

## Лабораторная работа №8

*Матрицы (двумерный массив).*

### Создание, вывод и ввод матрицы в Питоне

Для работы с матрицами в Python также используются списки. Каждый элемент списка-матрицы содержит вложенный список.

Таким образом, получается структура из вложенных списков, количество которых определяет количество строк матрицы, а число элементов внутри каждого вложенного списка указывает на количество столбцов в исходной матрице.

Рассмотрим пример матрицы размера 4 x 3:

```
matrix = [[-1, 0, 1],  
          [-1, 0, 1],  
          [0, 1, -1],  
          [1, 1, -1]]
```

Данный оператор можно записать в одну строку:

```
matrix = [[-1, 0, 1], [-1, 0, 1], [0, 1, -1], [1, 1, -1]]
```

Вывод матрицы можно осуществить одним оператором, но такой простой способ не позволяет выполнять какой-то предварительной обработки элементов:

```
print(matrix)
```

Результат:

```
[[[-1, 0, 1], [-1, 0, 1], [0, 1, -1], [1, 1, -1]]]
```

Для вывода матрицы в виде таблицы можно использовать специально заготовленную для этого процедуру:

#### 1. способ:

```
def printMatrix ( matrix ):  
    for i in range ( len(matrix) ):  
        for j in range ( len(matrix[i]) ):  
            print ( "{:4d}".format(matrix[i][j]), end = "" )  
        print ()
```

В примере *i* – это номер строки, а *j* – номер столбца;  
*len(matrix)* – число строк в матрице.

2 способ:

```
def printMatrix ( matrix ):
    for row in matrix:
        for x in row:
            print ( "{:4d}".format(x), end = "" )
            print ()
```

Внешний цикл проходит по строкам матрицы (row), а внутренний цикл проходит по элементам каждой строки (x).

Для инициализации элементов матрицы случайными числами используется алгоритм:

```
import random
for i in range(N):
    for j in range(M):
        matrix[i][j] = random.randint ( 30, 60 )
        print ( "{:4d}".format(matrix[i][j]), end = "" )
        print()
```

### Обработка элементов двумерного массива

Нумерация элементов двумерного массива, как и элементов одномерного массива, начинается с нуля. Т.е. matrix[2][3] — это элемент третьей строки четвертого столбца.

Пример обработки элементов матрицы:

Найти произведение элементов двумерного массива.

*Решение:*

```
p = 1
for i in range(N):
    for j in range(M):
        p *= matrix[i][j]
    print (p)
```

**Пример:** Найти сумму элементов двумерного массива.

*Решение:*

Более подходящий вариант для Python:

```
s = 0
for row in matrix:
    s += sum(row)
```

**print** (s)

Для поиска суммы существует стандартная функция `sum`.

**Задание Python 8\_0:** Получены значения температуры воздуха за 4 дня с трех метеостанций, расположенных в разных регионах страны:

Номер станции	1-й день	2-й день	3-й день	4-й день
1	-8	-14	-19	-18
2	25	28	26	20
3	11	18	20	25

Т.е. запись показаний в двумерном массиве выглядела бы так:

```
t[1,1]:=-8;          t[1,2]:=-14;          t[1,3]:=-19;          t[1,4]:=-18;
t[2,1]:=25;          t[2,2]:=28;          t[2,3]:=26;          t[2,4]:=20;
t[3,1]:=11;          t[3,2]:=18;          t[3,3]:=20;          t[3,4]:=25;
```

Распечатать температуру на 2-й метеостанции за 4-й день и на 3-й метеостанции за 1-й день.

Распечатать показания термометров всех метеостанций за 2-й день.

Определить среднюю температуру на 3-й метеостанции.

Распечатать, в какие дни и на каких метеостанциях температура была в диапазоне 24-26 градусов тепла.

**Задание Python 8\_1:** Написать программу поиска минимального и максимального элементов матрицы и их индексов.

**Задание Python 8\_2:** Написать программу, выводящую на экран строку матрицы, сумма элементов которой максимальна.

- Для обработки элементов квадратной матрицы (размером N x N):
- Для элементов главной диагонали достаточно использовать один цикл:

```
for i in range(N):
```

*# работаем с matrix[i][i]*

- Для элементов побочной диагонали:

**for** i **in** range(N):

*# работаем с matrix[i][N-1-i]*

**Пример:** Переставить 2-й и 4-й столбцы матрицы. Использовать два способа.

*Решение:*

1) **for** i **in** range(N):

    c = A[i][2]

    A[i][2] = A[i][4]

    A[i][4] = c

2) **for** i **in** range(N):

    A[i][2], A[i][4] = A[i][4], A[i][2]

**Задание Python 8\_3:** Составить программу, позволяющую с помощью датчика случайных чисел сформировать матрицу размерностью N. Определить:

- минимальный элемент, лежащий ниже побочной диагонали;
- произведение ненулевых элементов последней строки.