PYTHON – Matrices

Introduction

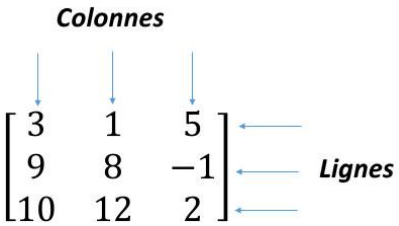
Nous avons vu, dans les leçons précédentes, qu'il était souvent utile de pouvoir définir des structures de données pouvant contenir des collections d'objets d'un même type simple (int, char, float, ...).

Ces listes unidimensionnelles proposent un accès indexé aux différents éléments qu'ils contiennent.

On peut également, dans d'autres cas, avoir besoin de structures de données représentant des tableaux à plusieurs entrées : on appelle, le plus souvent, ces structures des matrices en référence à l'objet mathématique.

Une matrice est une structure de données bidimensionnelle (2D) dans laquelle les nombres sont organisés en lignes et en colonnes.

Par exemple:



Cette matrice est une matrice **3x3** car elle comporte 3 lignes et 3 colonnes.

Déclaration-Initialisation d'une matrice

Python n'a pas de type intégré pour les matrices. Cependant, nous pouvons traiter une **liste de liste** comme une matrice.

Exemple :

```
Matrice = [[3, 1, 5], [9, 8, -1], [10, 12, 2]]
```

```
matValeurs=[[10,7,4,11,3],[0,-1,4,9,12],[8,-18,2,157,63]]
```

Ainsi présentée, une matrice apparait comme une liste dont les éléments sont des listes.

Cette matrice comportera 3 lignes et 5 colonnes (chaque ligne étant composée de 5 éléments).

indice	**0**	**1**	**2**	**3**	**4**
0	10	7	4	11	3
1	0	-1	4	9	12
2	8	-18	2	157	63

Accès aux valeurs contenues dans une matrice

```
matValeurs=[[10,7,4,11,3],[0,-1,4,9,12],[8,-18,2,157,63]]
print(type(matValeurs))
print(matValeurs[0])
print(matValeurs[1])
print(matValeurs[0][0])
print(matValeurs[0][1])
```

```
<class 'list'>
[10, 7, 4, 11, 3]
[0, -1, 4, 9, 12]
10
7
```

indice	0	1	2	3	4
0	10	7	4	11	3
1	0	-1	4	9	12
2	8	-18	2	157	63

Longueur d'une matrice

```
matValeurs=[[10,7,4,11,3],[0,-1,4,9,12],[8,-18,2,157,63]]
print(len(matValeurs))
print(len(matValeurs[0]))
```

```
3
5
```

Boucle sur une matrice

```
matValeurs=[[10,7,4,11,3],[0,-1,4,9,12],[8,-18,2,157,63]]
col3 = [] #liste vide
for ligne in matValeurs :
    print(ligne)
    col3.append(ligne[2]) # ajouter le 3ème élément
print("col3 : ",col3)
```

```
[10, 7, 4, 11, 3]
[0, -1, 4, 9, 12]
[8, -18, 2, 157, 63]
col3 : [4, 4, 2]
```

Initialisation d'une matrice avec une même valeur

la fonction initMatVal(nb_ligne,nb_colonne,valeur) renvoie une matrice possédant *nb_ligne* lignes et *nb_colonne* colonnes dans laquelle tous les éléments possèdent la même valeur *valeur*.

```
#Programme initMatVal.py
def initMatVal(nb_ligne,nb_colonne,valeur):
    #Création d'une matrice vide
    matrice=[None] * nb_ligne
    for i in range(nb_ligne):
        matrice[i]=[None] * nb_colonne
    #Initialisation de la matrice avec val
    for i in range(nb_ligne):
        for j in range (nb_colonne):
            matrice[i][j]=valeur
    return matrice
```

Exemple :

```
matrice=initMatVal(2,4,'a')
print(matrice)
matrice2=initMatVal(2,4,8)
print(matrice2)
```

```
[['a', 'a', 'a', 'a'], ['a', 'a', 'a', 'a']]
[[8, 8, 8, 8], [8, 8, 8, 8]]
```