

On suppose déjà importé le module `numpy` d'alias `np`.

**Exercice 1 (3 points)**

Pour chacun des programmes suivants, que va afficher Python à l'écran ?

Programme 1

```
A=np.linspace(10,20,5)
print(A)
```

Programme 2

```
B=np.arange(40,50,2)
print(B)
```

Programme 3

```
C=np.array(([1,3],[4,7]))
for i in range(2):
    C[i,0]=2*C[i,0]
print(C)
```

**Exercice 2 (3 points)**

Ecrire une fonction dont l'en-tête est `def trace(A)`: prenant comme paramètre d'entrée une matrice carrée A et renvoyant la somme des coefficients diagonaux de A.

**Exercice 3 (4 points)**

On dit qu'une matrice  $A \in \mathcal{M}_n(\mathbf{R})$  est *magique* si sur chacune de ses lignes, la somme de ses coefficients est toujours la même.

Par exemple,  $A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 2 & 4 \\ 6 & 2 & 1 \end{pmatrix}$  est *magique* car  $1+3+5=3+2+4=6+2+1$ .

Rappel : en Python, on numérote les lignes de A à partir de l'indice 0.

1)Ecrire une fonction dont l'en-tête est `def sommeliigne(A,i)`: prenant comme paramètre d'entrée une matrice carrée A et un entier  $i \in \llbracket 0, n-1 \rrbracket$ , et renvoyant la somme des coefficients de la ligne d'indice i de A.

2)Compléter la fonction suivante pour qu'elle retourne "magique" si A est magique et "non magique" sinon.

```
def magique(A):
    n=len(A)
    L=[sommeliigne(A,i) for i in range(n)]
    a=L.count(.....)
    if a== ..... :
        print("magique")
    else:
        print("non magique")
```