TD 20 -
Résolution de systèmes
d'équations différentielles du 1er Ordre

Informatique
MPSI-PCSI - Lycée Thiers

Exercice 1

Une solution composée de réactifs A, B, C, où deux réactions chimiques $A \longrightarrow B$ et $B \longrightarrow C$ d'ordre 1 rentrent en jeu selon les lois.

$$\begin{cases} \dfrac{d[A]}{dt} & = -[A] \\ \dfrac{d[B]}{dt} & = 2[A]-[B] \\ \dfrac{d[C]}{dt} & = [B]/2 \end{cases}$$

1. Tracer l'évolution des concentrations des 3 réactifs A , B, C entre 0 et 8s par pas de 0.01s, en partant des concentrations initiales $[A]=1$, $[B]=[C]=0$.
2. Au bout de 8s quelle est la concentration de chacun des réactifs ?
3. A quel moment, à 0.01s près, les concentrations en produits A et C sont-elles égales ?

Exercice 1 - Corrigé

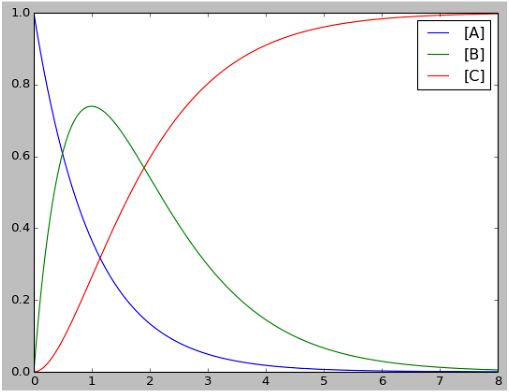
1).

```
import numpy as np
dt = 0.01
T=np.arange(0,8+dt,dt)
n = len(T)
A, B, C = np.empty(n), np.empty(n), np.empty(n)
A[0], B[0], C[0] = 1, 0, 0
for k in range(n-1):
A[k+1] = A[k] + dt * (-A[k])
B[k+1] = B[k] + dt * (2*A[k]-B[k])
C[k+1] = C[k] + dt * B[k]/2

import matplotlib.pyplot as plt
plt.figure(1)
plt.plot(T,A,label='[A]')
plt.plot(T,B,label='[B]')
plt.plot(T,C,label='[C]')
plt.legend()
```

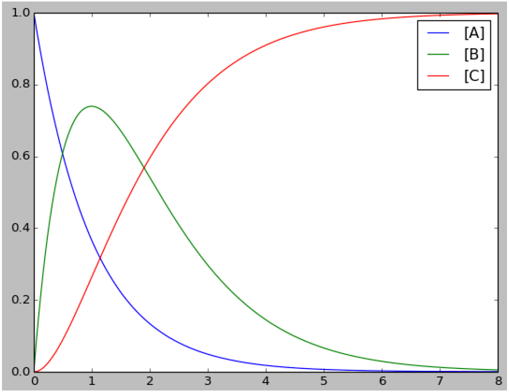
Exercice 1 - Corrigé

1)



Exercice 1 - Corrigé

1)



2)

```
In [2]: A[-1], B[-1], C[-1]
Out[2]: (0.00032222236288023605, 0.0052076341475593693, 0.99707396056334063)
```

Exercice 1 - Corrigé

3) Il faut pratiquer une recherche de racine par dichotomie sur le le tableau `F = A-C` :

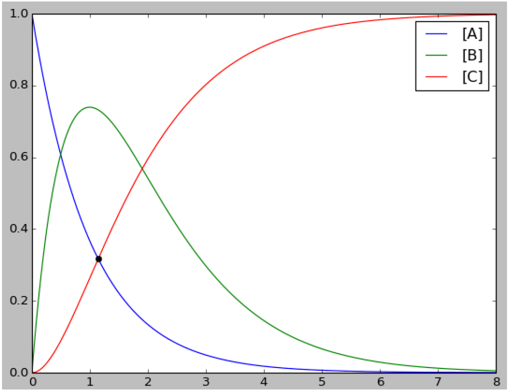
```
F = A-C
debut = 0
fin = n-1
while fin-debut > 1:
    milieu = (debut + fin)//2
    if F[debut] * F[milieu] <= 0:
        fin = milieu
    else:
        debut = milieu

print("Croisement à ",T[milieu],"secondes")
plt.plot(T[milieu],A[milieu],'o',color='black')
```

Exercice 1 - Corrigé

3) Résultats :

```
Croisement à 1.14 secondes
```



Exercice 2

Les équations différentielles de Lotka-Volterra dites "modèle proie-prédateur" permettent de modéliser l'évolution d'une population constituée de proies et de prédateurs.

– $x(t)$, $y(t)$ représentent respectivement l'effectif des populations de proies et de prédateurs en fonction du temps t.

– La constante a est le taux de reproduction des proies.

– La constante b est le taux de mortalité des proies dû aux prédateurs.

– La constante c est le taux de mortalité des prédateurs.

