

Mode d'emploi de la Calculatrice programmable de **Guillaume Nollet** (élève de seconde)

Un grand merci pour le travail accompli

On va aller étape par étape. On va pas tout expliquer en même temps. Ne vous précipitez pas. Ne vous bloquez pas si vous ne comprenez pas : relisez attentivement et intelligemment, demandez à quelqu'un d'autre, appelez-moi au téléphone (sauf qu'en fait j'en ai pas). Merci.

L'ensemble des commandes qui vont suivre ont été vérifiées sur la TI-Nspire. Elles marchent.

Les bases

Pourquoi on doit s'embêter à apprendre la programmation ?

Parce que des vieux monsieurs qui viennent du Ministère de l'Éducation Nationale en ont décidé ainsi. Et accessoirement parce que tous les objets électroniques, c'est de la programmation.

Comment réussir en programmation ?

Pour bien réussir en programmation, l'important c'est de noter toutes les commandes au fur et à mesure, sans oublier de noter leur signification. Et puis faut les apprendre, y a pas 36 solutions.

Les commandes de la calculatrice (et de tout logiciel de programmation) ne s'inventent pas. C'est comme ça, point. Alors ne venez pas vous plaindre si c'est mal recopié.

Les commandes de base de la calculatrice :

Formules simples	
Define [programme] ()= Prgm	Tout programme de la calculatrice commence ainsi. Au lieu de [programme], il faut mettre le nom du programme, sans les crochets. La calculatrice met cette partie automatiquement.
EndPrgm	Tout programme de la calculatrice finit avec ça. La calculatrice met aussi cette partie automatiquement.
Disp "coucou"	Permet d'afficher le texte "coucou". Les guillemets sont obligatoires.
[coucou]→a	Assigne à la variable "a" le résultat de [coucou]. Par exemple, si [coucou] c'est 5, alors "a" vaut 5. Il ne faut pas mettre les crochets.
Disp a	Affiche la variable "a".
Request a	Demande à l'utilisateur du programme combien vaut "a". La variable est quand même affichée dans la partie gauche de la calculatrice lors de l'exécution du programme.
Formules combinées	
Disp "j'aime le nombre", a	Affiche "j'aime le nombre", puis la valeur de "a", sur la même ligne. Il est possible de mettre tout ce qu'on veut, au lieu de "j'aime le nombre".
Request "entrez:", a	Demande à l'utilisateur du programme combien vaut "a", en mettant à côté "entrez:". Au lieu de "entrez:", il est aussi possible de mettre tout ce qu'on veut.
a+2→a	Assigne à "a" la valeur "a+2". Ça revient à rajouter 2 à la variable "a".
a+b→a	Assigne à "a" la valeur "a+b".

Signification de certaines de ces commandes :

- Prgm : abréviation de "programme"
- Request : une "requête" qu'on fait à l'utilisateur du programme
- Disp : abréviation de "display", "afficher" en anglais

Remarque 1 : Les indentations

L'indentation, c'est le fait de mettre des espaces juste avant la ligne de code. Par exemple :

```
Les cookies
  C'est
La vie
```

Ici, la 2^{ème} ligne "C'est" possède une indentation.

Les indentations servent à mieux lire le code, notamment quand on a des boucles et des instructions (on verra ça plus tard, pas la peine de se presser). Dans **certains** langages de programmation, les indentations sont obligatoires.

Pour programmer avec la calculatrice, les indentations **ne sont pas obligatoires**, mais elles permettent quand même de mieux lire le code. Donc quand il faudra faire des programmes compliqués, il faudra mettre des indentations quasi-systématiquement. Même si ça semble inutile pour l'instant, ayez le réflexe d'en mettre à chaque fois, ça sera très utile plus tard.

Remarque 2 : le pseudo-code

Le pseudo-code, c'est une étape avant d'écrire le vrai code. Il est mis d'une manière simplifiée, et en français parce qu'on parle français nous-autres. Par exemple, toutes les instructions qui sont dans les exercices du cahier d'algorithmique sont en couleur, et faites avec des phrases : ce sont des pseudo-codes. En soi, le pseudo-code ne fonctionne dans aucun langage de programmation, il aide juste à mieux comprendre.

Exemple de "vrai" code	Exemple de "pseudo-" code
Define ABCD()= Prgm 9→a Disp "A=", a EndPrgm	Définir le programme "ABCD" Début du programme a prend la valeur 9 Afficher "A=" puis a Fin du programme

C'est quand même beaucoup plus facile de comprendre le pseudo-code, du premier coup d'œil.

Pourquoi je vous parle de pseudo-code ? Parce que je vais en utiliser beaucoup dans ce qui va suivre.

Quelques calculs utiles

Ces calculs ne sont pas indispensables pour comprendre le reste de la fiche, ils sont juste là parce qu'une certaine personne de la classe ne savait pas comment les faire.

Calculs simples	
Remain(a,b)	Donne le reste de la division entière de a par b
Floor(a)	Donne l'arrondi à l'inférieur de a
Ceiling(a)	Donne l'arrondi au supérieur de a
Bonus	
Floor(a:b)	Donne l'arrondi à l'inférieur de a divisé par b, ce qui revient à donner le quotient de la division entière a/b !