

Matlab : Langage interprété

Expressions arithmétiques.

```
>> 2+2  
ans = 4
```

```
>> 5^2  
ans = 25
```

Variables Matlab.

```
>> ans+1  
ans = 26
```

```
>> x = pi/4  
x = 0.7854
```

Point-virgule : pas d'affichage.

```
>> x = pi/2; y = sin(x);
```

```
>> z = 3 + 2*i  
z = 3.0000 + 2.0000i
```

```
>> r = abs(z);  
>> theta = angle(z);  
>> y = r*exp(i*theta);
```

Toute variable Matlab est une matrice.

- Scalaire : matrice 1×1 ;
- Vecteur : matrice $1 \times N$ ou $N \times 1$.

```
>> x = [ -1.3, sqrt(3), (1+2+3)*4/5 ]  
x = -1.3000 1.7321 4.8000
```

```
>> x(2)  
ans = 1.7321
```

```
>> x(5) = abs(x(1))  
x = -1.3000 1.7321 4.8000 0.0000 1.3000
```

Matrices

```
>> A = [ 1, 2, 3 ; 4, 5, 6 ; 7, 8, 9 ]
```

```
A =  1 2 3  
     4 5 6  
     7 8 9
```

```
>> eye(2,3)
```

```
ans = 1 0 0  
      0 1 0
```

```
>> x = ones(2,5)
```

```
x = 1 1 1 1 1  
    1 1 1 1 1
```

```
>> size(x)      % taille d'une matrice
```

```
ans = 2  5
```

```
>> size(x,1)    % nombre de lignes
```

```
ans = 2
```

```
>> r1 = [10, 11, 12];
```

```
>> A = [A ; r1]      % rajout d'une ligne
```

```
A =  1  2  3  
     4  5  6  
     7  8  9  
    10 11 12
```

```
>> r2 = zeros(4,1);
```

```
>> A = [A, r2]      % rajout d'une colonne
```

```
A =  1  2  3  0  
     4  5  6  0  
     7  8  9  0  
    10 11 12  0
```

```
>> A'              % Transposée conjuguée
```

```
>> A.'             % Transposée
```

```
>> flipud(A)       % Retournement vertical
```

```
>> fliplr(A)       % Retournement horizontal
```

```
>> rot90(A)        % Rotation de 90 degrés
```

❑ Remise en forme.

```
>> B = [ 1 2 3 4 5 6 ];  
>> B = reshape(B,2,3)      % 2 lignes et 3 colonnes  
B = 1 3 5  
    2 4 6
```

❑ L'opérateur «:» d'énumération.

```
>> x = 0.5:0.1:0.85  
x = 0.5000 0.6000 0.7000 0.8000  
  
>> x = 1:5      % Le pas vaut 1 par défaut  
x = 1 2 3 4 5
```

❑ Sélection d'éléments dans une matrice.

```
>> A(1,3)      % Première ligne, troisième colonne  
>> A(1,1:3)    % Première ligne, trois premières colonnes  
>> A(1,:)      % Toute la première ligne  
>> A(:,3)      % Toute la troisième colonne  
>> A(:)        % Vecteur colonne contenant tous les éléments de A.
```

Extraction de sous-matrices

```
>> A = [ 1, 2, 3 ; 4, 5, 6 ; 7, 8, 9 ]  
A = 1 2 3  
    4 5 6  
    7 8 9
```

```
>> indi = [3, 2, 1];  
>> indj = [1, 3];  
>> A(indi,indj)  
ans = 7 9  
      4 6  
      1 3
```

```
>> indi = [ 3, 0, 2 ];  
>> % provoque un bug car les indices doivent etre > 0 :
```

```
>> A(indi,:)  
??? Index into matrix is negative or zero. See release notes on  
changes to logical indices
```

Opérations matricielles

```
>> 2*A      % Produit par un scalaire
>> A*B      % Produit de deux matrices
>> A^p      % Élévation à la puissance p
>> A.*B     % Produit élément par élément de deux matrices
```

Attention : $A*A$ est différent de $A.*A$...