– La constante d est le taux de reproduction des prédateurs en fonction du nombre de proie.

$$\begin{cases} \dfrac{dx}{dt}(t) = x(t)(a - b \cdot y(t)) \\ \dfrac{dy}{dt}(t) = -y(t)(c - d \cdot x(t)) \end{cases}$$

On prendra pour valeurs numériques :
$a = 1,15$, $b = 0,015$, $c = 1,25$ et $d = 0,005$

Exercice 2

1. Tracer sur un même graphique l'évolution des effectifs des proies et des prédateurs sur 30 unités de temps, pour les conditions initiales $x(0) = 300$, $y(0) = 150$.
2. Tracer sur un même graphe les courbes paramétriques $(x(t), y(t))$ des effectifs des populations à l'instant t avec comme condition initiales : $(x(0), y(0)) \in \{(300, 150), (150, 150), (150, 300), (300, 300)\}$. Tracer également la droite d'équation $x + y = 150$.
3. On constitue un bassin de pêche avec deux espèces de poissons, l'une proie, l'autre prédateur, dont l'évolution suit un modèle de Lotka-Volterra avec pour valeurs numériques a, b, c, d celles figurant ci-dessus. Parmi les 4 possibilités $(300, 150)$, $(150, 150)$, $(150, 300)$ et $(300, 300)$ de populations initiales, laquelle conseilleriez-vous, sachant que :
   - La population totale du bassin doit rester relativement stable, et ne pas descendre trop sous la barre de 150 individus.
   - Le remplissage initial du bassin a un coût financier important qu'il s'agit de modérer.

Exercice 2 - Corrigé

1) On résout avec Odeint. Cela nécessite d'écrire le système d'équations vectoriellement. En posant :

$$Y(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \qquad Y(0) = \begin{pmatrix} 300 \\ 150 \end{pmatrix} \qquad \Phi : \begin{pmatrix} x \\ y \end{pmatrix}, t \longmapsto \begin{pmatrix} x \times (a - b \times y) \\ y \times (c - d \times x) \end{pmatrix}$$

le système s'écrit sous forme vectorielle :

$$\frac{d}{dt} Y(t) = \Phi(Y(t), t) \quad \text{avec condition initiale : } Y(0) = \begin{pmatrix} 300 \\ 150 \end{pmatrix}$$

Exercice 2 - Corrigé

On le résout alors, par exemple à l'aide de la fonction `odeint` :

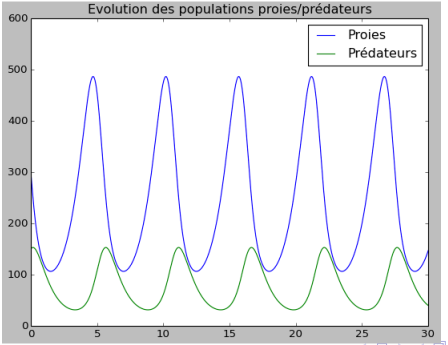
```
from numpy import array, linspace
from scipy import integrate

# Constantes du probleme
T = 30
a = 1.15
b = 0.015
c = 1.25
d = 0.005

# Résolution
phi = lambda Y,t: array([Y[0]*(a-b*Y[1]), -Y[1]*(c-d*Y[0])])
t = linspace(0,T,1000)
y0 = array([300,150])
result = integrate.odeint(phi,y0,t)
```

Exercice 2 - Corrigé

```
# Tracé
import matplotlib.pyplot as plt
plt.figure(1)
plt.plot(t,result)
plt.axis([0,30,0,600])
plt.legend(('Proies','Prédateurs')) plt.title("Evolution des
populations proies/predateurs")
```

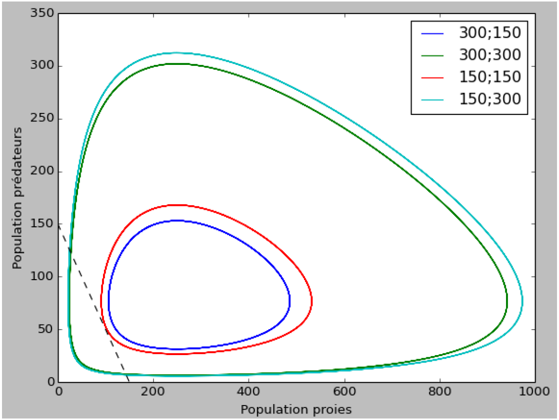


Exercice 2 - Corrigé

2) Il faut tracer le nombre de prédateur en fonction du nombre de proie.

```
plt.figure(2)
A,B = 300,150
L = [(A,B),(A,A),(B,B),(B,A)]
for y0 in L:
    result = integrate.odeint(phi,y0,t)
    x = result[:,0]
    y = result[:,1]
    plt.plot(x,y)
plt.legend((str(A)+';'+str(B),str(A)+';'+str(A),str(B)+';'+str(B)
            ,str(B)+';'+str(A)))
plt.xlabel('Population proies')
plt.ylabel('Population prédateurs')
pMin = 150
plt.plot([0,pMin],[pMin,0],'--k')
plt.show()
```

Exercice 2 - Corrigé



3) On le constate, les seules possibilités acceptables, pour lesquelles la population totale ne descendrait pas sous la barre des 150 individus sont $(300, 150)$ et $(150, 150)$. Vu la seconde contrainte on conseillerait une population initiale de 150 individus de chaque espèce.