

La syntaxe de VISUAL BASIC (partie 3)

Une procédure est un ensemble d'instructions, de lignes de code, un groupement d'instructions bien définies effectuant une tâche précise.

Déclaration d'une procédure :

SYNTAXE

```
[Private | Public] [Static] Sub nomproc [ (listearg) ]
    [ Déclaration ]
    .....
End Sub
```

*Public: Indique que la procédure Sub est accessible à toutes les autres procédures de tous les modules.

*Private: Indique que la procédure Sub n'est accessible à d'autres procédures qu'au sein du module dans lequel elle est déclarée.

*Static: Indique que les variables locales de la procédure Sub conservent leur valeur entre deux appels de procédure.

*nomproc: Nom de la procédure Sub ; il est soumis aux conventions standards d'appellation des variables.

*listearg: Liste de variables représentant des arguments qui sont passés à la procédure

La syntaxe de l'argument listearg est la suivante :

```
[ByVal | ByRef] nomvar [ () ] [As type]
```

*ByVal: Indique que l'argument est passé par valeur (Entrée)

*ByRef: Indique que l'argument est passé par référence (Sortie ou Entrée/Sortie)

*nomvar: Nom de la variable qui représente l'argument

*Type: Type de données de l'argument passé à la procédure.

Une procédure **Sub** peut être appelée de deux manières :

```
CALL nomproc (argument1, argument2, ...)
ou
nomproc argument1, argument2, ...
```

Remarques :

- + Les procédures Sub sont par défaut Public
- + Tout le code exécutable doit se trouver dans des procédures
- + On ne peut pas définir des procédures dans des procédures
- + Le mot clé Exit Sub entraîne la sortie immédiate d'une procédure Sub
- + Une procédure Sub ne peut pas être utilisée dans une expression.hile...Wend

Les fonctions en VB

Déclarent le nom, les [arguments](#) et le code formant le corps d'une [procédure](#) **Function**.

Syntaxe

[Public | Private | Friend] [Static] Function *name* [(*arglist*)] [As *type*]

[*statements*]

[*name* = *expression*] ' renvoie le résultat

EndFunction

Un tableau de variables permet de stocker **plusieurs variables de même type** sous un même nom de variable, chaque élément étant repéré par un **index** ou indice. C'est une suite finie d'éléments.

Exercice N°10

En utilisant la procédure Permute avec deux paramètres (vue en SI4) que vous définirez, écrire un programme VB console qui affiche trois nombres saisis au clavier de manière ordonnée (croissant).



```
file:///C:/Users/erwan.ESTRANSUP/Desktop/TP02 Syntaxe VB/
Nombre N°1 : 25
Nombre N°2 : 40
Nombre N°3 : 17
25 - 40 - 17
Tri en ordre croissant par permutations....
17 - 25 - 40
```

```
2 = function permut(&$A, &$B)
3 {
4     $tmp=$A;
5     $A=$B;
6     $B=$tmp;
7 }
```

Exercice N°11

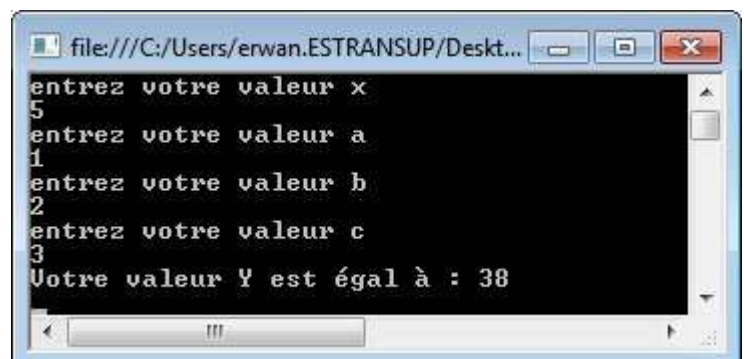
Ecrire la fonction permettant de calculer Y sachant que :

$$Y = A X^2 + B X + C$$

A, B, C et X sont donnés.

Ecrire aussi le programme VB console permettant de tester cette fonction.

Prévoir un tableau de valeur (Version 2).



```
file:///C:/Users/erwan.ESTRANSUP/Desktop/...
entrez votre valeur x
5
entrez votre valeur a
1
entrez votre valeur b
2
entrez votre valeur c
3
Votre valeur Y est égal à : 38
```

Exercice N°12

Créer une nouvelle version du programme CaiseEnregistreuse en utilisant le maximum de fonctions et de procédures.