**Урок No13. Двумерные списки.**

1.

# Задание No1

# С помощью цикла создайте матрицу вида 10x10 (пример):

# [[0, -2, -5, -5, 0, -4, -1, -23, -11, -20],

# [-4, -6, 16, 9, -8, 2, 10, -32, -18, -13],

# [21, 7, -6, -3, 34, -7, -23, 25, 27, 38],

# [7, 29, 38, 19, 55, 48, 46, 49, -8, -46],

# [80, -3, 76, 46, 72, -33, 14, 0, 71, -20],

# [82, 95, 23, -6, 1, 70, 50, 16, 25, 7],

# [52, 36, -25, -35, 18, -36, 93, 11, -44, 9],

# [114, 108, 33, 61, -4, 131, 34, -26, 47, -17],

# [37, 56, 135, -12, 34, 67, -17, 14, 93, -67],

# [-4, 23, 175, -44, 30, -38, 89, 159, 28, 178]]

# И ещё одну - такой же размерности. Числа в матрице выше приведены в

# качестве примера (одна из генераций)

# Итого у вас должно получиться сперва две матрица одинаковой размерности

# И теперь вам нужно сложить эти две матрицы в третью. Формулу сложения

# матриц, вы можете найти в интернете, либо посетив этот ресурс

# Итак. Вы должны создать две матрицы

# Чтобы заполнить матрицы различными значениями - воспользуйтесь

# модулем random

# Для подключения модуля random в код используйте строчку, которую нужно

# поместить в самое начало вашего скрипта

# import random

# Для справки:

# Random. Официальная документация по модулю (англ.яз)

# Random. Информация на русском языке

# Для начала вы можете просто создать переменные matrix\_1 и matrix\_2

# matrix\_1 = [[0, -2, -1, -6, -6, 0, -9, -8, -30, -9],

# [5, 12, 4, -16, -4, -9, -16, -15, 1, -26],[13, 39, 14, 23, -4, 40, 32, 6, -8, 23],

# [13, -8, 34, 49, 30, 18, 47, 11, -24, 11],

# [21, 73, 71, 61, -1, 79, -34, 22, 69, 67],

# [75, 25, 25, 39, 100, -12, -21, 81, -10, 87],

# [81, 63, 102, 104, 53, -44, 71, -36, -36, -9],

# [7, 98, 26, -3, 128, 94, 18, -26, 14, 21],

# [65, 128, 80, 124, 27, -32, 73, 59, 19, 34],

# [43, 111, 38, 149, 5, 112, 79, 53, 15, 92]]

# matrix\_2 = [[0, 4, 6, 11, 15, 6, 9, 26, 15, 21],

# [-5, 4, -15, -9, -4, 2, -8, 19, -4, -1],

# [-2, -39, -19, 14, 22, 5, -34, 15, 16, -9],

# [-22, -52, 11, -11, -3, 16, -11, -6, -32, -2],

# [-61, -47, -5, -58, 16, -13, 28, -36, -64, 2],

# [-29, 23, 19, 2, -14, -87, 7, -88, 39, 7],

# [-6, 18, -97, 26, -64, 0, -72, -34, -68, -92],

# [-120, -117, -72, -129, -139, 16, -61, 36, -137, -29],

# [-112, -83, 7, -119, -132, -129, -143, -154, -23, -34],

# [32, -67, -75, -92, 15, -163, 18, 31, -162, -16]]

# И теперь нужно придумать, как их сложить, чтобы получить matrix\_3:

# matrix\_3 = [[0, 2, 5, 5, 9, 6, 0, 18, -15, 12],

# [0, 16, -11, -25, -8, -7, -24, 4, -3, -27],

# [11, 0, -5, 37, 18, 45, -2, 21, 8, 14],

# [-9, -60, 45, 38, 27, 34, 36, 5, -56, 9],

# [-40, 26, 66, 3, 15, 66, -6, -14, 5, 69],

# [46, 48, 44, 41, 86, -99, -14, -7, 29, 94],

# [75, 81, 5, 130, -11, -44, -1, -70, -104, -101],

# [-113, -19, -46, -132, -11, 110, -43, 10, -123, -8],

# [-47, 45, 87, 5, -105, -161, -70, -95, -4, 0],

# [75, 44, -37, 57, 20, -51, 97, 84, -147, 76]]

# Потренируйтесь сперва в сложении матриц, а затем попробуйте с помощью

# модуля random сгенерировать матрицы подобных размеров и уже на них

# применить навыки сложения matrix\_1 и matrix\_2

# Задание считается выполненным, если вы напишите алгоритм, который будет

# уметь как складывать матрицы, так и генерировать матрица различных

# размерностей. Будь то матрицы 10х10 или 4х3

#

# https://github.com/A-l-E-v/PySynergy/blob/main/U-13/sum\_m.py

#

import random

test\_matrix\_1 = [[0, -2, -1, -6, -6, 0, -9, -8, -30, -9],

[5, 12, 4, -16, -4, -9, -16, -15, 1, -26],[13, 39, 14, 23, -4, 40, 32, 6, -8, 23],

