

## Corrigé type des Travaux Pratiques N°3

---

### -----Exercice 1-----

```
temperature = input('entrer la temperature du patient : ');  
    if temperature > 37  
        disp('Le patient a la fièvre');  
    else  
        disp('Le patient n"a pas la fièvre');  
    end
```

### -----Exercice 2-----

```
a = input('entrer la 1ere valeur : ');  
b = input('entrer la 2eme valeur : ');  
c = input('entrer la 3eme valeur : ');  
  
    if a > b  
        max = a;  
    else  
        max = b;  
    end  
  
    if max < c  
        max = c;  
    end  
  
    disp([' le max entre les trois nombres : ' num2str(max)])
```

### -----Exercice 3-----

```
mois = input('entrer le numéro du mois : ');  
  
    switch
```

mois

```
case 1
    disp('Janvier')
case 2
    disp('Février')
case 3
    disp('Mars')
case 4
    disp('Avril')
case 5
    disp('Mai')
case 6
    disp('Juin')
case 7
    disp('Juillet')
case 8
    disp('Août')
case 9
    disp('Septembre')
case 10
    disp('Octobre')
case 11
    disp('Novembre')
case 12
    disp('Décembre')
otherwise
    disp('erreur')
end
```

-----Exercice 4-----

```
m = zeros(5,4);
```

```

for l = 1:5
    for c = 1:4
        m(l,c) = l*c;
    end
end
disp(m)

```

-----Exercice 5-----

```

chaine = input(' entrer le mot à vérifier : ', 's');
i = 1;
j = length(chaine);
while i < j & chaine(i) == chaine(j)
    i = i + 1;
    j = j - 1;
end
if i >= j
    disp('le mot entré est palindrome')
else
    disp('le mot entré ce n"est pas un mot est palindrome')
end

```

-----Exercice 6-----

```

*****fonction locale sauvegarder dans le fichier "minTomaj.m" *****
function C = minTomaj(chaine)
a = "";
for k = 1:length(chaine)
    s = char(chaine(k)- 32);
    a = [a s];
end
C = a;

```

-----Exercice 7-----

\*\*\*\*\*fonction inline définie dans l'invite de commande matlab\*\*\*\*\*

```
f = inline('x*cos(x*pi/3)','x')
```

----> utilisation : f(3)

\*\*\*\*\*fonction locale sauvegarder dans le fichier "maFonction.m" \*\*\*\*\*

```
function y = maFonction(x)
```

```
y = x*cos(x*pi/3);
```

```
end
```

-----Exercice 8-----

\*\*\*\*\*fonction locale sauvegarder dans le fichier "cartTopol.m" \*\*\*\*\*

```
function [th, rho] = cartTopol(x,y)
```

```
rho = sqrt(x^2 + y^2);
```

```
th = atan(y/x);
```

```
end
```

## Graphisme

### Exercice 9.

Tracer les graphes des fonctions suivantes sur des figures différentes puis les quatre équations sur la même figure :

**Cas 1** : dans des figures différentes

- ```
>> x = [-5 : 1 : 5]; y = 2*x.^3 + 4*x.^2 - 7;  
>> plot(y, '-');  
>> xlabel('x'), ylabel('y')
```
- ```
>> x = [-4 : 1 : 4]; y1 = 3*exp(-x.^2) + 2;  
>> plot(y1, '-');  
>> xlabel('x'), ylabel('y')
```
- ```
>> x = [-0.5 : 0.1 : 3]; y2 = sin(exp(-x)) - cos(x);  
>> plot(y2, '-');  
>> xlabel('x'), ylabel('y')
```

```
4. >> x= [0 :1 :8] ; y3 = x.*log(x.^3 + 2) ;
    >> plot(y3,'.-')
    >> xlabel('x'), ylabel('y')
```

