

Un mystère BASICO-fracal...

par Gérard A. Langlet

```
1 INPUT N:SCREEN 1:CLS:FOR Y=1 TO N:CIRCLE (Z,Y),1,1:FOR X=1 TO N:  
LINE -(X,Y),ABS(POINT(P,Y)=POINT(X,Q)):P=X:NEXT X:Q=Y:CIRCLE (Z,Y),1,Z:  
NEXT Y:WHILE LEN(INKEY$)=Z:MEND
```

Le programme ci-dessus est un bon candidat pour un concours de l'été:

Comment dessiner en BASIC sur PC et compatibles, une *FIGURE FRACTALE* dérivée du tamis de Sierpinski, de la manière la plus simple qui soit.

a) Le programme doit avoir *une* ligne au maximum: c'est facile, le BASIC admet le séparateur d'instructions (";")... mais tout recours à GOSUB est prohibé, et GOTO serait peu utile...

b) Le programme ne doit comporter *aucune* variable dimensionnée (hmm!), c'est déjà plus difficile.

c) Le programme, qui ne doit contenir *aucun* branchement, ne doit pas contenir *non plus* de condition If (évidemment...).

d) Le programme ne doit pas comporter *d'autre* constante explicite que 1, mais nous remarquerons alors que le numéro de ligne sera forcément 1, puisque le programme ne doit avoir qu'une seule ligne...

e) Le programme ne doit faire appel à *aucun* opérateur arithmétique et utiliser un *minimum* de fonctions.

La solution que nous avons trouvée n'en utilise qu'une seule, considérée comme *arithmétique*: il s'agit de ABS() - certainement la plus simple -, et encore, uniquement parce que ce BASIC code le résultat VRAI (TRUE) des opérations logiques comme -1 et non pas 1, contrairement aux conventions de l'algèbre binaire ou d'autres implantations plus rigoureuses du BASIC (par exemple SUPERBASIC sur Sinclair QL). Mais à partir du moment où l'on a déjà utilisé ABS, il devient très facile d'éliminer tout recours à la constante 1 (en dehors du numéro de ligne): en effet, il suffit d'écrire: U=ABS(Z=Z) puis de remplacer toute référence à 1 par U.

Le signe "-" utilisé dans l'instruction LINE -(X,Y) n'est pas un signe *moins*: il indique simplement que le couple (X,Y) représente les coordonnées du point final, le point initial étant alors par défaut le point courant. Cette syntaxe est propre à ce BASIC.

Nous avons utilisé la fonction CIRCLE, quelle idée bizarre... C'est uniquement parce que ce BASIC ne possède pas d'instruction PLOT(x,y) et qu'un cercle de rayon 1 fait alors parfaitement l'affaire pour dessiner un simple point. Dans d'autres implantations, par exemple en BASIC GFA sur Atari, une instruction PLOT remplace avantageusement à la fois les instructions CIRCLE et LINE.

La fin du programme, donc de son unique ligne, fait appel à la fonction INKEY\$ à l'intérieur d'une boucle WHILE sans instructions: c'est une astuce pour ne pas polluer l'écran et son beau "fractal" si prestement dessiné, par impression d'un malencontreux message "OK". Une autre manière de procéder serait d'entreprendre la règle de branchement et de remplacer la boucle WHILE par GOTO 1, mais on ne pourrait plus obtenir le graphique final par recopie de l'écran sur une imprimante... sauf avec de la chance.

Le programme demande N : c'est une façon de ne pas introduire de constante dans la programmation. SCREEN 1 force le mode graphique CGA (200 par 320 pixels). Cela tombe bien: le programme fonctionnera sur tous les PC munis d'une carte CGA, VGA ou EGA; on peut répondre par une valeur positive inférieure à 200 (donc 199 au maximum) sans risquer de diagnostic. Il est facile d'accélérer encore l'exécution de ce programme en tenant compte de propriétés de symétrie, mais alors des instructions IF deviennent nécessaires, en contradiction avec la règle c).

f) Le programme ne doit faire appel qu'à UN SEUL OPERATEUR LOGIQUE. Nous avons interdit les opérateurs arithmétiques, mais alors là, c'en est trop. En fait, il existe PLUSIEURS solutions à cette condition apparemment draconiennes; celle que nous avons choisie rend le résultat encore plus étrange, à cause de l'ambiguïté de la notation dans toutes les versions du BASIC: le signe "=" est à la fois l'affectation d'une variable et un opérateur logique.

Ce programme délire-t-il l'imagination et le sens commun de la programmation?

"Et pourtant", comme disait Galilée, "il tourne...".

Mais COMMENT et POURQUOI?

Le problème est en fait beaucoup plus important qu'il n'y paraît. La réponse est autant un caractère philosophique que mathématique. En fait, elle est LOGIQUE, comme la Nature, qui produit des fractals tout autour de nous - et EN NOUS (poumons, réseau sanguin, cortex cérébral, etc...). Comme les biologistes le découvrent peu à peu en allant d'étonnement en étonnement...

Cette programmation réductrice et quelque peu baroque n'est pas due au hasard. Elle est le fruit de considérations théoriques, démontrables (et démontrées) à partir de quelques évidences, ce dont nous reparlerons un jour.

A titre d'information, il est possible d'engendrer cette figure de manière encore beaucoup plus courte, mais... pas en BASIC.