TP 8 – Fonctions

Motivations

Les fonctions jouent un rôle central en programmation. Voyons ici un exemple d'utilisation.

On souhaite calculer $3x^2 + 5y$ pour $(x, y) = (1, 1)$ puis pour $(x, y) = (2, 3)$ et éventuellement d'autres valeurs.

```
// Méthode 1 - dans la console
--> x = 1
--> y = 1
--> 3 * x^2 + 5 * y
8.
--> x = 2
--> y = 3
--> 3 * x^2 + 5 * y
27.
```

```
// Méthode 2 - avec input
x = input("x = ")
y = input("y = ")
disp(3 * x^2 + 5 * y)

// F5
--> exec('...')
x = 1
y = 1
8.

// F5 une 2e fois
--> exec('...')
x = 2
y = 3
27.
```

Une **fonction** permet d'être plus efficace sur ce problème.

```
// Méthode 3 - avec une fonction
function z = f(x,y)
    z = 3 * x^2 + 5 * y
endfunction

// F5 - une seule fois en tout
--> exec('...')
--> f(1,1)
ans =
    8.
--> f(2,3)
ans =
    27.
```

I Définir et utiliser une fonction

I.1 Syntaxe scilab

```
function variable_sortie = nom_fonction(x1, x2, ..., xn)
    instructions
    permettant de définir
    la variable_sortie
endfunction
```

- `nom_fonction` est le nom de la fonction qui sera utilisé ensuite. Pas d'espace, pas de caractère spécial. Préférez les noms de fonctions explicites.
- `x1, x2, ..., xn` sont les paramètres d'entrée. Ce sont les variables de la fonctions. Ne pas mettre de valeur ici, on garde des noms génériques.
- `variable_sortie` est le nom de la variable de sortie, c'est-à-dire du résultat.
- **Pas de disp, pas de input !**

I.2 Faire appel à la fonction

1. On définit notre fonction dans l'éditeur.

```
function z = f(x,y)
    z = 3 * x^2 + 5 * y
endfunction
```

2. On informe scilab de la création de cette fonction en **exécutant** (F5)

```
--> exec('...')
```

   Rien ne s'affiche dans la console, c'est normal. La fonction a juste été répertoriée.

3. Pour l'utiliser, il faut simplement donner le nom de la fonction suivi, entre parenthèses, des valeurs des variables d'entrée.

```
--> f(2,3)
ans =
    27.
```

   Le résultat est automatique affiché, sans avoir à écrire disp dans la fonction.

II Exemples

1 – Définir en scilab la fonction $g : x \mapsto \begin{cases} 2x & \text{si } x \leqslant 0 \\ \ln(x) & \text{si } x > 0 \end{cases}$.

Calculer alors $g(5)$.

2 – Soit $(u_n)_{n\in\mathbb{N}}$ la suite définie par $\begin{cases} u_0 = 0 \\ \forall n \in \mathbb{N}, \quad u_{n+1} = \sqrt{u_n + 2n} \end{cases}$.
Créer une fonction `Suite_u(n)` prenant en entrée un entier naturel n et renvoyant la valeur de u_n.

On considère maintenant la suite $(v_n)_{n\in\mathbb{N}}$ définie par $v_n = u_n + \dfrac{1}{n+1}$. Créer une fonction `Suite_v(n)` prenant en entrée un entier naturel n et renvoyant la valeur de u_n.

III Variables locales

Les **variables locales** d'une fonction sont :

- les variables d'entrée d'une fonction (les paramètres d'entrée) ;
- la variable de sortie ;
- les variables dont la valeur est définie dans les instructions de la fonction.

Cela signifie que leur portée est réduite à la fonction.

```
function z =  fonction1(x,y)
    z = 4*x + 5*y
endfunction
```

Variables locales de cette fonction :

```
--> exec(...)
--> fonction1(1,4)
 ans  =
   24.

--> x
Variable non définie : x

--> y
Variable non définie : y

--> z
Variable non définie : z

--> x = 2

--> fonction1(1,4)
 ans  =
   24.

--> x
 x  =
   2.
```

Les lettres x, y, z peuvent donc être utilisées ailleurs, sans aucune incidence sur la fonction1. Mais il ne faut pas non plus considérer que la variable z est connue globalement.
Seul le nom de la fonction est mémorisé.

Exercices d'application

Exercice 1

Définir une fonction appelée f, ayant pour paramètre d'entrée x et renvoyant la valeur $3x^2+5x-2$.

Tester la fonction en l'appelant plusieurs fois dans la console : $f(0)=-2$, $f(1)=6$, $f(2)=20$.

Exercice 2

1. Résoudre à la main, dans $\mathbb{R}$, les équations : $x^2+x+1=0$; $4x^2-4x+1=0$ et $2x^2-x-3=0$.

2. Écrire une fonction `degre2R(a,b,c)` qui, étant donnés trois réels a, b, c, renvoie les solutions réelles de $ax^2+bx+c=0$ sous forme de liste (éventuellement vide). On supposera que $a \neq 0$.

Exercice 3 **Factorielle**

1. Écrire une fonction `factorielle` ayant pour paramètre un entier n et renvoyant la valeur de $n!$.

2. Tester la fonction : $0! = 1$, $3! = 6$, $5! = 120$.

3. En déduire une fonction qui, pour $n \in \mathbb{N}$ donné, renvoie $\displaystyle\sum_{k=0}^{n} \frac{1}{k!}$

   Vérifier que, pour n grand, $\displaystyle\sum_{k=0}^{n} \frac{1}{k!} \approx \mathrm{e}$.

Exercice 4 Définir une fonction appelée g, ayant pour paramètre d'entrée x, renvoyant

$$g(x) = \begin{cases} 1 & \text{si } x = 0 \\ \dfrac{\ln(x)}{2x} & \text{si } x > 0 \\ \mathrm{e}^{-1/x^2} & \text{si } x < 0 \end{cases}$$

Exercice 5 **Comportement d'une suite**

On considère la suite (u_n) définie par $\begin{cases} u_0 = 100 \\ \forall n \in \mathbb{N},\ u_{n+1} = \sqrt{6 + \sqrt{3u_n}} \end{cases}$

1. Définir la fonction $h\colon x \mapsto \sqrt{6 + \sqrt{3x}}$.

2. Écrire une fonction `SuiteU`, ayant pour paramètre un entier naturel n et qui renvoie u_n. On utilisera la fonction h.

3. En utilisant cette fonction, conjecturer la limite ℓ de la suite (u_n).

4. Écrire une fonction d'entête `function n = seuil(s)` qui, étant donné un réel positif s, renvoie le premier rang n pour lequel $|u_n - \ell| \leqslant s$.

```
function n = seuil(s)
```

Tester avec $s = 10^{-3}$, $s = 10^{-5}$.