**Cas 2** : dans la même figure : à la fin de chaque graph on peut ajouter *grid on* pour afficher le quadrillage (après fonction *title*).

```
>> x = [-5:1:5]; y= 2*x.^3+4*x.^2-7 ;
>> subplot(221), plot(x,y)
>> xlabel('x'), ylabel('y'), title('y= 2x^3+4x^2-7')
>> x= [-4 :1 :4] ; y1 = 3*exp(-x.^2) + 2 ;
>> subplot(222), plot(x,y1)
>> xlabel('x'), ylabel('y'), title('y= 3exp(-x^2) + 2')
>> x= [-0.5 :0.1 :3] ; y2 = sin(exp(-x)) - cos(x) ;
>> subplot(223), plot(x,y2)
>> xlabel('x'), ylabel('y'), title('y= sin(exp(-x)) - cos(x) ')
>> x= [0 :1 :8] ; y3 = x.*log(x.^3 + 2) ;
>> subplot(224), plot(x,y3)
>> xlabel('x'), ylabel('y'), title('y= x*log(x^3 + 2) ')
```

### Exercice 10.

Soit  $f$  et  $g$  deux fonctions définies sur l'intervalle  $[0,20]$  par :

$$f(x) = x \cos\left(\frac{x\pi}{3}\right) \text{ et } g(x) = 2e^{-x/10}$$

**Cas 1** : une même fenêtre mais sur des graphes différents.

```
>> subplot(121), fplot('x.*cos(x*pi/3)', [0 20])
>> xlabel('x'), ylabel('f(x)'), title('f(x)= x*cos(x*pi/3) ')
>> grid
>> subplot(122), fplot('2.*exp(-x/10)',[0 20])
>> xlabel('x'), ylabel('g(x)'), title('g(x)= 2*exp(-x/10) ')
>> grid
```

**Cas 2** : dans une même fenêtre mais sur le même graphe, avec la fonction *plot*

```
>>x=[0:20];
>> f = x.*cos(x*pi/3) ; g = 2.*exp(-x/10);
```

```
>>plot(x,f,x,g)
>> xlabel('x'), ylabel('y'), legend ('x*cos(x*pi/3)','2*exp(-x/10)')
```

### Exercice 11.

On considère le tableau suivant défini comme suit :

|   |   |   |   |    |    |    |    |    |    |    |
|---|---|---|---|----|----|----|----|----|----|----|
| x | 1 | 2 | 3 | 4  | 5  | 6  | 7  | 8  | 9  | 10 |
| y | 0 | 4 | 9 | 15 | 26 | 34 | 49 | 64 | 80 | 98 |

- Courbe :

```
>> x=[1 :10] ; y =[0 4 9 15 ... 98]
>> plot(x,y,'g.-') ou plot(x,y,'g.-','LineWidth',2,'MarkerSize',10)
>>
```

- Bâton : bar(x,y)

```
>> bar(x,y)
>> xlabel('x'), ylabel('y')
```

- Tuyau d'orgue : stem(x,y)

```
>> stem (x,y)
>> xlabel('x'), ylabel('y')
```

Puis sauvegarder les figures dans deux fichiers (courbe.png, baton.pdf).

```
- print -dpng courbe
- print -dpdf baton
```

### Exercice 12.

A l'aide de la commande *plot3()* tracer les fonctions suivantes :

-  $z = (3 - x^2)(3 - y^2)$  sur la diagonale du carré  $[-3, 3]$ .

```
>> x = [-3:0.1:3]; y = x;
>> z = (3-x.^2).*(3-y.^2);
>> plot3(x,y, zeros(1,length(x)),x,y,z,'LineWidth',1.5)
>> xlabel('x'),ylabel('y'),zlabel('z')
>> grid on
```

- 
$$\begin{cases} x = \sin(\omega t) \\ y = \cos(\omega t) \\ z = \alpha t \end{cases} \text{ où } t \in [0,10] \text{ variée avec un pas de } 0.02; \omega = 4, \alpha = 0.2$$

```
>> t = 0 : 0.02 : 10 ;
```

```
>> x = sin(4*t) ;
```

```
>> y = cos(4*t) ;
```

```
>> z = 0.2*t ;
```

```
>> plot3(x,y,z)
```

```
>> xlabel('x'),ylabel('y'),zlabel('z')
```