

**Examen final d'Outils de Programmation 2**

Durée : 1h30

**Exercice 1. (6 pts)**

Donner le résultat Scilab pour chacune des commandes suivantes :

```
--> x=8, y=10
--> z = (x>y) | ~(x==y), t = x == y, ~((x-y > x) & (x+y == 10)), ans & y, ans| 0; ans| 0
--> [4,2,-2:2:3]~2*[2,1,-1,0,1]
--> A=[1 2 -1 3;7 2 0 1;3 -2 -1 -1;0 1 4 8]
--> A(1:3, [2 4])'
--> A(2,:) = A(2, :)-7*A(1, :)
--> A([1 3], [1 3]) = 10*ones(2,2)
--> A($:-1:1, $:-2:1)
--> A = [[A(2:3, [1,3]);A(2:3, [2,4])] ones(4,2)]
```

**Exercice 2. (4 pts)**

Le programme suivant calcule la racine carrée d'un nombre par la méthode de Newton.

```
a=input('donner un nombre positif:');
x=a/2;
precision = 6;
for i = 1:precision
    x = (x + a / x) / 2;
end
disp(x)
```

1. Remplacer l'instruction `for` par l'instruction `while` en préservant la fonctionnalité du programme.
2. Modifier le programme pour qu'il soit applicable sur un vecteur et pas uniquement sur un nombre.
3. Transformez le programme en une fonction.

**Exercice 3. (6 pts)**

Écrire une fonction `antidiag(x,y,z)` qui a pour arguments les trois vecteurs  $x = (x_1, x_2, \dots, x_n)$ ,  $y = (y_1, y_2, \dots, y_{n-1})$ ,  $z = (z_1, z_2, \dots, z_{n-1})$  et qui retourne la matrice :

$$A = \begin{pmatrix} 0 & 0 & \dots & 0 & z_1 & x_1 \\ 0 & 0 & \dots & z_2 & x_2 & y_1 \\ 0 & 0 & \dots & x_3 & y_2 & 0 \\ \vdots & \dots & \dots & \dots & \dots & \vdots \\ z_{n-1} & x_{n-1} & \dots & 0 & 0 & 0 \\ x_n & y_{n-1} & \dots & 0 & 0 & 0 \end{pmatrix}.$$

**Exercice 4. (4 pts)**

On veut représenter la fonction  $f : t \rightarrow f(t) = (\exp(-t/10) \sin(t), \exp(-t/10) \cos(t), \exp(-t))$  sur  $[-2\pi, 2\pi]$ .

1. Créez la fonction  $f$ .
2. Donner la commande **Scilab** permettant de représenter la fonction  $f$  avec une variation de  $t$  de  $-2\pi$  jusqu'à  $2\pi$  et un pas  $2\pi/100$ .

بِالتَّوْفِيقِ

**OPM2: Corrigé d'examen**

**Exercice 1. (6 pts)**

```
--> x=8, y=10
x =
8. (0.25 pt)
y =
10.
--> z = (x>y) | ~(x==y), t = x == y, ~((x-y > x) & (x+y == 10)), ans & y, ans | 0; ans | 0
z =
T (0.5 pt)
t =
F (0.25 pt)
ans =
T (0.5 pt)
ans =
T (0.25 pt)
ans =
T (0.25 pt)
--> [4,2,-2:2:3]~=2*[2,1,-1,0,1]
ans =
F F F F F (0.5 pt)
--> A=[1 2 -1 3;7 2 0 1;3 -2 -1 -1;0 1 4 8]
A =
    1.    2.   -1.    3.
    7.    2.    0.    1.
    3.   -2.   -1.   -1.
    0.    1.    4.    8.
--> A(1:3,[2 4])'
ans =
    2.    2.   -2.
    3.    1.   -1.
--> A(2,:)= A(2,:)-7*A(1,:)
A =
    1.    2.   -1.    3.
    0.  -12.    7.  -20.
    3.   -2.   -1.   -1.
    0.    1.    4.    8.
--> A([1 3],[1 3])= 10*ones(2,2)
A =
   10.    2.   10.    3.
    0.  -12.    7.  -20.
   10.   -2.   10.   -1.
    0.    1.    4.    8.
--> A($:-1:1,$:-2:1)
ans =
    8.    1.
   -1.   -2.
  -20. -12.
    3.    2.
```

```
--> A = [[A(2:3,[1,3]);A(2:3,[2,4])] ones(4,2)]
```

```
A =  
 0.    7.    1.    1.  
 10.   10.    1.    1.  
-12.  -20.    1.    1.  
 -2.   -1.    1.    1.  
      (1 pt)
```

### Exercice 2. (4 pts)

1. 

```
a=input('donner un nombre positif:');  
x=a/2;  
precision = 6;  
i = 1;  
while i <= precision    (1 pt)  
x = (x + a / x) / 2;  
i = i+1;  
end  
disp(x)
```
2. 

```
a=input('donner un vecteur nombres positifs:');  
x=a/2;  
precision = 6;  
for i = 1:precision  
x = (x + a ./ x) / 2; //opérations élément par élément    (1 pt)  
end  
disp(x)
```
3. 

```
function x=racine(a)  
x=a/2;  
precision = 6;  
for i = 1:precision    (2 pts)  
x = (x + a ./ x) / 2;  
endfunction
```

### Exercice 3. (6 pts)

```
function A=antidiag(x,y,z)    (1 pt)  
n=4;  
mat1 = diag(x);    (1 pt)  
mat2 = diag(y,-1);    (1 pt)  
mat3 = diag(z,1);    (1 pt)  
B = mat1 + mat2 + mat3;    (1 pt)  
A = B(:, $:-1:1);    (1 pt)  
endfunction
```

### Exercice 4. (4 pts)

```
function [x,y,z] = f(t)    (0.75 pt)  
x=exp(-t/10)*sin(t);    (0.75 pt)  
y=exp(-t/10)*cos(t);    (0.75 pt)  
z=exp(-t);    (0.75 pt)  
endfunction  
t = -2*%pi:2*%pi/100:2*%pi;    (0.25 pt)  
param3d(x,y,z);    (0.75 pt)
```