

## Лабораторная работа №5: «Двумерные массивы и функции»

### Цель работы:

Дать студентам практический навык в написании программ обработки одномерных массивов: поиск максимумов и минимумов, сортировка.

### Постановка задачи

Дана целочисленная прямоугольная матрица. Определить номер первого из столбцов, содержащих хотя бы один нулевой элемент.

Переставляя строки заданной матрицы, расположить их в соответствии с убыванием характеристик.

ПРИМЕЧАНИЕ: Характеристикой строки целочисленной матрицы назовем сумму ее отрицательных четных элементов.

### Теоретическое введение

#### *Организация двумерных массивов*

В Python реализовать массив можно через вложенные списки. В следующем листинге программы показан прием формирования двумерного списка и способы инициализации его элементов. В этой программе следует обратить внимание на то, что элемент списка, ранее инициализированный числом, получает значение строкового типа и преобразование делается по умолчанию.

### Описание алгоритма

1. Запросить размер массива  $N \times M$ . Форма массива - прямоугольная матрицы ( $N \times M$ ).
2. Изготовить массив - инициализировать набором псевдослучайных вещественных чисел (функция `MakeMatr()`).
3. Вывести полученную матрицу (функция `PrintMatr()`)
4. Вычислить среднее значение и дисперсию (функция `MidlDisp()`).
5. Выполнить корректировку элементов (функция `CorrectMatr()`).
6. Вывести откорректированную матрицу

### Описание входных и выходных данных

Программа запрашивает размер массива. Результат работы программы визуализируется на экране монитора. Тип элементов матрицы задан как вещественный (`float`) в соответствии с заданием.

### Описание подпрограмм

#### *Функция `MakeMatr(n, m, a, b)`*

Функция инициализирует квадратную матрицу размером  $n \times m$  псевдослучайными числами в диапазоне  $[a, b)$ .

Возвращается объект типа `array`.

#### *Функция `PrintMatr (Matr)`*

Служит для вывода массива на экран монитора.

### **Функция *ZeroCol(Mat)***

Служит для вывода массива на экран монитора.

### **Функция *SortMatr(Mat)***

Служит для вывода массива на экран монитора.

### **Функция *CorOut(Mat, nom)***

### **Листинг программы**

```
# -*- coding: UTF-8 -*-
import numpy as np

def MakeMatr(n, m, a, b):
    Matr = (b-a)*np.random.random(size=(n, m)) + a
    return Matr

def PrintMatr(Mat):
    (nRow, nCol) = Matr.shape
    for Row in range(nRow):
        for Col in range(nCol):
            print("{0: 7.3f}".format(Mat[Row][Col]), end="")
        print()
    print()

def ZeroCol(Mat):
    (nRow, nCol) = Matr.shape
    for Col in range(nCol):
        for Row in range(nRow):
            if Matr[Row][Col] == 0:
                return Col
    return -1

def SortMatr(Mat):
    (nRow, nCol) = Matr.shape
    har = []
    nom = []
    for Row in range(nRow):
        har.append(0)
        nom.append(Row)
        for Col in range(1, nCol):
```

```

        if Matr[Row][Col] < 0:
            har[Row] += Matr[Row][Col]
    for i in range(0, nRow-1):
        for j in range(i, nRow):
            if har[i] < har[j]:
                promh = har[i]
                promn = nom[i]
                nom[i] = nom[j]
                har[i] = har[j]
                nom[j] = promn
                har[j] = promh
    return nom

def CorOut(Matr, nom):
    (nRow, nCol) = Matr.shape
    print()
    for i in range(nRow):
        for Col in range(nCol):
            print("{0: 7.3f}".format(Matr[nom[i]][Col]), end="")
        print()

n = int(input('input matrix size N: '))
m = int(input('input matrix size M: '))
MyMatr = MakeMatr(n, m, -10, 10)
PrintMatr(MyMatr)
col = ZeroCol(MyMatr)
if col > -1:
    print("First col with zero element {0}".format(col+1))
else:
    print("No one zero element in matrix")
nom = SortMatr(MyMatr)
CorOut(MyMatr, nom)

```

### **Результат работы программы**

```

input matrix size N: 5
input matrix size M: 6
6.558  5.469  7.406  4.379  5.043  5.596
2.888 -8.401 -7.067 -6.329 -9.895 -9.695
1.098 -5.053  0.982  9.804  2.184  0.796
1.863 -1.556 -9.345  8.396 -8.747 -7.182
5.603  5.824 -0.063 -9.980 -3.291  4.039
No one zero element in matrix

```

6.558	5.469	7.406	4.379	5.043	5.596
1.098	-5.053	0.982	9.804	2.184	0.796
5.603	5.824	-0.063	-9.980	-3.291	4.039
1.863	-1.556	-9.345	8.396	-8.747	-7.182
2.888	-8.401	-7.067	-6.329	-9.895	-9.695

## **Задание к лабораторной работе №5 «Двумерные массивы и функции»**

Размерности двумерных массивов следует запрашивать у пользователя. Все необходимые данные должны передаваться в функции в качестве параметров. Все переменные, используемые только внутри функции, должны быть описаны как локальные. Использование глобальных переменных в функциях не допускается. Обеспечить вывод, как исходного массива, так и массива, полученного в результате работы программы, там, где это возможно по условию задачи. Пункты задания оформить в виде функций.

