Chapitre 3 : Les fonctions

En plus des fonctions usuelles décrites au chapitre 1 (logarithme, sinus …), on peut définir de nouvelles fonctions.

I Définition d'une fonction (notion importante)

La déclaration d'une fonction d'une seule variable se fait avec la syntaxe suivante :

```
function [y]=nom_de_la_fonction(x)
...(instructions)
endfunction
```

Une fonction peut être insérée dans un script ou utilisée sur la console. Voici un exemple :

```
-->function [y]=cube(x),y=x^3,endfunction

-->cube(3)
 ans  =
    27.
```

RQ : x est appelée variable d'entrée et y variable de sortie. Il peut y avoir plusieurs variables en entrée et/ou en sortie, la syntaxe plus générale d'une fonction est donc :

```
function [y1,y2,...,yn]=nom_de_la_fonction(x1,x2,...,xp)
...(instructions)
endfunction
```

Voici un autre exemple :

```
--> function[z]=f(x,y),z=abs(x)+sqrt(y),endfunction

--> f(-1,4)
 ans  =
   3.
```

On peut même n'avoir aucune variable en sortie : on met les crochets et rien entre [ ] , et/ou aucune variable en entrée : on met les parenthèses et rien entre () .

```
-->function [ ]=affichage(), disp('Bonjour!'),endfunction
-->affichage
  Bonjour!
```

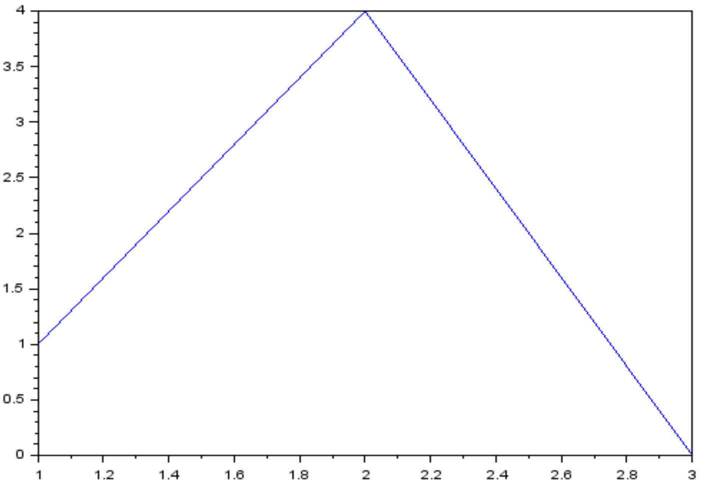
Revenons au cas courant des fonctions numériques d'une seule variable réelle et voyons comment tracer le graphe d'une telle fonction.

II Tracé du graphe d'une fonction : commande PLOT

Pour deux variables x et y contenant des vecteurs lignes (ou matrice ligne) de même dimension : $x=(x_1, x_2, \dots, x_n)$ et $y=(y_1, y_2, \dots, y_n)$, l'instruction `plot(x,y)` commande le tracé du graphe obtenu en reliant les points de coordonnées (x_i, y_i) par des lignes.

Ex : `-->x=[1,2,3];y=[1,4,0];plot(x,y)`

Scilab affiche dans une nouvelle fenêtre la figure suivante :



Pour tracer le graphe d'une fonction f sur un intervalle I, le principe est de relier les points de la forme (x_i , $f(x_i)$) en choisissant les points x_i suffisamment rapprochés pour donner l'illusion d'un tracé de courbe (en réalité, Scilab trace une succession de petits segments).

Il faut donc fournir au logiciel les abscisses x=(x_1 , x_2 , … , x_n) correspondant à une subdivision du segment I, avec un pas assez petit.

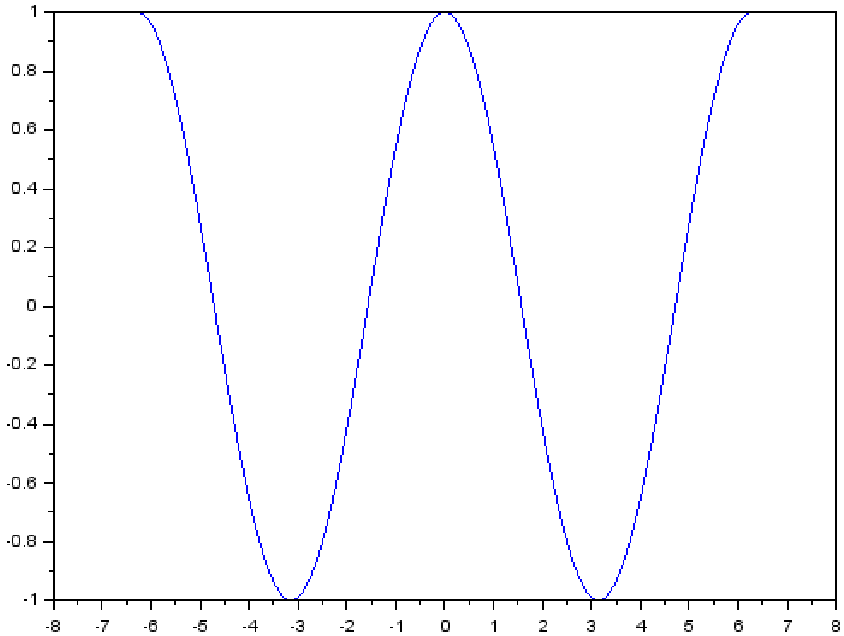
Il y a <u>deux possibilités</u> : `x=min : pas : max` ,
ou `x=linspace(min,max,nombre de points)`

Par exemple, pour subdiviser le segment I=[0,1] en 100 sous-segments, on a les deux commandes équivalentes : `x=0:0.01:1`
ou `x=linspace(0,1,101)`,
qui mettent dans x le vecteur ligne (0 , 0.01 , 0.02 … 0.99 , 1). (attention 101 correspond au nombre de valeurs attendues dans le vecteur).

