

# Fonction et procédures, échanges de données

## I Fonctions

Une fonction peut être vue comme une “boîte noire”

- La fonction appelante ne connaît pas le traitement effectué par la fonction appelée.
- La fonction appelée communique uniquement (avec la fonction appelante) par l'intermédiaire de ses entrées/sorties (souvent notée E/S)

Fonction 1 qui donne des paramètres à une fonction 2 et cette dernière renvoie un retour.

F° Somme : arg : a, b et sortie a+b

Une fonction est une section de code produisant un résultat unique (Sortie), éventuellement à partir d'une ou plusieurs données fournies en entrée (paramètres).

Pour définir une fonction, on doit :

- Établir une sortie unique : ce que la fonction retourne (*return* en basic), C'est obligatoire pour toutes les fonctions
- Établir zéro, une ou plusieurs entrées : ce sont les paramètres (ou arguments)
- Établir le traitement effectué par la fonction (sa mécanique interne)

Syntaxe :

```
[Private|Public] Function Nom_de_la_fonction(listeParamètres) As TypeRetour  
Instruction
```

Return Expression

End Function

*listeParamètres* vaut :

```
[[ByVal|ByRef]nomParamètres As typeParamètres]*
```

ou le pipe “|” représente une alternative et l'astérisque “\*” une ou plusieurs fois

Par défaut, il s'agit de *ByVal* et la liste des paramètres est séparée par une virgule

```
Public Function Fontion(ByVal N1 As Integer, ByVal N2 As Integer) As Boolean
```

```
Dim ret As Boolean
```

```
...
```

```
Return ret
```

```
End Function
```

Lorsque l'on veut utiliser une fonction, on appelle par son nom, en lui passant tous les paramètres.

- On peut déclarer une fonction sans paramètres. on place alors des parenthèses vides après le nom de la fonction.
- Une fonction peut appeler d'autres fonctions.
- Une fonction répond à une tâche, un besoin précis.

**En algorithmique :**

Appeller une fonction : rectangle avec double barre latérale.

Rqs :

Les variables déclarées dans la fonction appelante (ici le paramètres `n_teste` et résultat) n'ont pas le même nom que les variables utilisées dans le code de `est_prem` ("boite noire")

C'est le compilateur qui va établir des correspondance entre les variables d'entrée et les paramètres.

Il existe des fonctions qui ne retournent rien ... Ces fonctions particulières s'appellent des procédures.

En VB.net : ce sont des Sub (pour "subroutines")

```
[Private|Public] Sub NomProc(listeParamètres)
```

```
Instructions End sub
```

Les procédures événementielles sont des procédures particulières qui sont liée à un événement (c-a-d interruptions)

ByVal : copie de la valeur qui est transmise

ByRef : c'est la référence (adresse mémoire) qui est transmise