

Séance d'exercices n°1

Exercice 1

Encodez **successivement** les instructions de la liste ci-dessous **dans la fenêtre de commande**. Examinez le résultat de chaque opération et veillez à bien comprendre :

- a) ce que « fait » l'instruction encodée
- b) la signification des messages d'erreur générés par le logiciel Matlab.

Respectez scrupuleusement la suite de caractères à encoder et consignez les messages d'erreur et leur signification dans votre cahier d'exercices.

NB : après l'encodage d'une ligne il faut enfoncer la touche 'return' (ou 'enter' ou 'entrée').

```
5+8
rep=5+8;
pi^2 ; format long; ans
format long e; ans
rep
format short
A=[1 2 3 4 5]; B=[-1 2 -3 4 -5]
A
A*B
A*B'
Mat=A'*B
normA=sqrt(A*A')
A^2
A.^2
vec=A.*A
t=0:pi/5:2*pi
t'
S=sin(t); C=cos(t);
Table=[t' S' C'];
format long e
Table
figure(1); plot(t,S)
figure(2); plot(t,C)
figure(3); plot(C,S,'ro')
figure(3); plot(C,S,'mo-')
axis equal
axis off
who
save ex1
clear C S t
whos
x=linspace(0,5,5); y=cos(x.^2);
figure(1); plot(x,y)
hold on; grid on
```

```

x=linspace(0,25,5); y=cos(x.^2); plot(x,y,'r')
figure(2); x=linspace(0,5,25); y=cos(x.^2); plot(x,y,'r')
x=linspace(0,5,1000); y=cos(x.^2); plot(x,y,'b')
load ex1
whos
clear
whos
t=0:pi/50:10*pi;
x= 5*cos(t/5) + cos(2*t);
y= 5*sin(t/5) + sin(3*t);
clf
plot(x,y); axis equal; title('Une figure de Lissajous')
axis off
x= 5*cos(t/5) - cos(2*t); y= 5*sin(t/5) - sin(3*t); plot(x,y,'c')
quit

```

Exercice 2

Créez cinq matrices A, B, C, D et E de dimensions respectives (15x10), (15x15), (10x10), (10x15) et (1x15). Faites faire par Matlab les calculs suivants :

A*A, B*B, C*C, D*D et E*E
A*A', A'*A, C'*C, C*C', A*A'-A'*A, C'*C-C*C'
A*D, D*A, A'D, A*D', A'*D', A*D- D*A, D*A-A*D'
A*E, A'*E, A'E', A'*E', B*E, B'*E, B'E', B'*E'
A.*A, E*E, E'*E, E*E', E.*E
C^3, C.^3

Inventoriez les opérations illicites et, pour chacune d'elles, interprétez les messages d'erreur produits par le logiciel. Pour les opérations qui produisent un résultat, expliquez-en la nature.

Exercice 3

Créez un vecteur colonne b ayant n composantes et une matrice carrée F .

Calculez la solution x du système $Fx = b$ à l'aide de l'opérateur \backslash .

Calculez la norme du vecteur $r = Fx - b$. Que pensez vous de la qualité du résultat ?

Exercice 4

Créez un vecteur colonne b ayant $2n$ composantes et une matrice rectangulaire F ayant $2n$ lignes et seulement n colonnes.

Calculez la solution x du système $F'Fx = F'b$ à l'aide de l'opérateur \backslash .

Calculez la norme du vecteur $r = F'Fx - F'b$.

Effectuez le calcul suivant : $z = F'b - x$. Que constatez-vous ?

Exercice 5

Calculez une valeur approchée des zéros des fonctions ci-dessous qui sont dans l'intervalle $[-5,5]$ au moyen des procédures `fzero` et/ou `roots`.

$$e^x \sin(2x) - e^{2x} \cos(x)$$

$$x^3 - 9x^2 + 15x - 7$$

Exercice 6

Calculez une approximation des intégrales des fonctions de l'exercice 5 sur l'intervalle $[-5,3]$.