☐ Systèmes linéaires et inversion de matrices.

```
>> inv(A)    % Inversion d'une matrice carrée inversible
>> x = A\B    % Donne la solution de  $A*x = B$ 
>> x = B/A    % Donne la solution de  $x*A = B$ 
>> C = A./B   % Division éléments par éléments
```

☐ Autres fonctions matricielles.

```
>> det(A)
>> trace(A)
>> [V,D] = eig(A); % Valeurs propres et vecteurs propres
```

Matrices creuses

On ne conserve en mémoire que les éléments non nuls de la matrice.

```
>> A = eye(2); % matrice pleine
A=
    1    0
    0    1

>> B = sparse(A) % matrice creuse
B =
(1,1) 1
(2,2) 1

>> A*[ 1; 1];
ans =
    1
    1

>> B*[ 1; 1]; % résultat identique
ans =
    1
    1
```

```
>> x = 0:0.1:2; y = sin(x*pi);
>> figure(1), plot(x*pi,y)
>> title('Courbe y = sinus(pi*x)'), xlabel('x'), ylabel('y');

>> x = 1:0.1:2; y = -1:0.1:1;
>> [X,Y] = meshgrid(x,y);
>> figure(2), mesh(X,Y,cos(pi*X).*sin(pi*Y))
>> title('z=cos(pi*x)*sin(pi*y)'), xlabel('x');ylabel('y');zlabel('z');
```

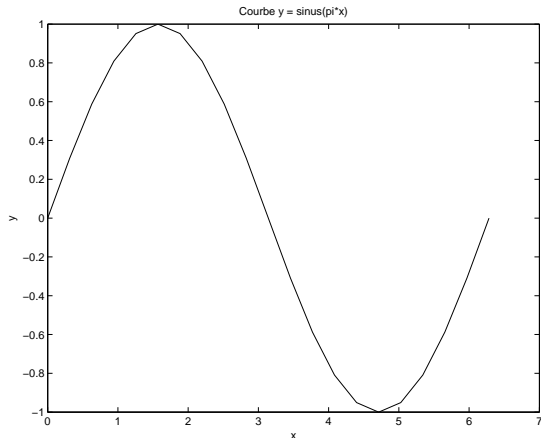


Figure 1 : Courbe 1D

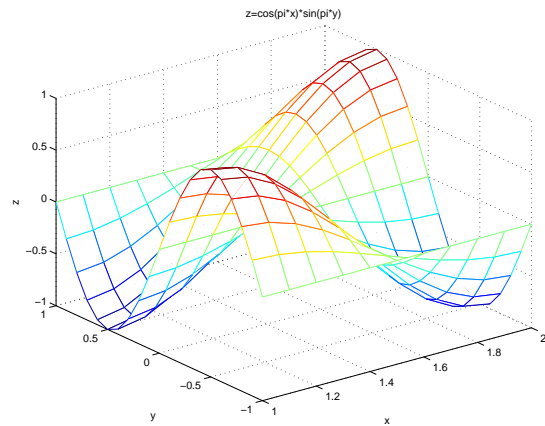


Figure 2 : Surface 2D

Variables Matlab, sauvegarde et chargement

```
>> save test.mat A, x, y;      % sauvegarde dans un fichier
>> load test.mat;             % chargement d'un fichier
>> whos                        % liste des variables
```

Name	Size	Bytes	Class
A	2x2	32	double array
B	2x2	36	sparse array
X	21x11	1848	double array
ans	2x1	16	double array

❑ Un script Matlab est :

- un fichier (script) `nom_fich.m` contenant la séquence d'instructions Matlab.
- commentaires **obligatoires** (commençant par %)
- Pour l'exécution sous Matlab, lancer :

```
>> nom_fich
```

Fonctions

❑ Créer un fichier `sinuscardinal.m` contenant :

```
function y = sinuscardinal(x)
z = sin(x);    % Variable de stockage
y = z./x;      % Résultat de la fonction
```

❑ Appel de la fonction :

```
>> sincpi = sinuscardinal(pi)
ans = 3.8982e-17
```

```
>> sinc0 = sinuscardinal(0)
ans = NaN
```

❑ Valeurs limites :

```
>> 1/0          % plus l'infini
ans = Inf
>> -1/0         % moins l'infini
ans = -Inf
>> 0/0          % Not a Number
ans = NaN
```

Boucles et contrôles

❑ Contrôle if.

```
if (abs(x) < 1e-10),
    y=1.;
else,
    y=sin(x)/x;
end;
```

Opérateurs logiques : et (&), ou (|), égal (==), supérieur (>,>=), inférieur(<,<=), non(~).

❑ Boucle for (à éviter au maximum).

```
for i=1:N,
    x(i) = i;
end;
```

est équivalente à :

```
x = 1:N;
```

```
>> help inv
  INV      Matrix inverse.
  INV(X) is the inverse of the square matrix X.
  .....

>> lookfor inverse
ACOS      Inverse cosine.
ASIN      Inverse sine.
ASINH     Inverse hyperbolic sine.
ATAN      Inverse tangent.
ATANH     Inverse hyperbolic tangent.
ERFCINV   Inverse complementary error function.
ERFINV    Inverse error function.
INV       Matrix inverse.
.....
```