### **Вариант 1**

Дана целочисленная прямоугольная матрица. Определить:

1. Количество строк, не содержащих ни одного нулевого элемента.
2. Максимальное значение из чисел, встречающихся в заданной матрице более одного раза.

### **Вариант 2**

Дана целочисленная прямоугольная матрица.

1. Определить количество столбцов, не содержащих ни одного нулевого элемента.
2. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.

*ПРИМЕЧАНИЕ.* Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов.

### **Вариант 3**

Дана целочисленная прямоугольная матрица. Определить:

1. Количество столбцов, содержащих хотя бы один нулевой элемент.
2. Номер строки, в которой находится самая длинная серия одинаковых элементов.

### **Вариант 4**

Дана целочисленная квадратная матрица. Определить:

1. Произведение элементов в тех строках, которые не содержат отрицательных элементов.
2. Максимум среди сумм элементов диагоналей, параллельных главной диагонали

### **Вариант 5**

Дана целочисленная квадратная матрица. Определить:

1. Сумму элементов в тех столбцах, которые не содержат отрицательных элементов.
2. Минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.

### **Вариант 6**

Дана целочисленная прямоугольная матрица. Определить:

1. Сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент.
2. Номера строк и столбцов всех седловых точек матрицы.

*ПРИМЕЧАНИЕ.* Матрица  $A$  имеет седловую точку  $A_{ij}$ , если  $A_{ij}$  является минимальным элементом в  $i$ -й строке и максимальным в  $j$ -м столбце.

### **Вариант 7**

Для заданной матрицы размером  $8 \times 8$  найти такие  $k$ , что элементы  $k$ -й строки матрицы совпадают с элементами  $k$ -ого столбца.

Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент.

### **Вариант 8**

Переставляя столбцы заданной матрицы, расположить их в соответствии с ростом характеристик.

Найти сумму элементов в тех столбцах, которые содержат хотя бы один

отрицательный элемент.

**ПРИМЕЧАНИЕ.** Характеристикой столбца целочисленной матрицы назовем сумму модулей его отрицательных нечетных элементов.

### Вариант 9

Соседями элемента  $A_{ij}$  в матрице назовем элементы  $A_{kl}$ , где  $i-1 \leq k \leq i+1$ ,  $j-1 \leq l \leq j+1$ ,  $(k, l) \neq (i, j)$ .

$$\begin{array}{cccccc}
 a_{1,1} & \boxed{a_{1,2}} & \dots & a_{1,j-1} & a_{1,j} & a_{1,j+1} \\
 a_{2,1} & \boxed{a_{2,2}} & \dots & a_{2,j-1} & a_{2,j} & a_{2,j+1} \\
 \\ 
 a_{i-1,1} & a_{i-1,2} & , & \boxed{a_{i-1,j-1}} & \boxed{a_{i-1,j}} & \boxed{a_{i-1,j+1}} \\
 a_{i,1} & a_{i,2} & & \boxed{a_{i,j-1}} & \boxed{a_{i,j}} & \boxed{a_{i,j+1}} \\
 a_{i+1,1} & a_{i+1,2} & & \boxed{a_{i+1,j-1}} & \boxed{a_{i+1,j}} & \boxed{a_{i+1,j+1}}
 \end{array}$$

В этой матрице соседи углового элемента  $a_{11}$  и элемента  $a_{ij}$  выделены полужирным шрифтом.

Операция сглаживания матрицы дает новую матрицу того же размера, каждый элемент которой получается как среднее арифметическое имеющихся соседей соответствующего элемента исходной матрицы.

Построить результат сглаживания заданной вещественной матрицы размером  $10 \times 10$ .

В сглаженной матрице найти сумму модулей элементов, расположенных ниже главной диагонали.

### Вариант 9

Элемент матрицы называется локальным минимумом, если он строго меньше всех имеющихся у него соседей.

Соседями элементами  $A_{ij}$  в матрице назовем элементы  $A_{kl}$ , где  $i-1 \leq k \leq i+1$ ,  $j-1 \leq l \leq j+1$ ,  $(k, l) \neq (i, j)$ .

$$\begin{array}{cccccc}
 \boxed{a_{1,1}} & \boxed{a_{1,2}} & \dots & a_{1,j-1} & a_{1,j} & a_{1,j+1} \\
 \boxed{a_{2,1}} & \boxed{a_{2,2}} & \dots & a_{2,j-1} & a_{2,j} & a_{2,j+1} \\
 \\ 
 a_{i-1,1} & a_{i-1,2} & , & \boxed{a_{i-1,j-1}} & \boxed{a_{i-1,j}} & \boxed{a_{i-1,j+1}} \\
 a_{i,1} & a_{i,2} & & \boxed{a_{i,j-1}} & \boxed{a_{i,j}} & \boxed{a_{i,j+1}} \\
 a_{i+1,1} & a_{i+1,2} & & \boxed{a_{i+1,j-1}} & \boxed{a_{i+1,j}} & \boxed{a_{i+1,j+1}}
 \end{array}$$

В этой матрице соседи углового элемента  $a_{11}$  и элемента  $a_{ij}$  выделены полужирным шрифтом.

Подсчитать количество локальных минимумов заданной матрицы размером  $10 \times 10$ .

Найти сумму модулей элементов, расположенных выше главной диагонали.