

Principales commandes utiles à la réalisation d'un programme : Casio

Pour écrire, modifier ou exécuter un programme

Dans  choisir le menu « PRGM »
puis choisir avec les touche F1 à F6 soit le nom d'un programme, puis EXE pour l'exécuter, soit NEW pour créer un nouveau programme, soit EDIT pour modifier le contenu d'un programme existant

Pour saisir les entrées du programme

Appuyer sur SHIFT puis  on voit alors apparaître diverses commandes dont  (éventuellement après avoir appuyé sur une flèche pour faire défiler les commandes proposées)

Il suffit alors , pour saisir une variable appelée A d'entrer   A

Pour que le programme affiche un nombre

Il suffit de mettre après le nombre dont on souhaite l'affichage le symbole  que l'on trouve à l'écran en faisant SHIFT puis 

Rq: En principe les Casio affichent le dernier nombre du programme, il est donc inutile d'en demander l'affichage ou alors celui-ci apparaîtra deux fois si on a demandé son affichage.

Pour écrire une condition avec « Si... alors... sinon... »

On trouve les commandes en appuyant sur SHIFT puis  puis à l'écran 
On obtient alors

If Then Else IEnd

à savoir « si; alors; sinon; fin de si » la dernière instruction I-End étant à écrire à la fin de la boucle conditionnelle constituée de

« si *condition* alors *instruction* sinon *instruction* I-End »

<p><u>ex</u>:</p> <pre>"COTE LE PLUS LONG"↵ ?→A↵ "AUTRE COTE"↵ ?→B↵ "DERNIER COTE"↵ ?→C↵ If A²=B²+C²↵ Then "RECTANGLE"↵ Else "PAS RECTANGLE"↵ IfEnd</pre>	<p>Le programme ci contre détermine si un triangle dont on connaît les longueurs est rectangle: On affecte trois nombres correspondant aux côtés du triangle dont on se demande s'il est rectangle, en commençant par celui qui peut être l'hypoténuse. Si l'égalité de Pythagore est vérifiée, le triangle est rectangle (réciproque du Th) et le programme renvoie « RECTANGLE » sinon le programme renvoie « PAS RECTANGLE » (d'après la contraposée du Th)</p>
---	--

Rq: En cas d'erreur, pour ne pas tout retaper, on peut utiliser « insérer »: SHIFT puis , pour entrer du texte, on peut bloquer le clavier en mode texte en faisant SHIFT puis ALPHA, en cas d'erreur, (*syntax error*) en appuyant sur la flèche de gauche on retourne dans le programme et le curseur se place au niveau de l'erreur

Pour écrire une boucle avec « tant que ... alors... »

On trouve les commandes en appuyant sur SHIFT puis  puis à l'écran  et ensuite en faisant défiler les menus proposés on finit par obtenir à l'écran:

While WEnd Do Lp-W

à savoir: While(tant que) While end (fin du tant que) Do (faire) Lap-While (qu'on n'utilisera pas)

On entre en général :

« tant que *condition* faire *instruction(s)* fin du tant que) »

mais on peut remplacer le « faire » par un simple retour à la ligne.

<p>ex:</p> <pre>?→A↵ ?→B↵ 0→C↵ While A<B↵ A×1.1→A↵ B+2→B↵ C+1→C↵ WhileEnd↵ C↵</pre>	<p>Dans le programme ci-contre, on demande deux nombres, A et B par exemple 500 et 600, puis on affecte à C la valeur 0. Ensuite tant que $A < B$ on remplace A par $1,1A$ et B par $B+2$, C augmentant de 1. A la fin, on affiche C (c'est à dire le nombre de fois où on a fait la boucle) Avec $A=500$ et $B=600$, on a bien $500 < 600$ donc A devient $500 \times 1,1 = 550$ et B devient 602, et C prend la valeur 1 Comme $550 < 602$ on refait la boucle, A prend alors la valeur $550 \times 1,1 = 605$ tandis que B devient 604 et C prend la valeur 2 Comme $605 > 604$ on ne refait pas la boucle, et on affiche la valeur de C: « 2 »</p>
--	--

Pour écrire une boucle avec « pour... de...à... »

On trouve les commandes en appuyant sur SHIFT puis  puis à l'écran  et ensuite en faisant défiler les menus proposés on finit par obtenir à l'écran:

For To Step Next

à savoir «for = pour » « to= jusqu'à (dans ce cas) » « step = pas (au sens de pas à pas) » et « next = suivant »

On entre en général: « Pour *une variable de une valeur entière* jusqu'à *une valeur entière* liste d'instructions Suivant »

En fait la syntaxe du début est un peu plus compliquée car le « de » est remplacé par la flèche

<p>ex:</p> <pre>For 1→A To K↵ (B+3)²→B↵ A+1→A↵ Next</pre>	<p>Ici l'extrait de programme va faire une boucle pour A allant de 1 à la valeur K qui doit être définie préalablement. Par exemple si $K=3$ et si $B=0$ au départ on aura: $A=1$, donc B devient $(0+3)^2=9$ et A devient $1+1=2$ $A=2$ donc on refait la boucle: B devient $(9+3)^2=144$ et A devient $2+1=3$ $A=3$ on fait une dernière fois la boucle: B devient $(144+3)^2=21609$ et A devient $3+1=4$ or K était égal à 3: on passe à la suite du programme.</p>
---	--

Rq: « step » sert lorsqu'on ne veut pas que la valeur augmente de 1 en 1 mais par exemple de 2 en 2, ou de 0,5 en 0,5...

Pour trouver les symboles >, <, = ...

Le symbole « = » se trouve au clavier ici en faisant SHIFT  pour les autres il faut passer par

SHIFT puis  et en faisant défiler on obtient **REL** dans lequel on trouve

= ≠ > < ≥ ≤

Pour trouver les liens logiques: et, ou, contraire

Appuyer sur  puis faire défiler les menus pour obtenir **LOGIC** dans lequel on trouve or, and, not.