Informatique en CPGE (2018-2019)
Corrigé : équations différentielles

Exercice 1 Corrigé

1.

```
def integRec(f,a,b,h):
    x,s=a,0
    while x<b:
        s+=h*f(x)
        x+=h
    return s

def f(x):
    return 1/x

print(integRec(f,1,2,0.01))
```

La valeur obtenue est 0.695653430481824.

2.

```
def euler(f,a,b,y0,h):
    x,y=a,y0
    liste_x=[x]
    liste_y=[y]
    while x<b:
        y+=h*f(x)
        liste_y.append(y)
        x+=h
        liste_x.append(x)
    return liste_x,liste_y

a,b=euler(f,1,2,0,0.01)
print(b[-1])
```

La valeur obtenue est encore 0.695653430481824. C'est normal puisque les deux méthodes sont "identiques".

La valeur exacte est $\ln(2) \simeq 0.6931471805599453$.

3.

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

plt.plot(a,b,"*") # solution approchée

a=np.linspace(1,2,101)
b=np.log(a)
plt.plot(a,b) # solution exacte

plt.show()
```

Exercice 2 Corrigé

1.

```
def integTrap(f,a,b,h):
    s=f(a)+f(b)
    x=a+h
    while x<b:
        s+=2*f(x)
        x+=h
    return s*h/2

from math import exp,pi
def f(x):
    return exp(-x**2)

I=integTrap(f,0,8,0.01)
```

Nous obtenons la valeur approchée $I \simeq 0.8862269254527576$.

2.

```
def euler(f,a,b,y0,h):
    x,y=a,y0
    liste_x=[x]
    liste_y=[y]
    while x<b:
        y+=h*f(x+h/2)
        liste_y.append(y)
        x+=h
        liste_x.append(x)
    return liste_x,liste_y

a,b=euler(f,0,8,0,0.01)
print(b[-1],I,pi**0.5/2)
```

Nous obtenons $y(2) \simeq 0.886226925452757$.

Le nombre $\sqrt{\pi}/2$ est une valeur approchée de $\displaystyle\int_0^8 \mathrm{e}^{-t^2}\mathrm{d}t$, soit environ 0.8862269254527579.

Note : avec la méthode d'Euler classique, (exercice 1), la valeur obtenue est légèrement supérieure ($\simeq 0.891226925452757$).

3.

```
import matplotlib.pyplot as plt
plt.plot(a,b)
plt.show()
```

La courbe se rapproche très vite de son asymptote d'équation $y = 1$.