array([[sumX4, sumYX2,sumX2],
                        [sumX3, sumYX, sumX],
                        [sumX2, sumY, an]])
det2=np.linalg.det(matrixYkramer)
matrixZkramer=np.array([[sumYX2, sumX3,sumX2],
                        [sumYX, sumX2, sumX],
                        [sumY, sumX, an]])
det3=np.linalg.det(matrixZkramer)
a=det3/det
b=det2/det
c=det1/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, arrTimeMin,s=3)
plt.xlabel('Размер массива\n Коэффициент парной корреляции  
равен: '+str(corr(x,arrTimeMin)))
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.