Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №5**

**«ИССЛЕДОВАНИЕ АЛГОРИТМОВ СОРТИРОВКИ»**

**ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

Выполнил: студент учебной группы

ИСПк-203-52-00

Поскребышев Роман Алексеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

**Цель работы**: получить сведенья о наиболее известных алгоритмах сортировки, изучить принципы работы с текстовыми файлами.

**Задание:**

1.Реализовать сортировку данных с помощью алгоритма подсчетом

2.Реализовать сортировку данных с помощью поразрядного алгоритма

3.В обоих случаях необходимо предусмотреть возможность изменения компаратора

4.считывание и вывод данных необходимо производить из текстового файла

5.для демонстрации работы программных реализаций самостоятельно подготовить варианты входных данных

**Код программы 1**

import math

import os

clear = lambda: os.system('cls')

clear()

import numpy as np

import matplotlib.pyplot as plt

import time

def countingSortForRadix(inputArray, placeValue):

countArray = [0] \* 10

inputSize = len(inputArray)

for i in range(inputSize):

placeElement = (inputArray[i] // placeValue) % 10

countArray[placeElement] += 1

for i in range(1, 10):

countArray[i] += countArray[i-1]

outputArray = [0] \* inputSize

i = inputSize - 1

while i >= 0:

currentEl = inputArray[i]

placeElement = (inputArray[i] // placeValue) % 10

countArray[placeElement] -= 1

newPosition = countArray[placeElement]

outputArray[newPosition] = currentEl

i -= 1

return outputArray

def radixSort(inputArray,z):

maxEl = max(inputArray)

D = 1

while maxEl > 0:

maxEl /= 10

D += 1

placeVal = 1

outputArray = inputArray

while D > 0:

outputArray = countingSortForRadix(outputArray, placeVal)

placeVal \*= 10

D -= 1

if z==1:

outputArray.reverse()

return outputArray

def sorting(x,z):

time1=time.perf\_counter()

sorted = radixSort(x,z)

time2=time.perf\_counter()

return(time2-time1),sorted

def countSort(x,z):

a=[]

for i in range(0,len(x)):

a.append (int(x[i]))

time1=time.perf\_counter()

cnt = [0] \* (max(a) + 1)

for item in a:

cnt[item] += 1

result = [num for num, count in enumerate(cnt) for i in range(count)]

if z==1:

result.reverse()

time2=time.perf\_counter()

return (time2-time1),result

def conv():

f=open('text.txt')

y=f.read()

y=y.split(' ')

x=[]

for i in range(len(y)):

x.append (int(y[i]))

x1=x

x2=x

print("вам нужна сортировка по убыванию\?(1-да,0-нет)")

z = input('введите свой выбор:\n')

z = int(z)

s1,o=countSort(x1,z)

s2,p=sorting(x2,z)

print('скорость сортировки подсчетом=',s1)

print(o)

print('скорость сортировки с помощью поразрядного алгоритма=',s2)

print(p)

print("вам нужна оценка погрешности?(1-да,0-нет)")

choice = input('введите свой выбор:\n')

choice = int(choice)

if choice == 1: print(abs(s1-s2))

print('вы хотите использовать программу ещё раз?(1/0)')

p=int(input())

if p==1:

clear()

conv()

else:

clear()

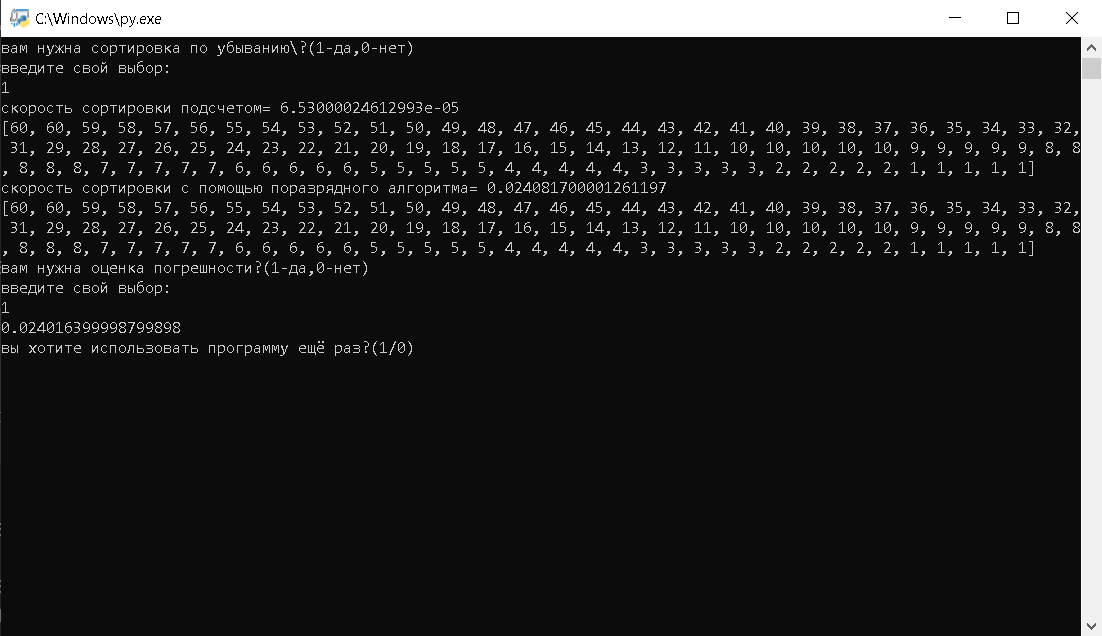
print('спасибо воспользовались этой программой')

input()

f.close()

conv()

**Результат выполнения программы 1**



**Рисунок 2 - Вывод данных**

**Вывод**

Исходя из общего положения и пользуясь в качестве довода и логического обоснования совокупность ранее упомянутых знаний можно сделать закономерный вывод, что в ходе работы мы использовали ранее изученные знания, восстановили забывшееся и узнали, как работать с текстовыми файлами. После проведенной домашней контрольной работы мы узнали, как создавать и работать с файлами, используя функции для работы с файлами. Функции для работы с файлами-это встроенные функции, которые позволяют создавать и работать с файлами. Таким образом, используя Python и встроенные функции для работы с файлами, мы можем использовать большие массивы данных. После проведенной лабораторной работы мы можем смело сказать, что полученные знания помогли быстрее и грамотнее справиться с лабораторной работой №5 и написать этот отчет.