

TP Scilab no 2

Dans cette séance de TP, on va se familiariser avec la programmation sous Scilab. Après avoir lancé Scilab, commencez donc par créer un document intitulé "TP2.sci" grâce à l'éditeur de textes de Scilab (Scinotes).

On peut aussi créer un document intitulé "TP2.sce" qui, lui, va contenir, les utilisations que l'on va faire des fonctions, exercice par exercice avec tous les commentaires que l'on veut. On n'oubliera pas de sauvegarder son travail régulièrement (toutes les lignes par exemple), conseil d'ami !

Avant de commencer à écrire une fonction, il faut s'astreindre à écrire les algorithmes correspondants en pseudo-langage, sur feuille, et les tester, à la main, sur un jeu d'exemples.

Le fichier "TP2.sci" va contenir un certain nombre de fonctions (on dit une bibliothèque) que vous allez faire lire à Scilab à chaque fois que vous vous voudrez les utiliser.

Quand vous aurez écrit une fonction dans "TP2.sci" , vous la chargerez dans Scilab.

- Pour charger uniquement la fonction que l'on vient de mettre au point, sélectionner à la souris l'ensemble de la fonction, puis bouton droit, "Evaluer la sélection", ou encore menu en haut "Exécuter" puis "Evaluer la sélection".

Cette opération oblige Scilab à lire la fonction. Il y aura peut-être (sûrement !) des messages d'erreur. Corrigez-les jusqu'à acceptation par Scilab. Testez ensuite la fonction sur des exemples simples pour vous assurer qu'elle fait bien ce qu'il faut. Vous pourrez ensuite l'utiliser n'importe quand pendant votre session.

- Pour charger tout le fichier "TP2.sci" et donc lire toutes les fonctions qui y sont inscrites : menu en haut "Exécuter" puis "Charger dans Scilab", ou touches "Ctrl L" ou encore taper `getf('TP2.sci')` dans la fenêtre Scilab.

_____ AU TRAVAIL _____

1. Ecrire une fonction `echange(x,y)` qui prend en entrée un couple de valeurs et, en sortie, renvoie ce couple dans l'ordre inverse.
2. Ecrire une fonction `pgcd(a,b)` qui prend pour arguments deux nombres entiers a, b et donne pour résultat le pgcd de ces deux nombres (en utilisant l'algorithme d'Euclide).
3. Ecrire une fonction `f`, qui prend comme argument un nombre entier n , et qui rend le nombre $\sum_{k=1}^n 1/k^2$ de différentes façons :
 - en construisant un vecteur de taille n contenant les $1/k^2$ et en calculant la somme de ses termes.
 - en utilisant une variable locale s et en effectuant une boucle qui, à la k -ième itération, ajoute $1/k^2$ à s .Comparer ces deux fonctions du point de vue d'une part de la rapidité et d'autre part de l'occupation mémoire.
4. Ecrire une fonction `somme` qui prend comme argument une matrice M et qui calcule $\sum_{i,j} m_{i,j}$
 - en utilisant deux boucles imbriquées (sur i et sur j),
 - en remplaçant la boucle interne par une fonction interne de scilab (somme de ligne par exemple).
5. Dans cet exercice, on va chercher à calculer les coefficients binomiaux $\binom{n}{p}$. On rappelle qu'ils vérifient l'égalité

$$\binom{n}{p} = \binom{n-1}{p-1} + \binom{n-1}{p}$$

- a) Construire une fonction `lignesuivante`, qui prend comme argument un vecteur ligne u de taille quelconque n et rend un vecteur ligne v de taille $n+1$ dont la première et la dernière coordonnée valent 1 et dont les autres vérifient $v_p = u_p + u_{p-1}$.

- b) Construire une fonction `ligne`, prenant comme argument un entier n et rendant le vecteur ligne de taille $n + 1$, constitué des coefficients binomiaux $\binom{n}{p}$.
- c) Construire une fonction `tableau`, prenant comme argument un entier n et rendant un tableau $n + 1 \times n + 1$ dont les coefficients sont les $\binom{n}{p}$. Par exemple, on veut que la commande `tableau(3)` rende le tableau

$$\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 3 & 3 & 1 \end{array}$$

6. La formule de Machin (John Machin, mathématicien anglais, 1706)

$$\pi = 16\text{Arctan}\frac{1}{5} - 4\text{Arctan}\frac{1}{239}$$

permet de calculer π de manière très efficace, en utilisant le développement de arctan en séries entières

$$\text{Arctan}x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1} + \dots$$

- a) Ecrire une fonction `arctan(x,h)`, qui prend en argument un réel x et une précision h et rend une valeur approchée de $\text{Arctan}x$ à h près calculée en tronquant le développement en série précédent.
- b) Ecrire une fonction `calculPi`, qui prend en argument un réel h et rend une valeur approchée de π à h près.
7. Ecrire une fonction `crible(n)` qui permet de sortir un tableau de tous les nombres premiers jusqu'à n en utilisant la méthode du crible d'Eratosthène.
8. Ecrire différentes façons de réaliser le tri d'un tableau (sans utiliser les fonctions intégrées de Scilab, qu'on peut rappeler).
- a) Ecrire une fonction `insertion(X)` qui prend comme argument un vecteur X et le rend trié, par ordre croissant, par la méthode d'insertion. On range le i -ième terme en le comparant aux $i - 1$ premiers. Commençant par $i = 2$, on compare $X(2)$ à $X(1)$ et on les échange s'il y a lieu ; puis, on compare $X(3)$ au 2 premiers termes et on le range par rapport à ceux-ci par des permutations éventuelles, etc...
- b) Ecrire une fonction `selection(X)` qui prend comme argument un vecteur X et le rend trié, en ordre croissant, par la méthode de sélection. On cherche le plus petit élément et on le met à la 1-ère place, puis on cherche le second plus petit, on le met à la 2-ème place, etc...
- c) Ecrire une fonction `rapide(X)` qui prend comme argument un vecteur X et le rend trié, en ordre croissant, par la méthode du tri rapide. Pour ce faire, on choisit un "pivot" et on casse X en deux vecteurs : le 1er est composé des éléments inférieurs au pivot, le 2ème des éléments plus grand que le pivot. Puis on réappelle la procédure sur chacun des vecteurs obtenus et ainsi de suite *récurivement*.