

Série d'exercices N° 03

Exemple d'application pour les tableaux 2D (Les matrices):

Soit M un tableau à deux dimensions de 5 par 4 entiers complètement initialisé :

$$M = \begin{pmatrix} 6 & 2 & 3 & 5 \\ 4 & 6 & 2 & 6 \\ 1 & 3 & 6 & 7 \\ 1 & 6 & 3 & 6 \\ 6 & 0 & 1 & 4 \end{pmatrix}$$

1. Effectuer les actions suivantes :

- Afficher le tableau.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
print(M)
```

- Afficher le sous-tableau de la 2^{ème} ligne à la 3^{ème} ligne comprises.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
print(M[1:3:])
```

- Afficher le sous-tableau du début à la 2^{ème} ligne comprise.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
print(M[0:2:])
```

- Afficher le sous-tableau de la 3^{ème} ligne comprise à la fin du tableau.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
print(M[2::])
```

- Afficher le tableau constitué uniquement des lignes d'indices paires.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
print(M[1::2])
```

- Afficher le tableau constitué uniquement des éléments d'indices impaires (ceci correspond à afficher uniquement les colonnes d'indice impaire de la matrice)

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
print(M[:,0::2])
```

2. Écrire un programme qui permet premièrement d'afficher la diagonale avec la somme des ces coefficients, puis afficher la deuxième diagonale avec la somme des ces coefficients.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
diag1=np.zeros(4)
som1=0
diag2=np.zeros(3)
som2=0
for i in range(0,4):
    for j in range(0,4):
        if i==j:
            diag1[i]=M[i,j]
            som1+=diag1[i]
print('le diagonale =',diag1 )
print('la somme des elements de diagonale',som1 )
for i in range(1,4):
    diag2[i]=M[i,i+1]
    som2+=diag2[i]
print('le 2 diagonale =',diag2 )
print('la somme des elements de 2 diagonale',som2 )
```

3. Tester si les nombres qui apparaissent dans les deux diagonales sont tous égaux à une seule et même valeur. Le programme affiche “Oui” ou “Non” en fonction des résultats.

```
val1=diag1[0]
k=0
for i in range(1,4):
    if diag1[i]==val1:
        k=k+1
if k==3:
    print('Oui')
else:
    print('Non')
val2=diag2[0]
h=0
for i in range(1,3):
    if diag2[i]==val2:
        h=h+1
if h==2:
    print('Oui')
else:
    print('Non')
```

4. Écrire un programme qui permet de remplacer les valeurs du triangle supérieure par des zéros et maintenir les valeurs du triangle inférieure.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
for i in range(0,5):
    for j in range(0,4):
        if i<j:
            M[i,j]=0
print(M)
```

5. Écrire un programme qui permet de calculer la moyenne par colonne. Les valeurs des moyennes seront sauvegardées dans une liste. Ensuite, indiquer l'indice de la colonne correspondant à la plus grande moyenne.

```
import numpy as np
M=np.array([[6,2,3,5],[4,6,2,6],[1,3,6,7],[1,6,3,6],[6,0,1,4]])
moy=np.zeros(4)
for i in range(0,4):
    som=0
    for j in range(0,5):
        som=som+M[j,i]
    moy[i]=som/5
print('la moyenne par colonne = ',moy)
maxm=moy[0]
for i in range(1,4):
    if moy[i]>maxm:
        maxm=moy[i]
        k=i
print('l indice de la colonne correspondant à la plus grande moyenne = ',k)
```