

TI-82, TI-83 et les bases de numération

Pour les calculatrices à affichage graphique de Texas Instruments, le passage de la TI-81 aux TI-82 et TI-83 a été marqué, entre autres choses, par l'apparition des *listes*. Il s'agit d'ensembles ordonnés qui peuvent comprendre, selon la mémoire disponible, jusqu'à 99 éléments dans la TI-82, jusqu'à 999 dans la TI-83 et la TI-83 Plus. Les concepteurs ont choisi les accolades pour dénoter les listes, bien qu'il s'agisse d'ensembles ordonnés.

La TI-82 peut recevoir jusqu'à six listes et elle les sauvegarde dans six mémoires spécialisées nommées L_1, L_2, \dots, L_6 . La TI-83 possède elle aussi les mémoires L_1, L_2, \dots, L_6 , mais elle accepte en outre des listes auxquelles l'utilisateur peut donner les noms de son choix.

Il est manifeste, à la lecture des manuels d'utilisation, que les listes sont destinées d'abord aux calculs statistiques. Les deux exemples d'applications des listes que je présente ici se rapportent plutôt à la théorie des nombres.

BDIX : d'une base B à la base 10

```
PROGRAM:BDIX
:Input "BASE DE DEPART ",B
:Input "LISTE CHIFFRES ",L1
:dim (L1)→C:0→D
:For(J,1,C)
:D*B+L1(J)→D
:End
:Disp D
```

Le programme BDIX traduit en base 10 un nombre qui est donné dans une base B quelconque. Il demande d'abord à l'utilisateur d'indiquer la base de départ. Disons que c'est 50. Le programme demande ensuite la liste des chiffres. En base 50, les chiffres vont de 0 à 49. Si on répond par la liste $\{1, 49\}$, le programme donnera la bonne réponse : 99.

Les chiffres en base B sont sauvegardés dans la liste L_1 et le programme les rappelle à la suite par la commande $L_1(J)$.

DIXB : de la base 10 à une base B quelconque

Dans le programme DIXB, la manipulation de listes est plus élaborée. La commande $J \rightarrow \dim(L_1)$ fait passer la dimension de la liste L_1 à J, sans affecter les éléments déjà sauvegardés dans L_1 . Ainsi la liste L_1 s'allonge à mesure que sont calculés les chiffres du nombre N en base B. La commande $\dim(L_1) \rightarrow C$ sauvegarde la dimension de la liste L_1 dans la variable C.

```
PROGRAM:DIXB
:Input "NOMBRE A TRAITER",N
:Input "BASE VISEE ",B
:0→J
:While N>0
:J+1→J:J→dim(L1)
:N-(int(N/B))*B→L1(J)
:int(N/B)→N
:End
:dim(L1)→C:C→dim(L2)
:For(K,1,C)
:L1(K)→L2(C-K+1)
:End
:Disp L2
```

Le programme demande d'abord à l'utilisateur le nombre à traiter. Disons que c'est 1999. Il demande ensuite la base visée. Disons que c'est la base 10. Il donne ensuite la réponse sous forme de liste. Dans notre exemple, c'est la liste $\{1, 9, 9, 9\}$. Cet exemple n'est pas très utile, bien sûr. Si on demande au programme comment exprimer 33 en base 2, il répondra par la liste $\{1, 0, 0, 0, 0, 1\}$, comme il se doit. ■