
Chapitre 4 : Corrigés des exercices

5 Exercices

Exemple 5.1. On applique la formule du cours :

```
def trapeze(f,a,b,n):
    A,x=0,a
    for k in range(n):
        A=A+f(x)+f(x+(b-a)/n)
        x=x+(b-a)/n
    return(((b-a)/(2*n))*A)
```

Exemple 5.2. On souhaite obtenir une valeur approchée des solutions éventuelles du problème de Cauchy, où y_0 est un réel (un flottant en Python) :

$$\begin{cases} y' &= 1 - y^2 \\ y(0) &= y_0 \end{cases}$$

1. On adapte à notre équation le cours :

```
def Euler(y0,T,h):
    t=0
    y=y0
    listey=[y]
    listet=[t]
    while t<=T:
        y=y+h*(1-y**2)
        t=t+h
        listey.append(y)
        listet.append(t)
    return(listet,listey)
```

2. On utilise la syntaxe habituelle :

```
import matplotlib.pyplot as plt
import numpy as np

plt.clf()
for pas in [1.5,1,0.5,0.1]:
    X,Y=Euler(f,5,pas)
    plt.plot(X,Y,label='pas='+str(pas))
plt.show()
```

3. On adapte les questions précédentes en complétant avec des temps décroissants :

```
def Euler(y0,T,h):
    t=0
    y=y0
    listey=[y]
    listet=[t]
    while t<=T:
        y=y+h*(1-y**2)
        t=t+h
        listey.append(y)
        listet.append(t)
    t=0
    y=y0
    while t>=-T:
        y=y-h*(1-y**2)
        t=t-h
        listey=[y]+listey
        listet=[t]+listet
    return(listet,listey)
```

4. On observe de l'instabilité numérique.