Corrigé partiel

Le premier exercice est très proche du premier exercice de la séance 1 de l'an dernier (certaines instructions de l'an dernier ont été supprimées).

L'exercice 2 est identique à l'exercice 2 de la séance 1 de l'an dernier.

Les exercices 3, 4, 5 et 6 sont nouveaux voici un copié collé de la fenêtre de commande où les exercices sont résolus en séquence

```
>> b=(1:40)'; F=rand(40); x=F\b
```

```
x =  
 28.5420  
-27.4877  
 57.0313  
-205.3043  
 18.6652  
218.2448  
 68.4399  
-34.4816  
 30.1533  
-77.6108  
164.0087  
-105.8189  
 84.7054  
-129.3757  
 2.2845  
-59.5741  
131.1585  
 57.7196  
 12.0801  
-64.7908  
-19.9782  
169.2681  
 -6.4283  
-83.2875  
-11.4742  
-86.0418  
-54.7032  
-99.9983  
168.4293  
 41.6732  
-161.6767  
 56.9298  
 40.8772  
-68.6595  
-37.7753  
-36.4323  
-22.8558  
 29.3342  
 24.4312  
 13.5748
```

```
>> r=norm(F*x-b)
```

```
r =  
 4.2905e-013
```

```

>> clear x b F; b=(-10:5)'; F=rand(16,8); x=(F'*F)\(F'*b)
x =
    -2.8837
     4.8552
    -3.8572
    -6.6667
     4.2818
     0.1434
    -2.2071
     2.4854
>> norm((F'*F)*x-F'*b)
ans =
    1.1783e-014
>> z=F\b-x
z =
    1.0e-013 *
    -0.0089
    -0.3286
     0.0888
     0.3464
    -0.2043
    -0.0494
     0.1554
    -0.0400

>> g=@(x) exp(x).*sin(2*x) - exp(2*x).*cos(x)
g =
    @(x)exp(x).*sin(2*x)-exp(2*x).*cos(x)

>> clear x; x=-5:0.001:5;plot(x,g(x)) ; grid on
>> format long; fzero(g,4.7)
ans =
    4.712388980384690

>> format long; fzero(g,4.7,1e-14)
ans =
    4.712388980384690

>> clear x; x=-5:0.001:2;plot(x,g(x)) ; grid on
>> fzero(g,1.5)
ans =
    1.570796326794896

>> clear x; x=-5:0.001:-1;plot(x,g(x)); grid on
>> fzero(g,-1.5)
ans =
   -1.570796326794897

>> fzero(g,-3.2)
ans =
   -3.162748870926365
>> fzero(g,-4.7)
ans =
   -4.712388980384690

```

```

>> p=[1 -9 15 -7]; roots(p)
ans =
    7.000000000000000
    0.999999983124572
    1.000000016875429

>> h=@(x) x.*(x.*(x-9)+15)-7
h =
    @(x)x.*(x.*(x-9)+15)-7

>> x=-5:0.001:5; plot(x,h(x)); grid on
>> fzero(h,1.001)
ans =
    7

>> fzero(h,1.000001)
ans =
    7

>> fzero(h,1.00000001)
ans =
    1.000000010000000

fzero(h,1)
ans =
    1

quad(g,-5,3)
ans =
    1.395305793350730e+002

>> quad(g,-5,3,1e-14)
ans =
    1.395305794045108e+002

>> quad(h,-5,3)
ans =
   -768

>> quad(h,-5,3,1e-14)
ans =
   -768

```

L'erreur à ne pas commettre : oublier de vectoriser

```

>> g=@(x) exp(x)*sin(2*x) - exp(2*x)*cos(x)

g =
    @(x)exp(x)*sin(2*x)-exp(2*x)*cos(x)

>> fzero(g,4.7)
ans =
    4.712388980384690

```

ici, pas de problème

```
>> quad(g,-5,3) ici, ça foire !
??? Error using ==> mtimes
Inner matrix dimensions must agree.

Error in ==> @(x)exp(x)*sin(2*x)-exp(2*x)*cos(x)

Error in ==> quad at 77
y = f(x, varargin{:});
```