[13, -8, 34, 49, 30, 18, 47, 11, -24, 11],

[21, 73, 71, 61, -1, 79, -34, 22, 69, 67],

[75, 25, 25, 39, 100, -12, -21, 81, -10, 87],

[81, 63, 102, 104, 53, -44, 71, -36, -36, -9],

[7, 98, 26, -3, 128, 94, 18, -26, 14, 21],

[65, 128, 80, 124, 27, -32, 73, 59, 19, 34],

[43, 111, 38, 149, 5, 112, 79, 53, 15, 92]]

test\_matrix\_2 = [[0, 4, 6, 11, 15, 6, 9, 26, 15, 21],

[-5, 4, -15, -9, -4, 2, -8, 19, -4, -1],

[-2, -39, -19, 14, 22, 5, -34, 15, 16, -9],

[-22, -52, 11, -11, -3, 16, -11, -6, -32, -2],

[-61, -47, -5, -58, 16, -13, 28, -36, -64, 2],

[-29, 23, 19, 2, -14, -87, 7, -88, 39, 7],

[-6, 18, -97, 26, -64, 0, -72, -34, -68, -92],

[-120, -117, -72, -129, -139, 16, -61, 36, -137, -29],

[-112, -83, 7, -119, -132, -129, -143, -154, -23, -34],

[32, -67, -75, -92, 15, -163, 18, 31, -162, -16]]

test\_matrix\_3 = [[0, 2, 5, 5, 9, 6, 0, 18, -15, 12],

[0, 16, -11, -25, -8, -7, -24, 4, -3, -27],

[11, 0, -5, 37, 18, 45, -2, 21, 8, 14],

[-9, -60, 45, 38, 27, 34, 36, 5, -56, 9],

[-40, 26, 66, 3, 15, 66, -6, -14, 5, 69],

[46, 48, 44, 41, 86, -99, -14, -7, 29, 94],

[75, 81, 5, 130, -11, -44, -1, -70, -104, -101],

[-113, -19, -46, -132, -11, 110, -43, 10, -123, -8],

[-47, 45, 87, 5, -105, -161, -70, -95, -4, 0],

[75, 44, -37, 57, 20, -51, 97, 84, -147, 76]]

def fill\_matrix (matrix):

for i in range (len(matrix)):

for j in range (len(matrix[0])):

matrix[i][j] = random.randint(-499, 499)

return

# функция вывода матрицы любой размерности.

def print\_matrix (matrix):

print()

for i in range (len(matrix)):

for j in range (len(matrix[0])):

print (f' {matrix[i][j]:4}', end='')

print()

print()

return

# функция суммирования двух матриц одинаковой размерности. Проверки размерности нет.

def sum\_matrix (m1,m2):

print()

m=len(m1[0])

n=len(m1)

# инициализируем пустую новую матрицу заполненную нулями

m3 = [ [0]\*m for i in range(n) ]

for i in range (n):

for j in range (m):

m3[i][j]=m1[i][j]+m2[i][j]

return m3

print()

print('--- Генерация и сложение матриц ---')

print()

print ('Выполним тест.')

print ('Допустим имеется матрица 1 размерностью 10х10:')

print\_matrix (test\_matrix\_1)

print ('и матрица 2 размерностью 10х10:')

print\_matrix (test\_matrix\_2)

print (', то результатом сложения двух матриц должна быть матрица: ')

print\_matrix (test\_matrix\_3)

print()

print ('Выполнив сложение, получаем третью матрицу. Сравните с тестовой результирующей матрицей выше:')

matrix\_3 = sum\_matrix(test\_matrix\_1,test\_matrix\_2)

print\_matrix (matrix\_3)

print()

print('Выполним реальный тест генерации и сложения матриц.')

print()

print ('Создаём и заполняем две матрицы размерностью m x n.')

m=int(input('Введите количество строк m: '))

n=int(input('Введите количество столбцов n: '))

print()

# инициализируем обе матрицы

matrix\_1 = [ [0]\*n for i in range(m) ]

matrix\_2 = [ [0]\*n for i in range(m) ]

# заполняем обе матрицы случайными числами

fill\_matrix (matrix\_1)

fill\_matrix (matrix\_2)

# выводим на экран обе матрицы

print ('Первая сгенерированная матрица:')

print\_matrix (matrix\_1)

print ('Вторая сгенерированная матрица:')

print\_matrix (matrix\_2)

