

Correction TP 7 :Fonction

EXERCICE 1.

1. Définir avec Scilab une fonction f qui à x associe $(1 + 3x + 2x^2 - x^4)e^{-x}$.

```
function y=f(x)
    y=(1+3*x+2*x^2-x^4)*exp(-x)
endfunction
```

2. Donner la valeur de $f(1)$, $f(-1)$ et $f(\sqrt{3})$.

`f(1)` : 1.8393972

`f(-1)` : -2.7182818

`f(sqrt(3))` : 0.5654671

3. Écrire un programme permettant de tracer la courbe de la fonction f entre 0 et 3 à partir de la fonction Scilab définie précédemment. Noter le message d'erreur que vous obtenez.

```
clf
x=[0 :0.01 :3]
y=f(x)
plot2d(x,y)
```

Dimensions ligne/colonne incohérentes.

4. Comment modifier le programme définissant la fonction f pour pouvoir la tracer ?

```
function y=f(x)
    y=(ones(x)+3.*x+2.*x.^2-x.^4).*exp(-x)
endfunction
```

EXERCICE 2.

1. Écrire un programme qui permet de tracer entre -1 et 1 et sur le même graphe les fonctions suivantes :

$$x \mapsto e^x ; x \mapsto 1 + x ; x \mapsto 1 + x + \frac{x^2}{2} ; x \mapsto 1 + x + \frac{x^2}{2} + \frac{x^3}{6}$$

```
clf
x=[-1 :0.01 :1]
y=exp(x)
a=ones(x)+x
b=ones(x)+x+x.^2./2
c=ones(x)+x+x.^2./2+x.^3./6
plot2d(x,y)
plot2d(x,a)
plot2d(x,b)
plot2d(x,c)
```

2. Que remarquez vous ?

On se rapproche de plus en plus de la courbe de la fonction exponentielle. Essayez d'ajouter un terme $\frac{x^4}{24}$ puis $\frac{x^5}{120}$ vous verrez que l'approximation devient de plus en plus précise.

EXERCICE 3.

1. Rappeler la formule permettant de calculer x^α pour $\alpha \in \mathbb{R}$ et $x > 0$.

$$x^\alpha = e^{\alpha \ln(x)}$$

2. Écrire une fonction `puissance(x,alp)` qui calcule la valeur de x^{alp} .

```
function y=puissance(x,alp)
    y=exp(alp*log(x))
endfunction
```

EXERCICE 4. On considère la suite $(u_n)_{n \geq 0}$ définie par $u_0 = 1$ et pour tout $n \in \mathbb{N}$, $u_{n+1} = \sqrt{n + u_n}$.

Écrire une fonction `suite` qui calcule le terme d'indice n de la suite $(u_n)_{n \geq 0}$.

```
function y=suite(n)
    y=1
    for k=1 :n
        y=sqrt(k+y)
    end
endfunction
```

EXERCICE 5. On définit la série de terme général $u_n = \sqrt{n}$.

1. Écrire un programme Scilab demandant à l'utilisateur d'entrer un entier n et renvoyant la n -ième somme partielle de la série de terme général u_n .

```
n=input("Entrer un entier n")
S=0
for k=1 :n
    S=S+sqrt(k)
end
disp(S)
```

ou

```
n=input("Entrer un entier n")
K=0 :n
S=sum(sqrt(K))
disp(S)
```

2. Écrire une fonction Scilab prenant en argument un entier n et renvoyant en sortie la valeur de la n -ième somme partielle de la série de terme général u_n .

```
function y=sommesqrt(n)
    y=0
    for k=1 :n
        y=y+sqrt(k)
    end
endfunction
```

ou

```
function y=sommesqrt(n)
    K=0 :n
    y=sum(sqrt(K))
endfunction
```

3. Écrire un programme permettant de tracer les 100 premières itérations de la série de terme général u_n .

```
clf
n=100
K=0 :n
S=cumsum(sqrt(K))
plot(K,S)
```