# Correction TP4 python

Exercice 1

1)On peut utiliser l'affectation parallèle : a,b,k=1,1,0

2)L'affectation parallèle signifie :
new_a=old_b, new_b=old_a+old_b et new_k=old_k+1.

Comme old_a=13, old_b=21 et old_k=6 , on déduit :
new_a=21, new_b=13+21=34 et new_k=6+1=7.

Après passage dans l'affectation parallèle, on a donc : a=21, b=34 et k=7.

3)Le programme affiche les termes $u_0$, $u_1$, ..., $u_10$ de la *suite de Fibonacci* $(u_n)_{n\in\mathbf{N}}$ définie par $\forall n \in \mathbf{N}, u_{n+2} = u_{n+1} + u_n$ et $u_0 = u_1 = 1$.
On obtient : 1,1,2,3,5,8,13,21,34,55,89.

Exercice 2

```
import numpy as np
A=np.array(([2,1,-2],[0,3,0],[1,-1,5]))
B=np.array(([1,-1,-1],[-3,3,-3],[-1,1,1]))
X=np.array(([3],[0],[-1]))
Y=np.array(([3],[0],[-2]))
k=0
while k<11:
    print('X_',k,'=',X)
    X,Y,k=Y,np.dot(A,Y)+np.dot(B,X),k+1
```

Exercice 3

Sur la console :

```
>>>import numpy as np
>>>A=np.array(([0,1,1],[0,2,0],[-2,1,3]))
>>>U=np.array(([1],[0],[1]))
>>>V=np.array(([0],[1],[-1]))
>>>W=np.array(([1],[1],[1]))
>>>np.dot(A,U)
array([[1],[0],[1]])
>>>np.dot(A,V)
array([[ 0],[ 2],[-2]])
>>>np.dot(A,W)
array([[2],[2],[2]])
```

La console donne $AU = U$, $AV = 2V$ et $AW = 2W$.
D'autre part, $U \neq 0$, $V \neq 0$ et $W \neq 0$.

Ainsi, $U$, $V$ et $W$ sont des vecteurs propres de $A$ associés respectivement aux valeurs propres 1,2 et 2.

Exercice 4

1)D'après le THM1, $\lambda$ est valeur propre de $A$ si et seulement si $A-\lambda I$ n'est pas inversible.

Or, les matrice inversibles de $\mathscr{M}_n(\mathbf{R})$ sont celles de rang n.

2)a)Sur la console :

```
>>>import numpy as np
>>>import numpy.linalg as al
>>>A=np.array(([0,1,0],[1,0,1],[1,1,1]))
>>>I=np.eye(3)
>>>al.matrix_rank(A+I)
2
>>>al.matrix_rank(A)
2
>>>al.matrix_rank(A-2*I)
2
```

On voit que les matrices $A+I$, $A$ et $A-2I$ ont un rang inférieur à 3, ce qui prouve que -1, 0 et 2 sont des valeurs propres de $A$.

2)b)La commande al.eig(A) renvoie les valeurs propres et une base des sous-espaces propres de A.

c)$A \in \mathscr{M}_3(\mathbf{R})$ et admet 3 valeurs propres. Donc $A$ est diagonalisable d'après le théorème de réduction.

Exercice 5

1)On a : $B = P^{-1}A$ et $C = BP$. Donc $C = P^{-1}AP$.

Comme $A$ est semblable à $C$, elle a les mêmes valeurs propres que $C$.

Or, $C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ est diagonale. Ses valeurs propres sont $-1$ et 1.

Donc les valeurs propres de $A$ sont $-1$ et 1.

2)$A$ est diagonalisable car semblable à une matrice diagonale. Une base des sous-espaces propres de $A$ s'obtient par les colonnes de $P$.

Ainsi, $E_1(A) = \text{Vect}\left(\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}\right)$ et $E_{-1}(A) = \text{Vect}\left(\begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}\right)$.

Exercice 6

1)$B = A + {}^tA$ avec $A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$. Donc $B = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$.

2)${}^tB = {}^t\left(A + {}^tA\right) = {}^tA + {}^t\left({}^tA\right) = {}^tA + A = B$. Donc $B$ est symétrique.

$B$ est donc diagonalisable.

3)En notant $C_i$ la $i$-ème colonne de $B$, on voit que $C_3 = C_1 + C_5$.
Les vecteurs colonnes de $B$ sont donc liés, ce qui prouve que $B$ n'est pas inversible. Donc 0 est valeur propre de $B$.

4)Pour trouver toutes les valeurs propres de $B$, on importe le module numpy.linalg en prenant l'alias al.
La commande al.eig(B) renvoie :
(array([ 1.73205081, -1.73205081, -1, -6.61284627e-17,1])
Les valeurs propres de $B$ sont donc $\sqrt{3}$, $-\sqrt{3}$, $-1$, 0 et 1.

