

smakynetinfo

Nouvelles possibilités graphiques dans le BASIC

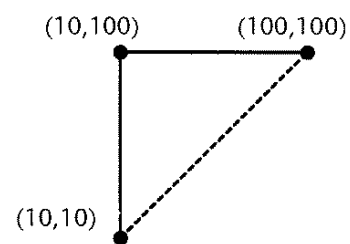
La version 9.21 du BASIC comporte la possibilité de remplir des polygones de forme quelconque avec des trames. Jusqu'à maintenant le remplissage n'était possible qu'avec l'ordre FILL dont les principaux désavantages étaient une certaine lenteur et une "animation" un peu bizarre à regarder, due à l'algorithme utilisé.

Il n'y a pas de nouveaux ordres mais seulement un nouveau type de trait numéro 15 pour TYPNT. Rappelez-vous que l'ordre TYPNT permet de spécifier un type de trait pour les commandes de dessins de traits et de cercles subséquents. Après l'exécution de l'ordre Typnt(15), les traits dessinés avec MOVETO et MOVETOR ne sont plus tracés à l'écran, mais ils définissent un contour de polygone. Au moment de l'exécution de l'ordre TYPNT avec un autre argument que 15, le contour du polygone est rempli avec la trame courante (définie avec TYPRASST).

Polygone simple

Pour dessiner un triangle rempli on peut procéder comme ceci:

```
100 Typnt (15) — passe en mode définition de polygone
110 Skipto (10,10)
120 Movetor (0,100)
130 Movetor (100,0)
140 Typnt (0) — le triangle rempli est dessiné
```

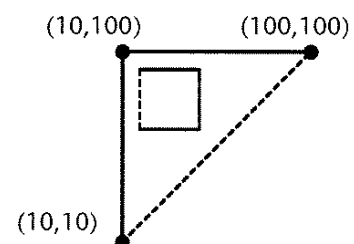


Vous constatez que seulement deux des trois côtés du triangle sont dessinés avec MOVETOR. Cela est suffisant, car le polygone est automatiquement fermé par le BASIC au moment du TYPNT(0).

Polygone comportant des trous

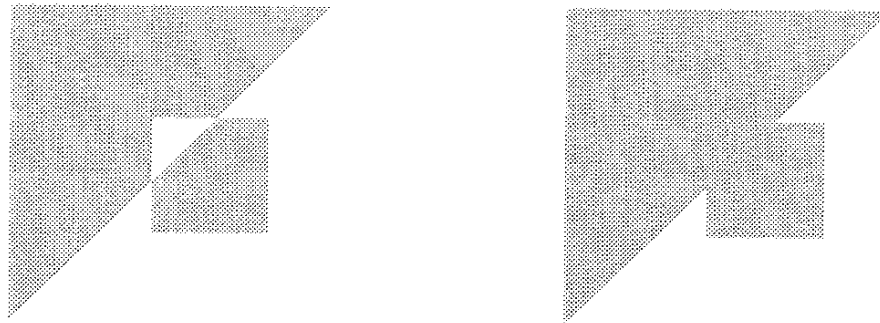
Il est également possible de dessiner des polygones comportant des "trous" sous forme d'autres polygones. Le programme suivant dessine un triangle rempli avec un trou carré au milieu.

```
100 Typnt (15)
110 Skipto (10,10) — dessine le triangle
120 Movetor (0,100)
130 Movetor (100,0)
140 Movetor (-100,-100)
150 Skipto (20,70)
160 Movetor (30,0) — dessine le carré
170 Movetor (0,30)
180 Movetor (-30,0)
190 Typnt (0) — dessine le triangle avec le trou
```



Dans un cas particulier on ne parle plus de polygone mais de *région* qui est un concept plus général. Une *région* est définie par un *chemin*, qui n'est rien d'autre que qu'une suite de MOVETO ou MOVETOR et de SKIPTO ou SKIPTOR. On peut également définir un chemin comme étant un ensemble de polygones.

Dans le dernier exemple le chemin définissant la région à tramer est composé de deux polygones, le grand triangle et le petit carré.

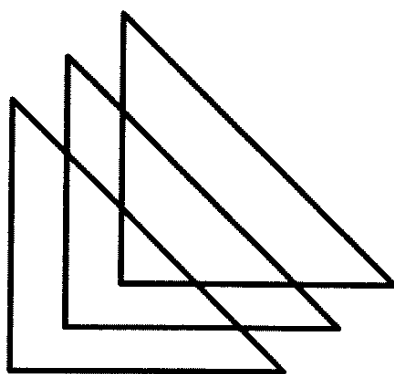


Dans le cas où plusieurs polygones d'un chemin s'intersectent il se pose la question suivante: est-ce que les points qui se trouvent à l'intérieur de plusieurs polygones font partie de la région ou non ? La réponse est simple. Un point fait partie de la région s'il se trouve à l'intérieur de n polygones définissant le chemin où n est un nombre impair. L'exemple suivant illustre la situation. Le chemin de la région est composé de trois triangles partiellement superposés.

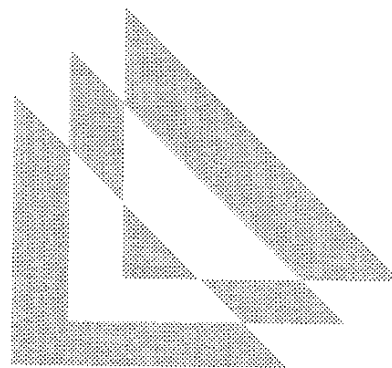
```

100 Typnt (15)
110 Gosub Label_TRIANGLE (10,10)
120 Gosub Label_TRIANGLE (25,25)
130 Gosub Label_TRIANGLE (40,40)
140 Typnt (1)
150 End
160 Label_TRIANGLE(X,Y)
170 Skipto (X,Y)
180 Movetor (100,0)
190 Movetor (-100,100)

```



Chemin



Région

Le petit triangle à l'intérieur est la réunion des trois (nombre impair) triangles, donc il fait partie de la région. Les parties qui sont l'intersection de deux (nombre pair) triangles ne font pas partie de la région.

