

**Informatique en CPGE (2018-2019)**  
**Corrigé TP 4 : les fonctions**

**Exercice 1**

$$\frac{f(x)}{g(x)} = \frac{\sqrt{1+x}-1}{x} \times (\sqrt{1+x}+1) = \frac{\sqrt{1+x}^2-1}{x} = \frac{1+x-1}{x} = 1.$$

```
from math import sqrt
def f(x):
    return (sqrt(1+x)-1)/x
def g(x):
    return 1/(sqrt(1+x)+1)
for i in range(1,5):
    print(f(10**(-5*i)),g(10**(-5*i)))
```

La limite est 0,5 donnée par la fonction  $g$ .

**Exercice 2** : recherche de solutions de l'équation  $\ln(x) - 2 + x = 0$  avec  $a = 1$  et  $b = 2$ .

```
from math import log
def dichotomie(f,a,b,eps):
    signe=f(a)
    while b-a>eps:
        m=(a+b)/2
        if signe*f(m)<0:
            b=m
        else:
            a=m
    return (a+b)/2
def f(x):
    return log(x)-2+x
print(dichotomie(f,1,2,0.001))
```

**Exercice 3** : recherche d'extremum de la fonction  $f$  définie par  $f(x) = xe^{-x}$  sur l'intervalle  $[0; 2]$ .

1. Programme 1 (déterministe à pas constant) :

```
from math import exp
def maximum1(f,a,b,n):
    dx=(b-a)/n
    x=a
    maxi=f(x)
    for k in range(0,n):
        x+=dx
        y=f(x)
        if y>maxi:
```

```
        maxi=y
    return maxi

def f(x):
    return x*exp(-x)

a=0
b=2
n=1000
print('Le maximum de f sur [a; b] est:',maximum1(f,a,b,n))
```

2. Programme 2 du cours (tabulation « aléatoire » d'une fonction) :

```
from math import exp
from random import random

def maximum2(f,a,b,n):
    maxi=f(a)
    for k in range(0,n):
        x=a+(b-a)*random()
        y=f(x)
        if y>maxi:
            maxi=y
    return maxi

def f(x):
    return x*exp(-x)

a=0
b=2
n=1000

print('Le maximum de f sur [a; b] est :',maximum2(f,a,b,n))
```

**Exercice 4** : test de la monotonie sur l'intervalle  $[0; \pi]$  avec  $f(x) = x + 1,0001 \cos x$ .

```
from math import cos,pi

def monotonie(f,a,b,n):
    sens=f(b)-f(a)
    x=a
    dx=(b-a)/n
    monotone=True
    for k in range(n):
        if (f(x+dx)-f(x))*sens<0:
            print("la fonction n'est pas monotone")
            print(x,x+dx)
            monotone=False
            break
        x+=dx
    if monotone==True:
        print("la fonction semble monotone")
```

```
def f(x):
    return x+1.0001*cos(x)

a=0
b=pi
n=100
monotonie(f,a,b,n) # affiche "la fonction semble monotone"
n=1000
monotonie(f,a,b,n) # affiche "la fonction n'est pas monotone"
```

### Exercice 5 : calcul d'intégrales

Approximation de l'intégrale  $\int_0^1 x e^{-x} dx$  en utilisant la méthode des trapèzes.

```
from math import exp

def trapezes(f,a,b,n):
    s=0
    h=(b-a)/n
    x=a
    for i in range(n):
        x+=h
        s+=(f(x)+f(x+h))/2
    s*=h
    return s

def f(x):
    return x*exp(-x)

a=0
b=1
n=100
print(trapezes(f,a,b,n))
```

### Exercice 6 : méthode de Monte Carlo

```
from math import exp
from random import random

def monte_carlo(f,a,b,n):
    s=0
    h=(b-a)/n
    for i in range(n):
        x=a+(b-a)*random()
        s+=f(x)
    s*=h
    return s

def f(x):
    return x*exp(-x)

a=0
b=1
n=100
print(monte_carlo(f,a,b,n))
```