

## TM SCHEME n°5 : Fractales

Dans ce TM, nous allons dessiner quelques fractales simples à l'aide de la tortue. Les dessins de fractales sont des figures géométriques généralement construites à partir de motifs simples que l'on peut composer à l'infini (en théorie) suivant des règles précises.

### 1. Flocon de Von Koch

Cette fractale, parfois appelée île de Von Koch, est construite sur la base d'un triangle équilatéral. Un flocon de Von Koch de génération 0 est un triangle équilatéral et un flocon de génération 1 est une étoile à 6 branches. La figure ci-dessous présente également les flocons de génération 2 et 3.

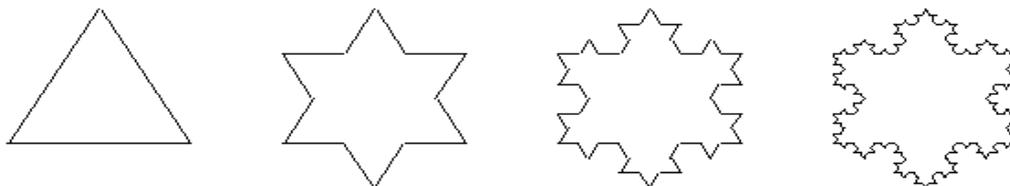


Figure 1 : flocons de Von Koch de générations 0, 1, 2 et 3

#### 1.1 Tracé de côtés

En observant la figure 1, on peut remarquer que ce qui change entre le tracé d'un flocon de génération 0 et celui d'un flocon de génération 1, c'est la façon de tracer les côtés du triangle équilatéral : chaque côté est remplacé par 4 segments. On peut donc considérer que le tracé d'un flocon de génération  $n$  ( $\forall n \geq 0$ ) correspond au tracé de 3 côtés, constituant un triangle équilatéral, ces côtés étant de génération  $n$ . La figure 2, ci-dessous, montre les côtés de générations 0 et 1.

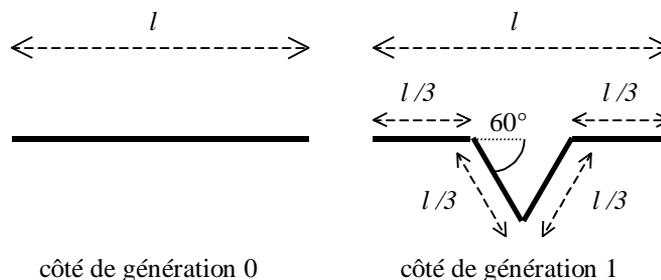


Figure 2 : côtés de générations 0 et 1

Tracer un côté de génération  $n$  de longueur  $l$ ,  $\forall n > 0$ , revient à tracer 4 côtés adjacents de génération  $n-1$  de longueur  $l/3$ , orientés les uns par rapport aux autres d'un angle de  $\pm 60^\circ$  comme le montre la figure 2.

Définir une fonction sans retour `traceCote` qui trace un côté d'un flocon de Von Koch, et qui prend en paramètres la génération et la longueur du côté à tracer.

#### 1.2 Tracé de flocons

Ecrire une fonction sans retour `flocon`, permettant de tracer un flocon de Von Koch, prenant en paramètres la génération et la longueur du côté du flocon à tracer. Il est déconseillé de tester cette fonction avec une génération supérieure à 7 ou 8 suivant les performances de votre ordinateur.

#### 1.3 Programme principal

Ecrire un programme traçant un flocon de Von Koch, centré dans la fenêtre de la tortue, après avoir demandé à l'utilisateur la génération et la longueur du côté du flocon à tracer.

## 2. Triangle de Sierpinski

Le triangle de Sierpinski est une autre fractale. La figure ci-dessous montre les triangles de Sierpinski de génération 0 à 3.



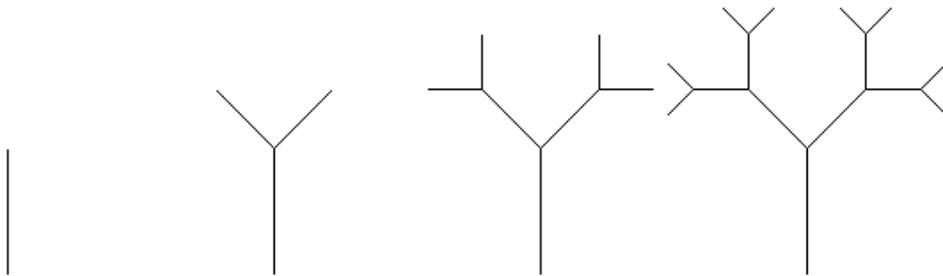
**Figure 3 : triangles de Sierpinski de génération 0 à 3**

Il y a plusieurs manières de décomposer le tracé d'un triangle de Sierpinski de génération  $n$  en tracés de triangles de Sierpinski de génération  $n-1$ . On peut par exemple considérer que le triangle de génération 1 est constitué de 3 triangles de génération 0, correctement placés les uns par rapport aux autres. On peut aussi décomposer le tracé en traçant l'un des côtés du triangle d'une manière spécifique par rapport aux autres.

Ecrire un programme, permettant de tracer un triangle de Sierpinski centré dans la fenêtre de tracé, après avoir demandé à l'utilisateur d'en saisir la génération et le côté. Un test du programme avec une génération supérieure à 8 peut s'avérer long, suivant les performances de l'ordinateur.

## 3. Un brocolis ?

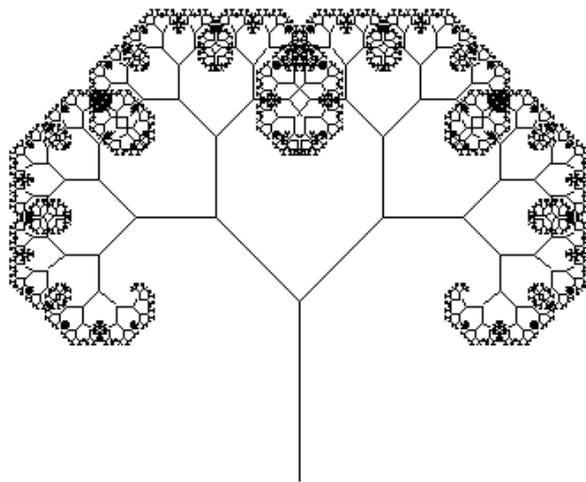
On s'intéresse maintenant à une fractale construite sur la base d'un Y, qui finit par ressembler à une coupe de brocolis. La figure 4 montre les premières générations de cette fractale.



**Figure 4 : Générations 0 à 3 de la fractale en Y**

La génération 1 de cette fractale forme un Y : l'angle entre les 2 branches est un angle droit, la longueur d'une branche est égale au 2 tiers de la longueur du pied du Y.

Ecrire un programme qui demande à l'utilisateur la longueur de la branche principale et la génération, et dessine une fractale Y. Si vous êtes capables de centrer cette figure sur la fenêtre de tracé, faites-le. Vous pouvez utiliser ce programme pour obtenir un brocolis avec une génération de 12, par exemple : le résultat alors obtenu est montré dans la figure 5.



**Figure 5 : génération 12, longueur 100, bon appétit !**