Nouveaux typ de trait rapides

Pour des raisons historiques et de compatibilité, le BASIC n'utilise pas les fonctions du système pour effectuer le tracé de traits et de cercles, mais ses propres fonctions. Trois nouveaux types de traits sont maintenant disponibles:

TYPNT(17) Points allumés

TYPNT(17) Points éteints

TYPNT(17) Points inversés

Ces types sont semblables aux types 1 à 3, mais le dessin est environ deux fois plus rapide. MOVETO et MOVETOR avec TYPNT 17 à 19 ne réagissent cependant pas tout à fait de la même façon qu'avec TYPNT 1 à 3.

Nouvelles fonctions TESTPC et TESTPCR

Ces deux nouvelles fonctions permettent de déterminer la couleur d'un pixel à l'écran. Elles sont semblables aux fonctions TESTP et TESTPR, mais elles ne donnent pas seulement TRUE si le pixel est allumé ou FALSE sinon, mais également le numéro de la couleur.

256 Couleurs

Enfin il est possible d'utiliser 256 couleurs en BASIC. Dès que BASIC tourne en mode 256 couleurs vous pouvez utiliser les couleurs 16–255 avec les ordres Colorset et Colorclr. Sur un Smaky 300 les couleurs 16–255 donnent des résultats non prévisibles à l'écran. La fonction Color(0) donne le nombre de couleurs disponibles, 2 (noir blanc), 16 ou 256.

Compression de données avec ZIP et UNZIP

Les volumes des données devenant de plus en plus importants, il est aujourd'hui indispensable de pouvoir compresser des données, afin qu'elles prennent moins de place sur un support ou qu'elles soient envoyées plus rapidement par modem.

On distingue deux types de compression, la compression avec pertes et la compression sans pertes. La compression avec pertes est valable dans le domaine des images et du son où, grâce à des algorithmes astucieux, des taux de compression impressionnants peuvent être réalisés sans perte de qualité significative. Une telle compression n'est évidemment pas utilisable avec du texte ou toute autre type de données.

ZIP et UNZIP sont deux utilitaires qui permettent de réunir plusieurs fichiers en un seul fichier avec l'extension .ZIP tout en comprimant le tout sans pertes. Un tel fichier s'appelle fichier ZIP ou archive ZIP. Le format des fichiers ZIP est standard et les utilitaires ZIP et UNZIP existent pratiquement sur tous les systèmes d'exploitation. Le taux de compression moyen est d'environ 50 %: après compression les données n'occupent plus que 50 % de la place qu'elles occupaient initialement.

Le programme ZIP permet de créer des archives ZIP et le programme UNZIP permet d'extraire des fichiers d'une archive ZIP. Ce sont des programmes non interactifs de type "ligne de commande" et ils doivent être exécutés depuis le Filer. Il ne faut pas cliquer sur ZIP.CODE ou UNZIP.CODE depuis START.

Créer une archive ZIP

La syntaxe du programme ZIP est la suivante:

```
zip [options] fichier liste [-x liste_à_exclure]
```

fichier	nom du fichier ZIP
liste	liste des fichiers à inclure dans le fichier ZIP. Cette liste peut très bien inclure des jokers (* et ?) afin de spécifier un groupe de fichiers.
liste_à_exclure	liste des fichiers qui ne doivent pas être archivés
options	diverses options (voir plus bas).

Exemple:

```
zip archive *.text *.image
```

 crée le fichier ARCHIVE.ZIP et comprime tous les fichiers avec les extensions TEXT et IMAGE. Après cette opération, le fichier ARCHIV.ZIP contient tous les fichiers TEXT et IMAGE du dossier courant.

Les options les plus utiles sont les suivantes:

- r traite également les sous-dossiers
- f met à jour les fichiers dans l'archive. Les fichiers qui ne sont pas encore dans l'archive sont ignorés
- u met à jour les fichiers dans l'archive
- x exclut les fichiers qui suivent -x
- m détruit les fichiers après les avoir transférés dans l'archive ZIP

Exemple: `zip -f ar a* -x *.code`
extrait tous les fichiers qui commencent avec a sauf les fichiers avec l'extension .code mais seulement si le fichier existe déjà sur disque.

Extraire des fichiers d'une archive ZIP

La syntaxe du programme UNZIP est la suivante:

```
unzip [options] fichier liste
```

fichier	nom du fichier ZIP
liste	liste des fichiers à extraire
options	diverses options (voir plus bas)

Exemples:

```
unzip archive mozart.text
```

 extrait le fichier mozart.text de l'archive ARCHIVE.ZIP

```
unzip archive
```

 extrait tous les fichiers de l'archive ARCHIVE.ZIP

Les options les plus utiles sont les suivantes:

- f met à jour les fichiers dans le dossier courant. Si le fichier existe déjà et qu'il est plus ancien que le fichier dans l'archive, il est mis à jour sur disque.
- v liste les fichiers dans l'archive avec plus d'informations.
- l liste les fichiers dans l'archive
- u comme -f mais extrait aussi les fichiers qui n'existent pas déjà.

Important: tapez les commandes ainsi que les options de unzip et zip en minuscules !!