# проводим сложение матриц

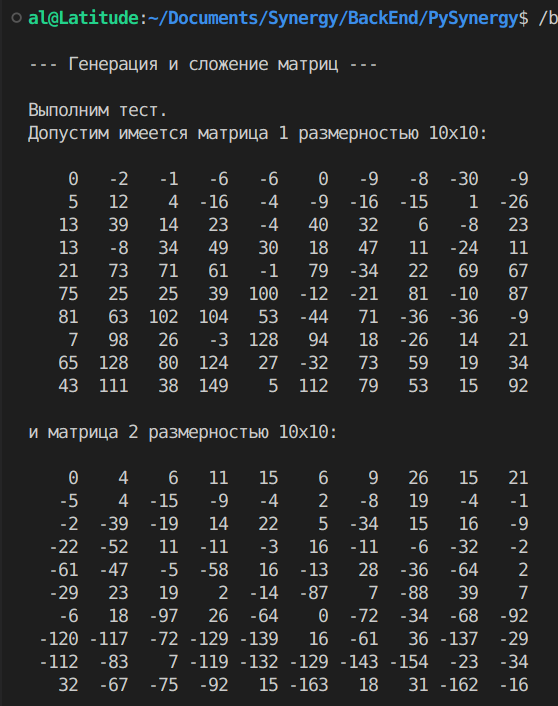
matrix\_3 = sum\_matrix (matrix\_1,matrix\_2)

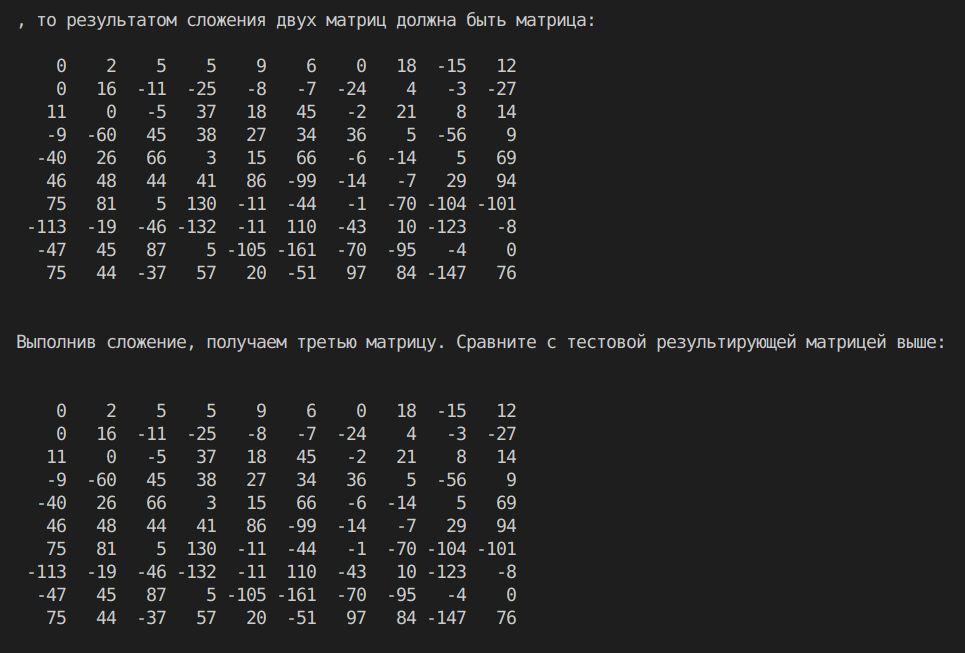
# выводим на экран третью матрицу

print ('Суммарная матрица:')

print\_matrix (matrix\_3)

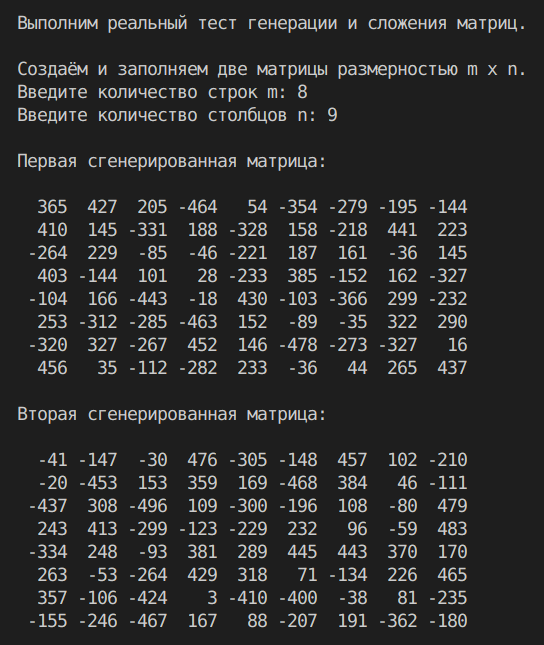
Испытаю программу на тестовых матрицах из задания:

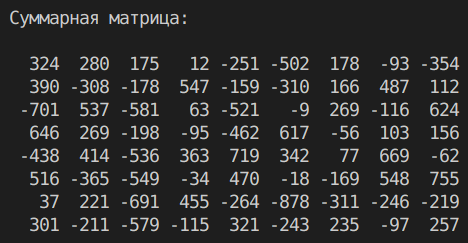




Результаты сложения совпали с тестовыми, значит программа работает правильно.

Испытаю на матрицах разных размеров, например 8х9:





И на размерности, где количество строк больше чем столбцов, например, 12 на 4:

