

Cours de J (Leçon 5) par Michel J.Dumontier



Définitions de fonctions

Rappel : ce qui suit est le résultat de l'exécution d'un script J.

Les instructions sont en gras (sauf les NB.), les variables, verbes et expressions sont aussi en gras dans les commentaires, le tout, en police « Courier New » car pour la police ISIJ, il est très difficile de distinguer le gras du non gras.

Par souci d'homogénéité les résultats sont aussi dans la même police sauf lorsqu'il est nécessaire de les mettre en police ISIJ. (Les accents des commentaires ont été rajoutés).

NB. RAPPELS:

NB. Les fonctions sont ambivalentes

NB. Pas d'ordre hiérarchique des fonctions

NB. Les verbes sont des actions (fonctions)

NB. Les adverbes et les conjonctions modifient les verbes

NB. Les vecteurs, matrices, tableaux sont des entités

NB. On peut assigner des noms à des fonctions:

NB. somme =. +/

NB. compte=. #

NB. moyenne =. somme % compte

NB. _ représente l'infini, __ représente moins l'infini (-_)

NB. DEFINITION TACITE

NB. LA PROGRAMMATION TACITE ou FONCTIONNELLE

NB. NE NÉCESSITE PAS D'ARGUMENTS EXPLICITES.

NB. Un train de verbes est composé de 2 ou 3 verbes ou plus

NB. Une FOURCHE monadique (trident)

NB. (f g h) y est équivalent à

NB. (fy) g (h y)

moy=. +/ % #

]a=.i.5

0 1 2 3 4

moy a

2

NB. Utilisation de RACINE diadique

NB. moyenne géométrique: # %: */

NB. #ème racine du produit

```

mg=.# %: */
]b=.>:a
1 2 3 4 5
mg b
2.60517
*/b
120
(mg b)^5
120

```

NB. Une FOURCHE diadique
 NB. x (f g h) y est équivalent à
 NB. (x f y) g (x h y)

```

5 (+ * -) 4 NB. (5+4)*(5-4) => (25-16)
9

```

NB. Un train de 2 verbes est un CROCHET
 NB. (g h) y est équivalent à
 NB. y (g h) y

```

(*>:) 1 2 3 NB. (1 2 3) * (2 3 4)
2 6 12

```

NB. x (g h) y est équivalent à
 NB. x g (h y)

```

3 4 (*>:) 1 2 NB. (3 4) * (2 3)
6 12

```

NB. CAPTURE de verbes

```

+/ + * - 2 3 4
_3

```

```

+/ (+ * -) 2 3 4
_29

```

```

c=. [: +/ + * -
c 2 3 4
_29

```

NB. [: capte une branche d'une fourche:
 NB. ici, +/ est appliqué au résultat
 NB. de la fourche + * -

NB. CAS :

NB. l'argument gauche spécifie
 NB. le cas monadique et le droit

NB. donne le cas diadique.

```
log=. 10&^. : ^.
```

```
a=. 1x1, 10 100
```

```
^. a
```

```
1 2.30259 4.60517
```

```
log a
```

```
0.4342945 1 2
```

```
1x1 log a
```

```
1 2.30259 4.60517
```

NB. Rappel de la conjonction AVEC &

NB. Voir Leçon 4

```
carre=.^&2
```

```
carre 3 7
```

```
9 49
```

```
racinecarree=. ^& 0.5
```

```
racinecarree carre 3 7
```

```
3 7
```

NB. Conjonction AUDESSUS @

```
troisfois2puissance=. (3&*)@(2&^)
```

```
troisfois2puissance 2 3
```

```
12 24
```

```
amoinsbpuissance3=.^&3@-
```

NB. Les adverbes et les conjonctions sont

NB. exécutés avant les verbes

```
3 4 5 6 amoinsbpuissance3 6 5 4 3
```

```
_27 _1 1 27
```

```
(3 4 5 6-6 5 4 3)^3
```

```
_27 _1 1 27
```

NB. La conjonction PUISSANCE agit sur les verbes comme

NB. la puissance agit sur les nombres

NB. $f \wedge 2$ est équivalent à $f f$

```
b=.>:i.5
```

```
b^2
```

```
1 4 9 16 25
```

```
#: b^2
```

```
1 2 3 4 5
```

```
#: b^4
```

```
1 4 9 16 25
```

```

%: ^:2 b^4
1 2 3 4 5
%: %: %: b^8
1 2 3 4 5
%: ^:3 b^8
1 2 3 4 5

