

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
Одеський національний політехнічний університет
Інститут Комп'ютерних Систем
Кафедра Інформаційних Технологій

Протокол лабораторної роботи №4
з дисципліни Численные методы

Виконав студент групи

АД-181

Батура М.Ю.

Прийняв

Рудниченко Н.Д.

Одеса, 2020

Оглавление

Практическая часть.....	3
Порядок выполнения работы:.....	3
Вывод:.....	5

Практическая часть

Задание:

1. Матрица размером $m \times n$ заполняется случайными положительными числами. Найти строку с наибольшим стандартным отклонением элементов. Определить вспомогательные функции:

- заполнение матрицы случайными числами;
- красивый вывод;
- нахождение стандартного отклонения элементов списка.

Примечание. В математической статистике стандартным отклонением выборки X длины n называется величина

$$\sigma_X = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2}, \quad \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i.$$

2. Два простых числа называются «близнецами», если они отличаются друг от друга на 2 (таковы, например, числа 41 и 43). Используя функцию `is_prime()` (проверка числа на простоту, см. конспект), напечатать все пары чисел-близнецов, не превышающих 200.

Порядок выполнения работы:

Task1:

```
from functionsTask1 import *

matrix = get_matrix(5, 5)

output(matrix)

average = get_standart_deviation(matrix)

print_average(average)
```

Task2:

```
def is_simple(num):
    d = 2
    while num % d != 0:
        d += 1
    return d == num

def print_geminy(array_of_nums):
    previous_num = arr_of_simple_nums[0]
    for num in arr_of_simple_nums:
        if (num - previous_num) == 2:
            print("Geminy numbers: "+str(previous_num)+" "+str(num))
        previous_num = num

arr_of_simple_nums = []
for num in range(2, 200):
    if is_simple(num):
        arr_of_simple_nums.append(num)

print_geminy(arr_of_simple_nums)

Functions:

import random

def get_matrix(m, n):
    matrix = []
    for rows in range(0, m):
        matrix.append([])
        for columns in range(0, n):
            matrix[rows].append(round(random.uniform(0, 50)))
    return matrix

def output(matrix):
    for rows in range(0, len(matrix)):
        print(*str(matrix[rows]))

def print_average(matrix):
    for num in range(0, len(matrix)):
        print("Среднее отклонение строки №"+str(num)+" : "+str(matrix[num]))

def get_standart_deviation(matrix):
    # average_in_row = 0
```

```
average_of_rows = []
for rows in range(0, len(matrix)):
    x_dash = 0
    average_in_row = 0
    for cols in range(0, len(matrix[rows])):
        average_in_row += matrix[rows][cols]
    x_dash = average_in_row/len(matrix[rows])
    powed_average_of_row = 0
    for num in range(0, len(matrix[rows])):
        powed_average_of_row += (matrix[rows][num] - x_dash)**2
    average_of_rows.append(powed_average_of_row/len(matrix[rows]))

return average_of_rows
```

Вывод:

В ходе лабораторной работы получены навыки работы с языком Python.

GitHub: <https://github.com/bet9nz9/Python>