

## Petit récapitulatif

**Entrées Sorties** ces commandes sont dans le menu **Prgm** Rubrique E/S  
**Prompt Variable** : sert à demander à l'utilisateur une valeur que la calculatrice stockera dans la variable.

**Exemple :**  
Prompt N la calculatrice affichera : N= ? et quand l'utilisateur aura proposé une valeur et appuyé sur entrer, elle stockera la valeur dans N.

**Disp "phrase 1", " phrase 2", "etc"** : affichera la phrase 1 sur une ligne, puis la phrase deux sur la ligne suivante. Disp peut aussi servir à afficher les valeurs contenues dans des variables.

**Exemple :**  
Disp "N VAUT ",N la calculatrice affichera N VAUT sur une ligne et la valeur contenue dans N à la suivante.

**ClrHome / EffEcran** ces commandes effacent l'écran principal

### Boucles

Il y en a deux type, la boucle conditionnelle (While), qui tourne tant que certaines conditions sont réalisées, il y a aussi la boucle "semi-rigide" (For), qui tournera un nombre fixé de fois.

Syntaxe :

<b>While (condition)</b>	<b>For (variable, valeur du début, valeur de fin)</b>
<i>petite</i>	<i>petite</i>
<i>routine</i>	<i>routine</i>
<b>End</b>	<b>End</b>

### **Exemple**

For (I,1,15) Quand la boucle commence I vaut 1  
Disp I<sup>2</sup> A chaque fois que la boucle recommence I augmente de 1  
Pause Jusqu'au moment ou I vaut 15 et là la boucle s'effectue une dernière fois  
End  
Ce programme affiche les carrés de 15 premiers entiers

### Condition

Des fois l'on veut que la calculatrice fasse des actions, mais pas tout le temps, juste quand certaines conditions sont réunies. Une condition en informatique c'est une égalité / inégalité, elle sera vraie ou fausse.

Par exemple A=B est une condition, les deux variables A et B ont chacune une valeur précise au moment du test, et dans ce cadre là l'égalité est soit vraie soit fausse.

Des fois on combinera plusieurs condition avec les opérateurs logiques And /Et, Or/Ou, etc car on peut vouloir que deux conditions soient réalisées pour qu'une action soit effectuée (ou encore qu'au moins une sur les deux le soit).

### **If (condition)**

**Then**  
*|Petite routine à exécuter* ou encore  
*|quand la condition est remplie*  
**End**

### **If (condition)**

**Then**  
*|Petite routine à exécuter*  
*|quand la condition est remplie*  
**Else**  
*|Petite routine à exécuter*  
*|quand la condition n'est pas remplie*  
**End**

On utilisera la fonction **If** lorsque l'on veut que la calculatrice exécute une séquence d'action seulement si certaines conditions sont remplies

### **Exemple**

Disp "BORNES DE L INTERVALLE" la calculatrice affiche un message clarifiant  
Prompt A,B les questions posées par le prompt

**If A<B**  
**Then**  
Disp "L INTERVALLE A", " POUR LONGUEUR",B-A  
**Else**  
Disp "ON DOIT CHOISIR", "A ET B TELS QUE", "A<B"  
**End**

### Divers :

**Rand / NbrAléat** : donne un nombre aléatoire de l'intervalle [0,1[

**iPart( / PartEnt(** : donne l'entier immédiatement inférieur au nombre

**Exemple** PartEnt(5,6) donne 5 et PartEnt(-3,4) donnera -4

**int( / ent(** : donne la partie entière du nombre

**Exemple** ent(5,6) donne 5 et ent(-3,4) donnera -3

### Listes

**{0,5,8,9} → L<sub>2</sub>** stocke la liste {0,5,8,9} dans L<sub>2</sub>

**L<sub>2</sub> / 4** divise tous les éléments de la liste L<sub>2</sub> par 4

**L<sub>2</sub> +11** ajoute 11 à tous les éléments de la liste L<sub>2</sub>

**L<sub>2</sub> (3)** donne le troisième élément de c'est-à-dire 8

**7 → L<sub>2</sub> (2)** stockera 7 dans le deuxième élément de la liste L<sub>2</sub>

**L<sub>2</sub> (D)-3 → L<sub>2</sub> (D)** diminuera de 3 l'élément numéro D de la liste si D est 1, 2, 3 ou 4 si D correspond à un élément non présent la calculatrice affichera un message d'erreur