Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №2**

Дисциплина: «Разработка профессиональных приложений»

Двумерные структуры данных

Вариант № 5

Выполнил:

студент группы ИВТАПбд-21

Гаязов Т. Р.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

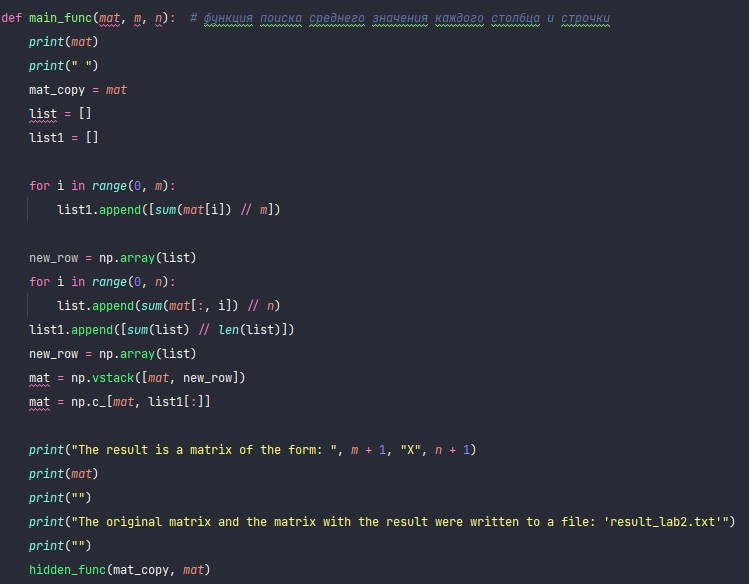
Необходимо реализовать 5 вариант задания, а именно: обработать матрицу по варианту. Выполнить обработку элементов прямоугольной матрицы A, имеющей N строк и M столбцов. Определить средние значения по всем строкам и столб- цам матрицы. Результат оформить в виде матрицы из N + 1 строк и M + 1 столбцов.

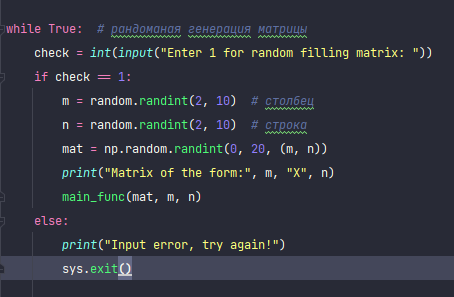
**Требования к программе:**

1. Ввод элементов списка должен быть доступен путем автоматической генерации. Необходимо использовать библиотеку numpy. Результаты выполнения должны сохраняться в файл (исходные данные и результат обработки).
2. Исходный код должен быть откомментирован
3. Необходимо реализовать правильную декомпозицию программы на методы.

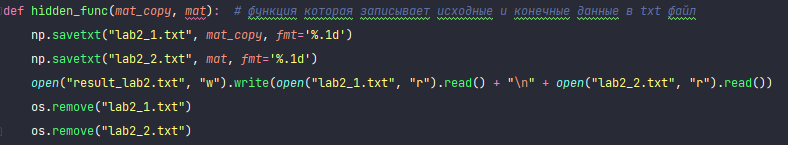
**Описание реализации**

**Функция обработки матрицы:**

****

****

С помощью random.randint заполняем матрицу с рандомным количеством строк и столбцов. С помощью цикла for пробегаемся сначала по строкам, вычисляем среднее значение каждой. Затем по столбцам вычисляем среднее значение каждого столбца. Данные значения заносим в списки, после чего эти списки склеиваем с исходной матрицей благодаря методам библиотеки numpy 'vstack' – для склеивания строки и 'c\_' – для склеивания столбца. После чего результат выводим в консоль и записываем 2 матрицы в txt файл благодаря написанной функции hidden\_func.

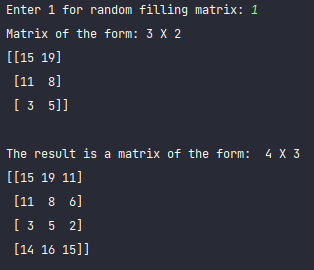


**Описание возникших затруднений**

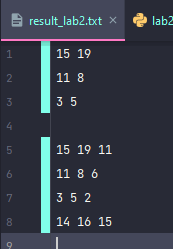
В результате работы с библиотекой numpy возникли трудности с добавлением средних значений в матрицу в виде новых строки и столбца. После изучения данного вопроса на просторах интернета было принято решение добавлять данные в списки и производить склеивание с исходной матрицей благодаря функциям библиотеки numpy 'vstack' и 'c\_'.

Так же возникли трудности с добавлением одновременно 2х матриц в txt файл.  
Так как при повторном вызове метода savetxt, файл полностью отчищался и записывалась только матрица с результатом. После некоторого времени было придумано не стандартное решение проблемы. Заносить отдельно каждую матрицу в отдельный файл и затем в финальный файл записывать содержимое 1+2 файлов. Созданные файлы (1, 2) удаляются благодаря методу remove библиотеки os.

**Результаты работы программы**

****

**1 рис. Вывод двух матриц в консоле**

****

**2 рис. Вывод двух матриц в одном .txt файле**

**Код программы**

*# UTF-8*import numpy as np  
import sys  
import os  
import random  
  
  
def hidden\_func(*mat\_copy*, *mat*): *# функция которая записывает исходные и конечные данные в txt файл* np.savetxt("lab2\_1.txt", *mat\_copy*, *fmt*='%.1d')  
 np.savetxt("lab2\_2.txt", *mat*, *fmt*='%.1d')  
 *open*("result\_lab2.txt", "w").write(*open*("lab2\_1.txt", "r").read() + "\n" + *open*("lab2\_2.txt", "r").read())  
 os.remove("lab2\_1.txt")  
 os.remove("lab2\_2.txt")  
  
  
def main\_func(*mat*, *m*, *n*): *# функция поиска среднего значения каждого столбца и строчки  
 print*(*mat*)  
 *print*(" ")  
 mat\_copy = *mat* list = []  
 list1 = []  
  
 for i in *range*(0, *m*):  
 list1.append([*sum*(*mat*[i]) // *m*])  
  
 new\_row = np.array(list)  
 for i in *range*(0, *n*):  
 list.append(*sum*(*mat*[:, i]) // *n*)  
 list1.append([*sum*(list) // *len*(list)])  
 new\_row = np.array(list)  
 mat = np.vstack([*mat*, new\_row])  
 mat = np.c\_[*mat*, list1[:]]  
  
 *print*("The result is a matrix of the form: ", *m* + 1, "X", *n* + 1)  
 *print*(*mat*)  
 *print*("")  
 *print*("The original matrix and the matrix with the result were written to a file: 'result\_lab2.txt'")  
 *print*("")  
 hidden\_func(mat\_copy, *mat*)  
  
  
while True: *# рандоманая генерация матрицы* check = *int*(*input*("Enter 1 for random filling matrix: "))  
 if check == 1:  
 m = random.randint(2, 10) *# столбец* n = random.randint(2, 10) *# строка* mat = np.random.randint(0, 20, (m, n))  
 *print*("Matrix of the form:", m, "X", n)  
 main\_func(mat, m, n)  
 else:  
 *print*("Input error, try again!")  
 sys.exit()