

Séance 4 : Exercices Avancés sur les fonctions Scilab

Vous trouverez toutes les informations et les outils sur scilab, sur le site : <http://www.scilab.org/>

1.1 Rappel :

1.1.1 Aide en ligne Scilab

La première prise en main du logiciel passe aussi par la découverte de l'aide en ligne dont le recours sera ensuite constant. Cette aide, entièrement en anglais, est accessible à partir de l'option *Help* du menu. Elle se présente sous la forme d'une liste de fichiers d'aide pour chaque instruction disponible sous Scilab. Pour effectuer une recherche par mot clé dans l'index (par exemple, trouver la fonction qui retourne le cosinus hyperbolique), on peut taper `apropos cosine` dans la fenêtre de commande (et découvrir que la fonction s'appelle `cosh`). Au contraire, lorsqu'une instruction est connue (par exemple `rand`), il est possible d'accéder directement à l'aide de celle-ci en tapant `help rand` sur la ligne de commande.

1.1.2 Première fonction Scilab

Les fonctions sous scilab peuvent prendre divers types d'arguments (scalaire, matrice, etc...) et peuvent réaliser des opérations mathématiques plus ou moins complexes. Attention à la syntaxe et aux mots clefs en caractères gras (Voir le cours).

L'entête d'une fonction qui retourne n paramètres et prend m valeurs en entrée, s'écrit :

```
function [y1, ..., yn]=foo(x1, ..., xm)
```

Le corps de la fonction est la suite d'instructions entre l'entête et le mot clef `endfunction`.

Exemple :

```
function [x, y]=myfct(a, b)
x=a+b
y=a-b
endfunction
```

Une fois le fichier `.sci` dans lequel est déclarée la fonction chargé, cette dernière peut s'appeler ainsi :

```
[x, y]=myfct(3, 2)
```

ou encore

```
a=5
b=6
[x, y]=myfct(a, b)
```

1.2 Exercices Avancés sur les fonctions

1.2.1 Exercice 1

On sait que si f est une fonction continue strictement monotone sur un intervalle $[a;b]$ et que $f(a)$ et $f(b)$ sont de signes contraires, l'équation $f(x) = 0$ admet une racine unique α appartenant à $]a;b[$.

Par dichotomie, on peut obtenir une suite de nombres convergeant vers α . L'algorithme en est le suivant :

- on connaît au départ : - la fonction f (supposée continue strictement monotone),
- les réels a et b tels que $f(a) \times f(b)$ soit négatif,

Séance 4 : Exercices Avancés sur les fonctions Scilab

- une constante epsilon indiquant la précision attendue.

- la variable auxiliaire c désignera la demi-somme de a et b .
- algorithme :

début

tant que $f((a+b)/2) > \text{epsilon}$

debut

$c \leftarrow (a+b)/2$; //afficher(c) ici si on désire toutes les approximations

si $f(a)*f(c) < 0$ alors $b \leftarrow c$ sinon $a \leftarrow c$;

fin

afficher($(a+b)/2$); //ou retourner($(a+b)/2$) s'il s'agit d'une fonction

fin

- 1) Ecrire une fonction $[y]=f(x)$ permettant la déclaration de la fonction f définie par $\cos(x) - x$
- 2) Ecrire une fonction **DICHO** réalisant l'algorithme précédent, a et b étant reçus en paramètres entrants et c comme paramètre de sortie.

Ecrire la ou les instructions sous Scilab permettant d'obtenir une solution de $\cos(x) = x$ à 10^{-10} près.

1.2.2 Exercice 2

On démontre que la série de somme partielle $S_n = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$ tend vers $+\infty$ lorsque n tend vers $+\infty$.

Le problème consiste à trouver le nombre de termes de cette série, dite harmonique, nécessaire au dépassement d'une valeur arbitraire donnée :

$$1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} > \text{valeur}$$

- 1) Ecrire une fonction $[s]=\text{harmonic}(n)$ qui, pour un entier n passé en paramètre, calcule la somme $s = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$.

- 2) Ecrire un programme dans Scilab

- demandant à l'utilisateur une valeur strictement plus grande que 1 et plus petite ou égale à 8 (et répétant la saisie jusqu'à ce que la condition soit satisfaite)
- affichant le nombre de termes nécessaire au dépassement de cette valeur et le résultat de la somme (vous pourrez utiliser la fonction **harmonic** définie ci-dessus).

1.2.3 Exercice 3

- 1) Ecrire une fonction **PPDP** renvoyant le Plus Petit Diviseur Premier d'un entier au moins égal à 2 reçu en paramètre.
- 2) Ecrire une fonction **DECOMPOSER** affichant à l'écran la décomposition en facteurs premiers d'un entier au moins égal à 2 reçu en paramètre. Cette procédure devra appeler la fonction précédente. L'affichage devra ressembler à ceci :

$$360 = 2 * 2 * 2 * 3 * 3 * 5 \quad \text{ou à} : 360 = 2^3 * 3^2 * 5.$$

- 3) Ecrire une fonction **SAISIR** demandant à l'utilisateur un entier au moins égal à 2 (et répétant la saisie jusqu'à ce que cette condition soit satisfaite), puis renvoyant cet entier dans un paramètre (passé bien sûr par adresse).
- 4) Vous pourrez réaliser le programme suivant sous Scilab

Séance 4 : Exercices Avancés sur les fonctions Scilab

a = SAISIR()
 DECOMPOSER(a)

1.2.4 Exercice 4

La méthode de Monte Carlo pour obtenir une valeur approchée de π

Le principe de la méthode est le suivant :

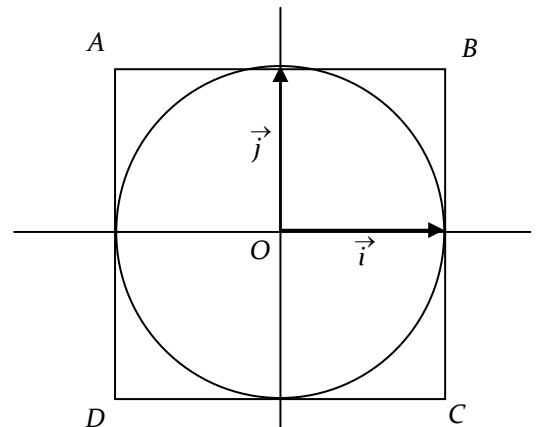
Considérons un disque de rayon 1 inscrit dans un carré (donc de côté 2). Pour plus de simplicité, considérons le disque de centre O , origine d'un repère orthonormal et le carré $ABCD$ défini comme sur la figure ci-contre :

La méthode de Monte Carlo repose sur le principe suivant : choisissons un grand nombre de points dans le carré. La proportion des points se trouvant dans le disque par rapport à ceux se trouvant dans le carré fournit une valeur approchée de la proportion de l'aire de ce disque par rapport à l'aire du carré soit ici $\frac{\pi}{4}$.

En utilisant la fonction `rand()` de Scilab, construire les coordonnées (dans les variables x et y) d'un point pris aléatoirement dans le carré.

Déterminer si ce point se trouve dans le disque (on rappelle que ce point se trouve dans le disque si $x^2 + y^2 \leq 1$).

Ecrire un programme Scilab qui répète cette expérience 100 fois et qui détermine le nombre de points se trouvant dans le disque puis fournir une valeur approchée de π .



2 Références :

Cette partie du TD reprend et s'inspire de nombreux supports comme :

- Le tutoriel de Vivien Frédéric, vivienfrederic.free.fr/Scilab/