Aide mémoire  
Calculatrice RPN

# Notation Polonaise Inverse

La calculatrice utilise la notation polonaise inverse (RPN, Recverse Polish Notation). Dans cette notation, les nombres sont successivement stockés sur une pile et les opérandes doivent être rentrés avant les opérateurs ou fonctions.

Pour effectuer l’opération 1+2, il faut rentrer :

1 RETURN 2 +

L’appui sur la touche RETURN permet de stocker 1 sur la pile. L’appui sur + stocke 2 sur la pile avant de réaliser l’addition.

Pour les expressions complexes, les parenthèses ne sont pas utilisées ainsi le calcul de cos(1)+(1+2) x3 s’effectuera avec la séquence :

1 COS 1 ENTER 2 + 3 \*

La touche ENTER permet de mettre le nombre sur la pile et d’éditer un secon nombre. Lorsque l’on tape une fonction, il n’est pas nécessaire de taper ENTER.

Sur cette calculatrice la taille de la pile n’est limitée que par la mémoire.

# Opérandes et Fonctions mathématiques

* Les opérandes arithmétiques sont + - \* /. Ils consomment deux objets sur la pile et retournent un résultat.
* Les opérandes logiques sont = <> < <= > >=. Ils effectuent la comparaison entre les deux derniers objets de la pile et renvoient 1 si vrai et 0 si faux.

Exemple :

1 2 <

renvoit vrai car 1 est inférieur à 2.

* Les fonctions mathématiques sont :

COS, SIN, TAN, ACOS, ASIN, ATAN, EXP, LOG, ABS, SGN

Elles consomment le sommet de la pile et renvoient un résultat.

# Manipulation de la pile

* SWAP inverse les deux objets placés au sommet de la pile.
* DROP supprime le sommet de la pile.

# Variables

Stocker une valeur dans une variable

* STO (valeur ‘nom’ -- ) stocke une valeur dans la variable nom.

Exemple stocker la valeur 1 dans la variable A

1 ‘A’ STO

RCL (‘nom’ -- valeur) Rappelle le contenu de la variable nom sur la pile

Exemple rappeler le contenu de la variable A

‘A’ RCL

# Complexes

Un nombre complexe peut être défini à partir du sommet de la pile. Par exemple pour définir le nombre 1+2j, on rentre :

1 ENTER 2 COMP

Les opérateurs +, - \* / fonctionnent sur des complexes.

Les fonctions ARG, ABS permettent de calculer l’argument et le module du nombre complexe. REAL et IMAG fournissent la partie réelle et imaginaire.

# Programmation : fonctions utilisateurs

Une fonction utilisateur est une suite de mots séparées par des espaces.

Par exemple la fonction LOG2 (logarithme en base 2) peut être définie par :

LOG2 : LOG 2 LOG /

Les mots peuvent être :

* Des nombres flottants. Ils seront alors placés sur la pile à l’exécution.
* Une chaine ntourés d’apostrophes.
* Une fonction ou un opérateur nterne tels que / ou LOG
* Une fonction utilisateur.

## Instructions conditionnelles

Les instruction conditionnelles sont IF END et IF ELSE END. Par exemple la fonction signe peut se définir :

SGN : 0 < IF -1 ELSE 1 END

Boucles

* La boucle while permet d’éxécuter une suite de commandes lorsque une condition est vraie. Elle s’écrit :

WHILE condition DO END

La boucle FOR exécute un suite de commande pour un compteur variant entre deux valeurs. Par exemple pour calculer cos(n) pour n variant de 1 à 10 on écrit :

1 10 ‘N’ FOR ‘N’ RCL COS END

Lorsque l’incrément n’est pas 1 ont peut utiliser la commande NEXT à la place de END. Pour générer les 10 premiers nombres pairs on peut écrire :

2 10 ‘N’ FOR ‘N’ RCL 2 NEXT