TP4 Python (matrices, réduction)

Exercice 1

On considère le programme suivant :

```
a=1
b=1
k=0
while k<11:
    print(a)
#affectation parallèle
    a,b,k=b,a+b,k+1
```

1)Par quelle instruction pouvait-on regrouper les trois premières lignes en une seule ?

2)A un moment de la boucle, on a : k=6, a=13 et b=21.
Que deviennent ces valeurs après passage dans l'affectation parallèle ?

3)Exécutez ce programme. Il affiche les termes consécutifs d'une suite. Laquelle ?

Exercice 2 (inspiré d'ecricome 2018)

Soient $A = \begin{pmatrix} 2 & 1 & -2 \\ 0 & 3 & 0 \\ 1 & -1 & 5 \end{pmatrix}$ et $B = \begin{pmatrix} 1 & -1 & -1 \\ -3 & 3 & -3 \\ -1 & 1 & -1 \end{pmatrix}$.

Soit $(X_n)_{n\in\mathbf{N}}$ la suite de $\mathscr{M}_{3,1}(\mathbf{R})$ définie par :

$$\forall n \in \mathbf{N}, X_{n+2} = AX_{n+1} + BX_n,\ X_0 = \begin{pmatrix} 3 \\ 0 \\ -1 \end{pmatrix} \text{ et } X_1 = \begin{pmatrix} 3 \\ 0 \\ -2 \end{pmatrix}.$$

Ecrivez un programme accomplissant les tâches suivantes :

− création des matrices A, B, X_0 et X_1,

− affichage des matrices X_0, X_1, ..., X_{10} (utiliser une affectation parallèle comme pour l'exercice 1).

Exercice 3

Soient $A = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ -2 & 1 & 3 \end{pmatrix}$, $U = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$, $V = \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix}$ et $W = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$.

Vérifiez sur la console que U, V et W sont des vecteurs propres de A et précisez leur valeur propre associée.

Exercice 4

1)Soit $A \in \mathscr{M}_n(\mathbf{R})$. Justifiez que λ est valeur propre de A si et seulement si $rg(A-\lambda I) < n$.

2)Soit $A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$.

a)A l'aide de la question 1) et de la fonction matrix_rank, vérifiez sur la console que -1, 0 et 2 sont des valeurs propres de A.

b)Confirmez ce résultat par la commande eig.

c)A est-elle diagonalisable ?

Exercice 5 (extrait HEC 2017)

```
import numpy as np
import numpy.linalg as al
A=np.array(([0,1,0],[1,0,0],[0,0,1]))
P=np.array(([1,1,0],[1,-1,0],[0,0,1]))
B=np.dot(al.inv(P),A)
C=np.dot(B,P)
print(C)
```

Le programme ci-dessus affiche la matrice $\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$.

1)Déterminez sans calculs les valeurs propres de A.

2)Donnez une base de chacun des sous-espaces propres de A.

Exercice 6

On considère le programme suivant :

```
import numpy as np
A=np.zeros(shape=(5,5))
for i in range(5):
    for j in range(5):
        if j==i+1:
            A[i,j]=1
B=A+np.transpose(A)
print(B)
```

1)Ecrivez la matrice B affichée par le programme.

2)B est-elle diagonalisable ?

3)Déterminez une valeur propre « évidente »de B.

4)Déterminez toutes les valeurs propres de B à l'aide de la commande eig.

Exercice 7

1)Ecrivez une fonction f d'argument A matrice carrée et qui renvoie le nombre d'éléments nuls de A.

2)Ecrivez une fonction g d'argument A matrice carrée et qui renvoie $True$ si A est une matrice diagonale et $False$ sinon.

Exercice 8

1)Soit la fonction mystere d'arguments $p \in \mathbf{N}$, $T \in \mathscr{M}_n(\mathbf{R})$.

```
import numpy as np
def mystere(T,p):
    temp=np.eye(len(T))
    for k in range(p):
        temp=np.dot(temp,T)
    return temp
```

Que renvoie t-elle ? Par quelle commande simple pouvait-on obtenir le même résultat ?

2)A l'aide de la fonction mystere, vérifiez sur un exemple de matrice le théorème suivant :

« si tous les élements diagonaux d'une matrice triangulaire T sont dans l'intervalle $]-1,1[$, alors T^p tend vers 0 quand $p \to +\infty$ ».

3)On souhaite démontrer ce théorème dans le cas particulier où $T = \begin{pmatrix} a & b \\ 0 & c \end{pmatrix}$ avec a,b,c réels tels que 0<c≤a< 1 et b≥ 0.

a)Montrez par récurrence que pour tout entier p≥0, il existe un réel b_p tel que $T^p = \begin{pmatrix} a^p & b_p \\ 0 & c^p \end{pmatrix}$ où $0 \leq b_p \leq pa^{p-1}b$.

b)Conclure.

4)Démontrez le théorème dans le cas où T est une matrice triangulaire diagonalisable de $\mathscr{M}_2(\mathbf{R})$.