```

NB. L'infini utilisé en puissance pour les fonctions:
 NB. Exemple: algorithme convergent,
 NB. Solution de $y = \cos y$

```

cos=.2&o. NB. cosinus est la 2ième fonction circulaire
d=.1
]y=. cos ^: _ d
0.7390851
y=cos y
1

```

NB. Definition de la fonction inverse par
 NB. la conjonction puissance.
 NB. avec l'argument _1

```

%: ^: _1 b
1 4 9 16 25
e=.0.6
%: ^: _ e
1

```

NB. Calcul de la constante d'Euler $e = 2.718281828\dots$
 NB. si nous calculons $(1+1\%_)^_$ nous obtiendrons 1 à la
 NB. puissance l'infini ce qui donne 1.

```

(1+1%_ )^_
1

```

NB. il faut lier les infinis: écrivons donc la fonction e
 NB. et on l'appliquera à un argument x que l'on fera tendre
 NB. vers l'infini.

```

e=.>:&% ^ ]
e 10
2.59374
e 100
2.70481
e 1000
2.71692
e 10000
2.71815
11.7 ": e 100000000
2.7182818

```

NB. DEFINITION EXPLICITE D'UNE FONCTION

NB. DEFINITION D'UNE FONCTION MONADIQUE

```
ee=. 3 : 0  
(>:%y.)^y.  
:  
1 NB. ici on mettrait la définition diadique  
)  
  e      NB. forme tacite  
>:&% ^ ]  
  ee     NB. forme explicite  
3 : 0  
(>:%y.)^y.  
:  
1 NB. ici on mettrait la définition diadique  
)
```

```
  e 10  NB. forme tacite  
2.59374  
  ee 10 NB. forme explicite  
2.59374
```

```
  e 10000  
2.71815  
  ee 10000  
2.71815
```

NB. On peut définir directement
NB. une fonction `f=. 3 : 0`)
NB. une conjonction `conj=. 2 : 0`
NB. un adverbe `adv=. 1 : 0`

NB. On peut définir aussi un "script"
NB. `s=. 0 : 0` et l'utiliser pour définir
NB. un verbe, une conjonction ou un adverbe
NB. par `(3 : s)`, `(2 : s)`, `(1 : s)`
NB. aussi un substantif par `(0 : s)`

NB. exemple:

```
  carre=.*:  
  
  s=. 0 : 0  
carre y.  
:  
(carre x.)+carre y.  
)  
  
  f=. (3 : s)  
  f 3 4 5
```

9 16 25
3 f 4
25

NB. ASSIGATION GLOBALE ET LOCALE
NB. DANS LES FONCTIONS

```
a=.i.3 4

f=. 3 : 0

1
:
a=.2 NB. assignation locale
bb=:3 NB. assignation globale
a+bb
)
2 f 3

5
a
0 1 2 3
4 5 6 7
8 9 10 11
bb
3
```

NB. FIXATION DE FONCTIONS

NB. ajout de 1 à la fonction de tirage aléatoire (à cause
NB. de l'origine 0)

```
alea=.>:@?
loto=.alea
monloto=.loto f.
4 loto 4

4 2 1 3
4 monloto 4

3 1 2 4
alea=.? NB. redéfinition de la fonction alea normale
4 loto 4

2 1 3 0
4 monloto 4

1 4 2 3
6 monloto 49 NB. pour avoir un loto correct

25 24 33 1 23 47
```

NB. Pour mettre une valeur dans un tableau

```
MONTRE=.]
PREND =.{.
LAISSE=}.
MONTRE alpha=.26 PREND 65 LAISSE a.
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

MONTRE x=.4 5\$alpha

ABCDE
FGHIJ
KLMNO
PQRST

in=. 3 : 0

1
:
(_1{.x.})+(1{.x.})*_1{.\$y.
)

NB. 'a' (1 2 in x)}x fonctionnait, bien que peu
NB. orthodoxe !
NB. il a fallu la compliquer pour obtenir ce qui était
NB. désiré

(\$x)\$ 'a' (1 2 in x) } ,x

ABCDE
FGaIJ
KLMNO
PQRST

(à suivre...)