Ensuite, il faut fournir à Scilab les ordonnées y=($f(x_1)$, $f(x_2)$, … , $f(x_n)$).
Il y a plusieurs solutions ; celle qui me semble la plus sure est l'utilisation de la commande **FEVAL** , avec la syntaxe : `y=feval(x,f)`
Mais attention ! La fonction f doit obligatoirement avoir été définie préalablement, même si c'est une fonction usuelle (f doit être une fonction externe à Scilab) ! (voir l'exemple ci-dessous)
Cette commande (feval) n'est pas toujours indispensable, et dans certains cas simples, elle alourdit un peu le script, mais au moins on est sur que ça marche.
On pourra ainsi utiliser également une écrire directe du vecteur y si celui n'utilise que des commandes de base : par exemple si la fonction est définie par $f(x)=x^2+1$ alors on peut poser y=x^2+1 qui appliquera la fonction f a tous les éléments du vecteur x.

On poursuit par l'instruction `plot(x,y)`.

Ex :
```
-->function [y]=f(x),y=cos(x),endfunction
-->x=linspace(-2*%pi,2*%pi,201);y=feval(x,f);plot(x,y)
```
Ex : `x=linspace(0,1,101) ; y=x^2+1 ; plot(x,y)`



On obtient le même résultat avec :

```
--> x=linspace(-2*%pi,2*%pi,201);plot(x,cos(x))
```

remarque : il existe des options permettant de rajouter un titre, d'agir sur le trait (couleur, épaisseur), les axes. On pourra se reporter à l'aide de Scilab pour les utiliser (taper `help plot`).

On peut tracer sur une même figure plusieurs courbes. Par exemple, la commande suivante permet d'obtenir sur la même figure le graphe de la fonction ln et la droite d'équation y=x-1 sur I=[0,5 ;1,5].

```
-->x=linspace(0.5,1.5,500);plot(x,log(x),'r');plot(x,x-1,'c')
```

Essayez ! ('r' et 'c' servent à choisir une couleur)

Pour terminer sur le tracé des graphes, mentionnons que ça peut aussi être fait avec la commande PLOT2d.

TP 3 : La commande `clf` permet d'effacer la figure. Celle-ci est également effacée si on ferme le fenêtre de dessin. L'utilisation de Scinotes n'est pas nécessaire jusqu'à l'exercice 6, mais ce n'est pas interdit !

Ex 1 : Avec cette ligne de commande, quelle est la figure obtenue ? Essayer de répondre d'abord sans tester en console (commencer par décrire les valeurs qu'on a dans x). Vérifer ensuite.

---> x=0 : 1 : 4 ; plot(x,(1-x)^2)

Ex 2 : Écrire un script qui définit les fonctions $f: x \rightarrow e^x + e^{-x}$ et $g: x \rightarrow e^x - e^{-x}$ puis qui trace leur graphe pour x entre -3 et 3 sur une même figure. Choisir deux couleurs différentes pour différencier les courbes ('r' pour red, 'g' pour green par exemple)
Démontrer trois observations que l'on fait sur cette figure :

- Une sur la position relative des 2 courbes.
- Une sur les antécédents de 0 par f et une autre sur ceux de 0 par g.
- Facultatif : montrer que 3 a deux antécédents par f que l'on précisera.

Ex 3 : Tracer le graphe de la fonction partie entière dans l'intervalle [-3,3]. Que remarquez-vous qui n'est pas comme le graphe que vous auriez tracé vous même ?

Ex 4 : Définir en console la fonction que vous appellerez f qui à x associe $f(x) = \frac{\ln(1+x)}{x}$.
Lancer ensuite la commande :

---> x=linspace(0.001,3,500);y=feval(x,f) ; plot2d(x,y,rect=[0,0,3,1])

Qu'observe-t-on comme limite de f lorsque x tend vers 0 (par valeurs positives) ?

Ex 5 : Représentation graphique d'une suite.
Soit la suite définie pour tout entier n strictement positif par $U_n = \left(1 + \frac{1}{n}\right)^n$
Représenter les points de coordonnées (n, U_n) pour n allant de 1 à 50.
<u>Indication</u> : pour éviter que Scilab ne trace des lignes entre deux points, utilisez la syntaxe `plot(n,z,'xr')` avec laquelle chaque point de coordonnées $(n_i, z_i = f(n_i))$ sera matérialisé par une croix (x) rouge (r) .

Que conjecturez-vous sur la convergence de cette suite ?
Montrer le. (On pourra utiliser le logarithme).

Ex 6 : Soit n nombres réels strictement positifs.
Leur moyenne <u>arithmétique</u> est la moyenne usuelle, c'est -à-dire leur somme divisée par n.
Leur moyenne <u>géométrique</u> est la racine n-ième du produit des n nombres.
Leur moyenne <u>harmonique</u> est l'inverse de la moyenne arithmétique de leurs inverses.
Leur moyenne <u>quadratique</u> est la racine carrée de la moyenne arithmétique de leurs carrés.

Ecrire un script qui saisit trois réels > 0 dans des variables nommées x,y et z, puis qui calcule les quatre moyennes ci-dessus grâce à des fonctions nommées arith, geo, harmo et quad, puis qui affiche les résultats avec un message indicatif.
<u>On impose que les trois dernières fonctions utilisent la fonction arith</u>.

Faire tourner plusieurs fois pour différentes saisies de x,y et z et constater qu'à chaque fois, le classement par ordre croissant de ces moyennes est le même.