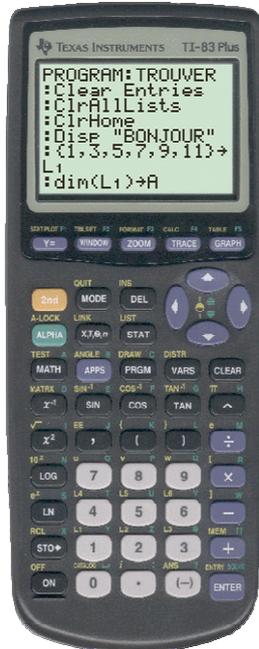


Initiation à la programmation sur calculatrice TI

Commençons par créer un nouveau programme pour tester les différentes fonctions du basic TI.

Touche Prgm puis New et choisir un nom.



Fonctions d'affichage:

Disp et Output affichent du texte à mettre entre guillemets ou des valeurs stockées ou non et même des résultats d'opération.

Exemple: Disp "Bonjour"
 Disp 3 + 4

Exemple: Output 5,5, "Bonjour"
affiche à la ligne 5, cinquième colonne, Bonjour

Variables et stockage:

Pour stocker des données utilisables ultérieurement il suffit de choisir une lettre et de cliquer sur le bouton STO→:

Exemple: 8 STO→ N
 affecte la valeur 8 à N
 4 sto→ M
 Disp M+N va afficher 12

Les listes: plusieurs données peuvent être stockées dans une seule liste. Des opérations sur les listes sont possibles:

Exemple: {1, 3, 5, 7, 9, 11}→L1

Disp L1^2

Disp 2*L1

Boucles conditionnelles:

Il y en a plusieurs, pour cette introduction, nous n'en utiliseront qu'une: IF THEN ELSE END

Ecriture: IF condition1
 THEN
 Action1
 ELSE (facultatif)
 Action2
 END

Exemple: 3→M
 4→N
 IF N=M
 THEN
 Disp "Bravo!"
 ELSE
 Disp "Perdu!!"
 END

Boucles de comptage:

Servent pour des actions répétitives indicées ou non.

Ecriture : FOR(J,1,8,2) où J est le compteur, qui compte de 1 à 8 par pas de 2.

Exemple simple : FOR(J,1,8,1)
 DISP "Compteur=",J
 END

Deuxième exemple: FOR(K,1,8,1)
 K^2→A
 Disp A
 END

Amélioration de l'affichage:

```
FOR(K,1,8,1)
K^2→A
OUTPUT(K,K,A)
END
```

Amusement: FOR(K,1,8,1)
 K^2→A
 OUTPUT(K,K,A)
 END
 FOR(K,7,1,-1)

```
K^2→B
OUTPUT(K,16-K,B)
END
```

```
GOTO D
END
DISP " Trop petit" (seul
Possibilité restante)
GOTO D
LBL F
```

Remplissage de listes:

```
FOR(J,1,8,1)
J^2→L1(J)
Disp L1
END
```

Quand la liste est trop longue elle ne s'affiche pas entièrement. Les flèches permettent de se déplacer, mais quand il s'agit de l'affichage d'un programme les flèches sont inopérantes.

Exercice: utiliser la fonction OUTPUT dans la boucle FOR pour afficher tous les éléments de la liste.

Exemple:

```
FOR(J,1,8,1)
9-J→L1(J)
OUTPUT(1,2*J,L1(J))
END
```

Premier programme:

La calculette choisi un nombre aléatoire entre 0 et 10. L'utilisateur doit trouver ce nombre. Une information est donnée comparant le nombre entré avec celui choisi par la calculette.

Amélioration compter en combien de coups le bon nombre a été trouvé.

Listing:

```
CLRHOME
0→C (compteur)
INT(RAND*10)→A
LBL D (étiquette pour un saut)
INPUT "Nombre ≤ 10",B
C+1→ C (incréméntation du
Compteur)
IF A = B
THEN
DISP "Bravo! Gagné en"
DISP C
DISP "coups"
GOTO F (étiquette de fin du
prog)
ELSE
IF A ≥ B
THEN
DISP "Trop grand"
```

Exercez-vous:

Conversion binaire – décimale.

Le programme demande l'entrée d'un nombre binaire sur 8 bits et calcule le nombre décimal correspondant. Il affiche à la fin le nombre binaire et le nombre décimal.

ATTENTION! Le bit 0 est le bit de poids faible, c'est le huitième de la liste!

Fonctions utilisées: DISP, FOR, INPUT, LISTES, OUTPUT

```
Programme BINDEC
Clear Entries
ClrHome
Disp "BIN->DEC 8BITS"
{0,0,0,0,0,0,0,0}->L1
0 -> Z
For (J,8,1,-1)
Disp "BIT ",8-J
Input X
X -> L1(9-J)
L1(9-J)*2^(8-J) + Z-> Z
End
CLRHOME
OUTPUT(1,1,"Binaire = ")
For (k,1,8,1)
Output(2,k,L1(9-k))
End
OUTPUT(4,1,"DECIMAL = ")
OUTPUT(4,10,Z)
```

Entraînement:

Imaginez un programme de conversion décimale - binaire en utilisant la méthode des divisions successives par 2.