

Fonction et Pseudo code (partie 3)

Sébastien Jean

IUT de Valence
Département Informatique

v1.0, 21 septembre 2026

Rappel : fonction factorielle

```
FONCTION factorielle (n : entier) : entier
```

```
    VARIABLE resultat : entier
```

```
    VARIABLE compteur : entier
```

```
    resultat ← 1
```

```
    POUR compteur DE 1 A n AVEC UN PAS DE 1
```

```
        resultat ← resultat * compteur
```

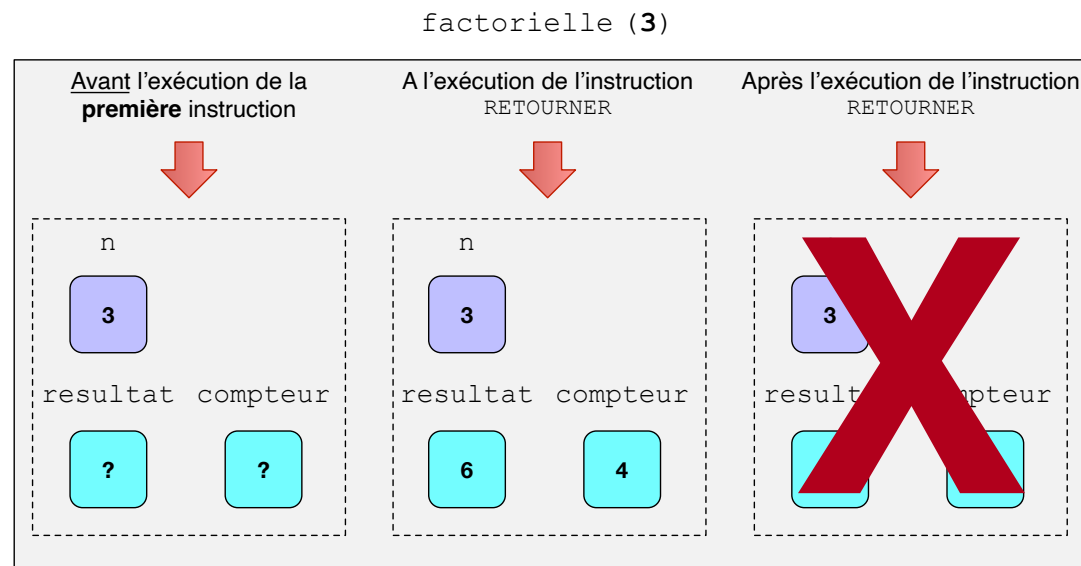
```
    FIN POUR
```

```
    RETOURNER resultat
```

```
FIN FONCTION
```

Pseudo code : contexte d'appel

- L'**appel de la fonction** factorielle avec la valeur 3 en paramètre crée un **contexte** constitué :
 - des **variables déclarées** dans la fonction
 - d'une **copie de chaque paramètre déclaré** par la fonction



- Il évolue au fil des instructions et est **détruit** à la fin de l'appel

Exemple d'une fonction appelant factorielle

```
FONCTION somme_factorielles (i : entier) : entier
```

```
    VARIABLE r : entier
```

```
    VARIABLE c : entier
```

```
    resultat  $\leftarrow$  0
```

```
    POUR c DE 0 A i AVEC UN PAS DE 1
```

```
        r  $\leftarrow$  r + factorielle(c)
```

```
    FIN POUR
```

```
    RETOURNER r
```

```
FIN FONCTION
```

Appel de somme_factorielles(3) ?



Pseudo code : réaffectation de paramètres

- On s'autorise à **réaffecter les paramètres** dans une fonction
-

```
FONCTION quelconque (n : entier) : entier
```

```
    TANT QUE NON est_pair(n)
```

```
        n ← factorielle(n)
```

```
    FIN TANT QUE
```

```
    RETOURNER n
```

```
FIN FONCTION
```

- Appel de quelconque(3)



Retour sur la factorielle

- Réécrire factorielle sans la variable compteur



Retour sur la fonction factorielle

FONCTION factorielle (n : entier) : entier

VARIABLE resultat : entier

resultat $\leftarrow$ 1

TANT QUE n > 1

 resultat $\leftarrow$ resultat * n

 n $\leftarrow$ n - 1

FIN POUR

RETOURNER resultat

FIN FONCTION

Pseudo code : passage de paramètres

- Les **paramètres** sont passés par **copie**
- L'affectation dans une fonction appelée n'a **aucun effet dans la fonction appelante**

```
FONCTION quelconque2 () : entier
```

```
    VARIABLE s : entier
```

```
    s ← 3
```

```
    RETOURNER quelconque(s) + s
```

```
FIN FONCTION
```

- Appel de quelconque2()



Pseudo code : variables / constantes locales

- Les variables et constantes déclarées dans une fonction sont **locales**
 - On ne peut y faire référence que dans la fonction, invisibles ailleurs

```
FONCTION f1 (...) : ...  
  
    VARIABLE c : entier  
  
    c <- 1  
  
FIN FONCTION
```

```
FONCTION f2 (...) : ...  
  
    VARIABLE x : réel  
  
    x <- 1  
    x <- c    interdit !  
  
FIN FONCTION
```

Pseudo code : variables / constantes globales

- Les variables et constantes déclarées **avant** les fonctions sont **globales**
- On peut y faire référence dans toutes les fonctions

```
VARIABLE c : réel  
CONSTANTE PI : réel (3.14)
```

```
FONCTION f (...) : ...  
  
    VARIABLE x : réel  
  
    x <- 1  
    x <- c                autorisé !  
    RETOURNER x * PI      autorisé !  
  
FIN FONCTION
```

- N.B. On s'interdit de définir une variable/constante locale de même nom qu'une variable/constante globale

Fin !