Exercice 7

1)programme :

```
import numpy as np
def f(A):
    c=0
    for i in range(len(A)):
        for j in range(len(A)):
            if A[i,j]==0:
                c+=1
    return c
```

2)programme :

```
import numpy as np
def g(A):
    for i in range(len(A)):
        for j in range(len(A)):
            if i!=j and A[i,j]!=0:
                return False
    return True
```

Exercice 8

1)La fonction renvoie $T^p$.
On obtenait directement ce résultat par la commande al.matrix_power(T,p) après avoir importé le module numpy.linalg d'alias al.

2)Prenons par exemple $T = \begin{pmatrix} 1/2 & 1 & 3 \\ 0 & -1/2 & 5 \\ 0 & 0 & 1/3 \end{pmatrix}$.

On obtient $T^{10} = \begin{pmatrix} 0,000976 & 0 & 0,0518 \\ 0 & 0,000976 & -0,00575 \\ 0 & 0 & 0,000017 \end{pmatrix}$.

Les coefficients de $T^{10}$ sont proches de zéro.

3)a)Soit $\mathscr{P}(p)$ la proposition : « il existe un réel $b_p$ tel que $T^p = \begin{pmatrix} a^p & b_p \\ 0 & c^p \end{pmatrix}$ où $0 \leq b_p \leq pa^{p-1}b$ ».

$\mathscr{P}(0)$ est vraie car $T^0 = I$. On prend alors $b_0 = 0$. L'égalité et la double inégalité sont vérifiées.

Soit $p \in \mathbf{N}$. Supposons $\mathscr{P}(p)$ vraie, montrons que $\mathscr{P}(p+1)$ est vraie.

$$T^{p+1} = T^p T = \begin{pmatrix} a^p & b_p \\ 0 & c^p \end{pmatrix}\begin{pmatrix} a & b \\ 0 & c \end{pmatrix} = \begin{pmatrix} a^{p+1} & ba^p + cb_p \\ 0 & c^{p+1} \end{pmatrix}.$$

Posons $b_{p+1} = ba^p + cb_p$.

On a $b_p \geq 0$ par hypothèse de récurrence. De plus, $a$, $b$ et $c$ sont positifs ou nuls donc $b_{p+1} \geq 0$.

De plus, $b_p \leq pa^{p-1}b$ par hypothèse de récurrence donc $cb_p \leq cpa^{p-1}b$.

De plus, $cpa^{p-1}b \leq pa^p b$ car $c \leq a$.

Par recollement d'inégalités : $cb_p \leq pa^p b$.

On déduit que $a^p b + cb_p \leq a^p b + pa^p b$, soit $b_{p+1} \leq (p+1)a^p b$.

Donc $\mathscr{P}(p+1)$ est vraie.

4)Soit $T$ une matrice triangulaire diagonalisable de $\mathscr{M}_2(\mathbf{R})$.

Posons $T = \begin{pmatrix} a & b \\ 0 & c \end{pmatrix}$.

$T$ est triangulaire donc ses valeurs propres sont $a$ et $c$.

$T$ est diagonalisable donc il existe une matrice $D$ diagonale portant sur sa diagonale les valeurs propres de $T$ et une matrice $P$ inversible telles que $T = PDP^{-1}$.

Par récurrence, on obtient $\forall p \in \mathbf{N}, T^p = PD^pP^{-1}$.

Posons $D = \begin{pmatrix} a & 0 \\ 0 & c \end{pmatrix}$, $P = \begin{pmatrix} d & e \\ f & g \end{pmatrix}$ et $P^{-1} = \begin{pmatrix} i & j \\ k & l \end{pmatrix}$.

On déduit pour tout $p \in \mathbf{N}$ :

$$\begin{aligned} T^p &= \begin{pmatrix} d & e \\ f & g \end{pmatrix}\begin{pmatrix} a^p & 0 \\ 0 & c^p \end{pmatrix}\begin{pmatrix} i & j \\ k & l \end{pmatrix} \\ &= \begin{pmatrix} da^p & ec^p \\ fa^p & gc^p \end{pmatrix}\begin{pmatrix} i & j \\ k & l \end{pmatrix} \\ &= \begin{pmatrix} ida^p + kec^p & jda^p + elc^p \\ ifa^p + kgc^p & jfa^p + lgc^p \end{pmatrix}. \end{aligned}$$

Par hypothèse, $-1 < a < 1$ et $-1 < c < 1$.

Donc $\lim\limits_{p\to+\infty} a^p = 0$ et $\lim\limits_{p\to+\infty} c^p = 0$.

On déduit que $\lim\limits_{p\to+\infty} T^p = 0$.