

ФГБОУ ВО «Московский Политехнический университет»

Лабораторная работа №5

Двумерные массивы и функции

Задание 1

Вариант №27

По дисциплине:

Основы Программирования

Выполнил Шукуров Ф.Ф группа 181-362

Проверил _____ Никишина И.Н.

Москва 2018

Лабораторная работа №5

задание:

Даны две матрицы одного порядка $M \times N$ (M строк \times N столбцов).
Написать программу сложения, вычитания и транспонирования матриц.

1. Сложение и вычитание: $C_{ij} = a_{ij} \pm b_{ij}$
2. Транспонирование $b_{ij} = a_{ji}$

Описание программы:

Программа была написана на python 3.6, реализованна в среде os Linux, отвечает за ввод данных, вычисление и вывод данных на экран. Был импортирован ранее установленный модуль `numpy`, а так же, `random`, `math`, для проверки пользовательского ввода был использован блок `try→except`

Описание Алгоритма:

1. Импортируем все функции, создаем блок `try→except`.
2. с помощью метода `input()` а так же присваивания его к вещественному значению создаем «m» и «n».
3. с помощью модуля «numpy» (далее «np»), присваиваем к двум матрицам `matrix_a` и `matrix_b` случайные числа в диапазоне `[-100;100]` размерностью указанной ранее пользователем «n» и «m»
4. спрашиваем пользователя какой вид математических действий будем совершать над матрицами. Ответ присваиваем к переменной «output»
5. Создав блок исключений (`if→elif→else`) и задав необходимые условия, выполняем те или иные действия.
6. Для транспонирования матрицы используем функцию `np→transpose(matrix_a)`
7. Вывод ответа на экран
8. В случае ошибки метод `except` выводит информацию о ошибке ввода. (`except → print('Ты не прав.')`)

Листинг Программы:

```
# -*- coding: utf-8 -*-
import numpy as np
import random, math
try:
    sum_or_subtraction = None
    m = int ( input( 'Укажите количество столбцов: \n' ))
    n = int ( input( 'Укажите количество строк: \n' ))
    matrix_a = np.random.randint(100, size=(n,m))
    matrix_b = np.random.randint(100, size=(n,m))
```

```

print('Задание№1\n_____ \nСложение или вычитание матриц?')
output = str(input('\n1)+\n2)-\n'))
if output == "1" or output == "+":
    matrix_c = matrix_a + matrix_b
    sum_or_subtraction = " \n+\n "
elif output == "2" or output == "-":
    matrix_c = matrix_a - matrix_b
    sum_or_subtraction = " \n-\n "
print("_____ \n"+ str(matrix_a) + "\n"+ sum_or_subtraction
transpose_matrix_a = np.transpose(matrix_a)
print(str(matrix_a) + "\n\n" + str(transpose_matrix_a))
except:
    print('Ты не прав.')

```

Результат программы:

Укажите количество столбцов:

10

Укажите количество строк:

10

Задание№1

Сложение или вычитание матриц?

1)+

2)-

-

```

[[98 13 48 44  2 39  7 76 72 41]
 [59 40 78 95 78 98 72 28 99 69]
 [42 72 90 63 16 82 91 67 66 33]
 [17 91 24 34 76 39 40 37 14 44]
 [12 31 66 53 98 71 64  0 91 11]
 [76 55  6 82 39 90 59 14  1 60]
 [ 6 91 12 36 80 81 98 22 95 98]
 [51 93 63 91 72 88 47 29 91 21]
 [76 10 52 29  1 25 10 68  6 90]
 [ 8  5 38 51 13 88 91 49 92 12]]

```

-

```

[[79 30 22 13 82 86 74  9 41 81]

```

```

[23 81 80 70 26 6 27 99 6 93]
[42 3 50 50 97 73 48 94 64 91]
[ 4 72 77 62 16 43 67 45 67 56]
[69 14 50 16 97 40 87 92 12 4]
[35 55 42 54 35 29 33 86 9 29]
[27 21 92 36 69 17 12 33 49 63]
[ 0 47 5 94 94 25 11 29 50 24]
[98 0 28 18 32 72 57 23 74 99]
[32 68 20 2 41 93 64 43 30 8]]
=
[[ 19 -17 26 31 -80 -47 -67 67 31 -40]
[ 36 -41 -2 25 52 92 45 -71 93 -24]
[ 0 69 40 13 -81 9 43 -27 2 -58]
[ 13 19 -53 -28 60 -4 -27 -8 -53 -12]
[-57 17 16 37 1 31 -23 -92 79 7]
[ 41 0 -36 28 4 61 26 -72 -8 31]
[-21 70 -80 0 11 64 86 -11 46 35]
[ 51 46 58 -3 -22 63 36 0 41 -3]
[-22 10 24 11 -31 -47 -47 45 -68 -9]
[-24 -63 18 49 -28 -5 27 6 62 4]]

```

----- Задание №2

```

[[98 13 48 44 2 39 7 76 72 41]
[59 40 78 95 78 98 72 28 99 69]
[42 72 90 63 16 82 91 67 66 33]
[17 91 24 34 76 39 40 37 14 44]
[12 31 66 53 98 71 64 0 91 11]
[76 55 6 82 39 90 59 14 1 60]
[ 6 91 12 36 80 81 98 22 95 98]
[51 93 63 91 72 88 47 29 91 21]
[76 10 52 29 1 25 10 68 6 90]
[ 8 5 38 51 13 88 91 49 92 12]]

[[98 59 42 17 12 76 6 51 76 8]
[13 40 72 91 31 55 91 93 10 5]
[48 78 90 24 66 6 12 63 52 38]
[44 95 63 34 53 82 36 91 29 51]
[ 2 78 16 76 98 39 80 72 1 13]
[39 98 82 39 71 90 81 88 25 88]
[ 7 72 91 40 64 59 98 47 10 91]
[76 28 67 37 0 14 22 29 68 49]
[72 99 66 14 91 1 95 91 6 92]
[41 69 33 44 11 60 98 21 90 12]]

```

