Quelques opérations de base de programmation sur TI

(chapitre 16 du manuel de la TI82 Stats-*fr*)
http://education.ti.com/educationportal/downloadcenter/SoftwareDetail.do?website=FRANCE&tabId=2&appId=6622

Commencer un programme:
(prgm) NOUV
Entrer le nom du programme
appuyer sur (entrer)

Pour revenir sur l'édition d'un programme:
(prgm) EDIT
Sélectionner le programme
appuyer sur (entrer)

Sortir de l'écriture d'un programme:
(2nde)[quitter]

Les entrées sorties: (prgm) E/S

Entrer une valeur au clavier : deux possibilités : `Prompt` ou `Input`

`Prompt A` : écrit à l'écran " `A=?` " et demande la valeur A.

`Input "valeur A",A` : écrit "`valeur A`" et demande la valeur de A.

Afficher un résultat à l'écran : `Disp`

Afficher un texte: `Disp " le texte "`

Pour afficher plusieurs textes simultanément, on peut les séparer par une virgule. [,]

Contrôle des données: (prgm) CTL

1) Pour réaliser un test, on peut utiliser la fonction `If... Then... Else.`

```
If condition
Then
        commande à réaliser si la condition est vérifiée
        commande à réaliser si la condition est vérifiée
Else
        commande à réaliser si la condition n'est pas vérifiée
        commande à réaliser si la condition n'est pas vérifiée
End
```

2) Une autre façon de faire un test: `While` (tant que)

```
While condition
        commande à réaliser si la condition est vérifiée
        commande à réaliser si la condition est vérifiée
End
```

Pour trouver les signes <, >, ⩽, ⩾, utiliser la combinaison: (2nde)[tests]

Pour arrêter un programme: `Stop`

Faire une pause dans les calculs ou l'affichage: (prgm) CTL `Pause`

Enregistrer une valeur dans une mémoire: touche (sto→)

3) Pour créer une boucle de répétition

```
For (nom de la variable, départ, arrivée, [incrément])
        commande à réaliser tant qu'arrivée n'est pas atteint
        commande à réaliser tant qu'arrivée n'est pas atteint
End
```

Remarque: Si la variable est A, elle est incrémentée ou décrémentée jusqu'à son dépassement.
Illustration par l'exemple:
Si l'incrément est 1 et que l'on veut que A varie entre 1 et 8, la boucle va s'effectuer pour les valeurs 1, 2, 3, 4, 5, 6, 7 et 8.
Toutefois à la sortie de la boucle, la valeur de A sera 9.
Si l'incrément est 3 et que l'on veut que A varie entre 1 et 8, la boucle va s'effectuer pour les valeurs 1, 4, et 7.
Toutefois à la sortie de la boucle, la valeur de A sera 10.
Il est nécessaire d'y faire attention lorsque l'on a besoin de réutiliser les la dernière valeur de la variable dans la suite du programme (ce qui est souvent le cas pour les boucles imbriquées).

4) Les balises Lbl Goto.

Il est également possible de placer des balises dans le programme.

`Lbl` (étiquette) et `Goto` (aller à) permettent de contrôler les branchements.

`Lbl` désigne l'étiquette d'une commande. L'étiquette se compose d'un ou deux caractères (**A** à **Z**, **0** à **99**, ou θ): permet de repérer un début de procédure.

rédaction du programme: `Lbl` étiquette

`Goto` provoque le branchement du programme vers l'étiquette au moment où l'instruction `Goto` est exécutée.

rédaction du programme: `Goto` étiquette