Chapitre 5 : Les tests logiques et l'instruction conditionnelle IF … THEN … ELSE … END

I Opérateurs logiques, booléens

Tapez sur la console de Scilab les instructions suivantes et observez les réponses :

```
-->a=2;b=4;a<b
-->c=(a==b)
-->(a<b)&(a>-b)
-->(a<>0) | (b==6)
-->d=(a<>0)&(b==6)
```

Les variables c et d sont appelées des booléens (du nom du logicien anglais Georges Boole). Ce type de variable correspond au résultat d'un test logique : « la proposition est-elle vraie ou fausse ? » et peut prendre deux valeurs T (true) ou F (false).
Les propositions sont écrites à l'aide des opérateurs suivants :

>	strictement supérieur à
<	strictement inférieur à
>=	supérieur ou égal à (≥ en maths)
<=	inférieur ou égal à (≤ en maths)
==	égal à (= en maths)
<>	différent de (≠ en maths)

et des connecteurs logiques :

&	ET
|	OU (inclusif, comme en maths)

<u>Exemple</u> : On considère l'équation ax+b=0 : que fait le script suivant ?

```
a =input ('entrez un réel a')
b=input ('entrez un réel b')
infinite = (a==0) & (b==0)
une = (a<>0)
zero = (a==0) & (b<>0)
disp (infinite , 'tous les réels sont solution ? ')
disp (une , 'il y a une unique solution ? ')
disp (zero , 'il n''y a aucune solution ?')
```

II L'instruction conditionnelle IF … THEN … ELSE … END

<u>Exemple</u> : le script précédent un peu modifié :

```
a=input('entrez un réel a ')
b=input('entrez un réel b ')
if (a==0)&(b==0) then disp('tous les réels sont solution.')
elseif (a==0)&(b<>0) then disp('il n''y a aucune solution.')
else disp(-b/a,'il y a une unique solution qui est : ')
end
```

Ce script donne l'ensemble des solutions de l'équation du premier degré $ax+b=0$ selon les valeurs de a et b . On y voit la syntaxe de la structure conditionnelle de base qui permet de donner des instructions différentes selon qu’un test est vrai ou faux :

IF condition **THEN** instruction_si_vrai
ELSE instructions_si_faux
END

ou, s'il y a plus d'alternatives :

IF condition1 **THEN** instruction_si_condition1-vraie
ELSEIF condition2 **THEN** instructions_si_condition2-vraie
ELSE instructions_si_condition1&condition2-fausses
END

La dernière alternative commençant par ELSE n'est pas obligatoire.

TP 5 :

Ex 1 : Que fait ce programme ?

```
a1=input('Donner la valeur du premier nombre : ')
a2=input('Donner la valeur du second nombre : ')
a3=input('Donner la valeur du troisième nombre : ')
m=a1
if a2<m then m=a2
end
if a3<m then m=a3
end
disp(m)
```

Ex1 bis : Construire un script pour qui prend en entrées les 3 longueurs d'un triangle et donne en sortie un texte signalant si le triangle est rectangle ou non, et dans le cas où il serait rectangle, signale de plus si il est isocèle.
Tester ce programme avec 1,1 et $\sqrt{2}$. Comment le modifier pour palier à ce problème ?

Ex 2 : Soit le script :

```
a=-10;b=2
for i=1:4
        c=(a+b)/2
        if a*c<0 then b=c
        else a=c
        end
end
```

Réaliser un tableau à 5 colonnes montrant le contenu des variables i, c, ac<0 (booléen), b et a au cours de la boucle.

Ex 3 : Ecrire un script qui saisit trois réels : a non nul, b et c , puis qui détermine et affiche l'ensemble des solutions de l'équation $ax^2+bx+c=0$ dans $\mathbb{R}$.

Ex 4 : Suites de Syracuse.
Ce sont des suites célèbres définies par un premier terme u_0 qui est un entier strictement positif,
puis pour tout $n\in\mathbb{N}$, $u_{n+1}=\frac{u_n}{2}$ si u_n est pair et $u_{n+1}=3u_n+1$ si u_n est impair.

1) On prend $u_0=6.$ Calculer les 15 premiers termes de cette suite.
2) Montrer que pour tout entier naturel n, on a : n est pair $\Leftrightarrow n=2\lfloor n/2\rfloor$.
3) Utiliser le résultat qui précède pour écrire un script permettant de calculer le terme d'indice n d'une suite de Syracuse (u_n) définie par : $u_0=a$, a étant un entier naturel non nul entré par l'utilisateur.
   L'entier n sera aussi demandé à l'utilisateur. Faire afficher tous les termes de la suite depuis u_0 jusqu'à u_n .
4) Faire des essais pour voir que la suite finit toujours par prendre la valeur 1. Ne pas essayer de démontrer ce résultat qui reste pour l'instant une conjecture, aussi simple soit-elle !

Ex 5 : Soit f la fonction carré de $\mathbb{R}$ vers $\mathbb{R}$ et g la fonction de $\mathbb{R}$ vers $\mathbb{R}$ définie par $g(x)=x$ si $x<0$ et $g(x)=x+1$ si $x\geq 0.$

1) Ecrire une fonction définissant g.
2) Quelle est la valeur de $g(-2)+g(2)$? Donner une instruction permettant l'affichage de cette valeur.
3) Quelle est la valeur de $gof(-3)$? Donner une instruction permettant l'affichage de cette valeur.
4) Dessiner le graphe de gof .
5) Ecrire une fonction définissant gof puis faire tracer son graphe sur [-2,2].

Ex6 : Que fait le script scilab suivant ? (sachant que modulo(x,y) donne le reste de la division euclidienne de x par y)

```
n=input('choisir un entier')
for k=1:n
        if  modulo(n,k)==0  then disp(k,'il est divisible par')
        end
end
```

Ex7 : Donner un script scilab qui donne le nombre de nombre premier avant n (où n est un entier donné par l'utilisateur).
(vous pouvez utiliser le crible d'Ératosthène)**.**

Défi : Donner un script scilab qui donne le jour de la semaine en sortie, et qui prend en entrée une date (jours, mois et années).