

Syntaxe et commandes Scilab

Sauf cas assez particuliers, tous les objets manipulés par Scilab sont des matrices de réels. Notamment, les réels sont vus comme des matrices de taille 1×1 et les vecteurs lignes comme des matrices de taille $1 \times n$.

Pour accéder à la documentation d'une commande on peut utiliser la commande `help unecommande` ou aller dans l'aide du logiciel : . Pensez à aller voir les exemples à la fin de la documentation !

DÉFINITION D'UNE MATRICE (OU D'UN VECTEUR)

| | |
|--|---|
| <code>[1 2 3 4]</code> ou <code>[1,2,3,4]</code> | Définition d'un vecteur ligne |
| <code>[1;2;3;4]</code> | Définition d'un vecteur colonne |
| <code>[1 2 ; 3 4]</code> | Définition d'une matrice |
| <code>[]</code> | La matrice vide (de taille 0×0) |
| <code>[1:10]</code> | Vecteurs des entiers de 1 à 10 |
| <code>[0:0.1:2]</code> | Vecteurs des réels de 0 à 2 avec un pas de 0,1 |
| <code>zeros(3,4)</code> | Matrice nulle à trois lignes et quatre colonnes |
| <code>ones(3,4)</code> | Matrice de 1 à trois lignes et quatre colonnes |
| <code>eye(4,4)</code> | Matrice identité de taille quatre |
| <code>diag([1 2 3 4])</code> | Matrice diagonale de taille quatre, avec les éléments 1, 2, 3 et 4 sur la diagonale |

UTILISATION DES MATRICES

| | |
|-----------------------------|--|
| <code>A(2,3)</code> | Élément situé sur la deuxième ligne et la troisième colonne de A |
| <code>A(2)</code> | Deuxième élément du vecteur A (si A est un vecteur) |
| <code>A(2,\$)</code> | Dernier élément de la deuxième ligne de A |
| <code>A(\$)</code> | Dernier élément du vecteur A (si A est un vecteur) |
| <code>A(\$-1,3)</code> | Troisième élément de l'avant-dernière ligne de A |
| <code>A(2,:)</code> | Deuxième ligne de A |
| <code>A([1,3],[2,4])</code> | Sous-matrice de A constituée des éléments situés à la fois sur la première ou troisième ligne de A et sur sa deuxième ou quatrième colonne |
| <code>length(A)</code> | Nombre d'éléments dans la matrice A |
| <code>size(A)</code> | Taille de la matrice A sous la forme <code>[n,m]</code> avec n le nombre de lignes et m le nombre de colonnes |

OPÉRATIONS SUR LES MATRICES

| | |
|-----------------------|--|
| <code>A'</code> | Transposée de A |
| <code>A+B</code> | Somme des matrices A et B |
| <code>A*B</code> | Produit (matriciel) des matrices A et B |
| <code>A.*B</code> | Produit terme à terme des matrices A et B |
| <code>A^2</code> | Carré de la matrice A |
| <code>A^-1</code> | Inverse de la matrice A |
| <code>A.^2</code> | Matrice dont les éléments sont les carrés des éléments de la matrice A |
| <code>A\b</code> | Solution x du système linéaire $Ax = b$ |
| <code>exp(A)</code> | Matrice dont les coefficients sont les exponentielles de ceux de A |
| <code>expm(A)</code> | Exponentielle matricielle de A |
| <code>A=[A,12]</code> | Ajoute l'élément 12 à la fin du vecteur A |
| <code>[A,B]</code> | Concaténation de deux vecteurs/matrices A et B |

DÉFINITION DE FONCTIONS

Il existe deux méthodes pour définir des fonctions en scilab : la première peut être utilisée dans la console et la seconde dans un fichier. On utilisera majoritairement la seconde méthode.

| | |
|--|---|
| <code>def(' [z]=somme(x,y)', 'z=x+y')</code> | Définition de la fonction <code>somme</code> qui prend en paramètre deux nombres/vecteurs/matrices <code>x</code> et <code>y</code> et renvoie leur somme |
|--|---|

| | |
|--|---|
| <pre>function z = somme(x,y) z = x+y endfunction</pre> | Définition de la fonction <code>somme</code> qui prend en paramètre deux nombres/vecteurs/matrices <code>x</code> et <code>y</code> et renvoie leur somme |
|--|---|

On peut ensuite utiliser la fonction `somme` en console ou dans un fichier avec par exemple la commande `somme(2,3)` ou `somme([1 2 3],[4 5 6])`

SYNTAXE POUR LES BOUCLES ET LES CONDITIONS

| | | |
|--|---|---|
| <pre>x=rand() a=0 if x < 1/3 a=1 elseif x < 2/3 a=2 else a=3 end</pre> | <pre>n=0 for i=1:10 n=n+1 end</pre> | <pre>n=0 while n^2 < 100 n=n+1 end</pre> |
|--|---|---|

OPÉRATEURS LOGIQUES

| | |
|-------------------------|--------------------------|
| <code>1==2</code> | Teste si $1 = 2$ |
| <code>1<=2</code> | Teste si $1 \leq 2$ |
| <code>1<2</code> | Teste si $1 < 2$ |
| <code>1>=2</code> | Teste si $1 \geq 2$ |
| <code>1>2</code> | Teste si $1 > 2$ |
| <code>1<>2</code> | Teste si $1 \neq 2$ |
| <code>&</code> | Opérateur Booléen « et » |
| <code> </code> | Opérateur Booléen « ou » |

AFFICHAGE GRAPHIQUE

| | |
|---|--|
| <code>scf(2)</code> | Définit la fenêtre 2 comme fenêtre graphique courante |
| <code>clf()</code> | Efface la fenêtre graphique courante |
| <code>plot2d(X,f(X))</code> | Si <code>X</code> est un vecteur (par exemple <code>X=[0:0.1:10]</code>) et si <code>f</code> est une fonction réelle, <code>plot2d(X,f(X))</code> trace les points de coordonnée $(x_i, f(x_i))$ et les relie pour former une courbe |
| <code>plot2d(X,cos(X),-3,rect=[0,-2,10,2])</code> | Le troisième paramètre définit le style des points et le quatrième définit la fenêtre (voir documentation) |

DIVERS

| | |
|-----------------------------|--|
| <code>%i</code> | Le nombre complexe i |
| <code>%e</code> | Le nombre $\exp(1)$ |
| <code>%pi</code> | Le nombre π |
| <code>abs(a)</code> | La valeur absolue de a |
| <code>floor(a)</code> | La partie entière de a |
| <code>sqrt(a)</code> | La racine carrée de a |
| <code>modulo(a,n)</code> | Le reste de la division euclidienne de a par n |
| <code>rand()</code> | Nombre tiré aléatoirement (de manière uniforme) dans l'intervalle $[0, 1]$ |
| <code>linspace(a,b)</code> | Vecteur de nombres équidistants entre les bornes a et b |
| <code>ode(y0,t0,t,f)</code> | Solution numérique de l'équation différentielle $y' = f(t,y)$, $y(t_0) = y_0$ |
| <code>format(10)</code> | Configure le nombre de chiffre à afficher pour les nombres décimaux |
| <code>disp(a)</code> | Affiche le contenu de la variable a . Utile dans les fichiers et pour déboguer les fonctions |
| <code>exec('tp.sci')</code> | Exécute les commandes contenues dans le fichier <code>tp.sci</code> |
| <code>abort()</code> | Annule un calcul préalablement interrompu par <i>Ctrl-C</i> |
| <code>resume()</code> | Relance un calcul préalablement interrompu par <i>Ctrl-C</i> |