

Fonctions Récursives (2)

Compétences

- Découvrir le module turtle de Python
- Récursivité et figures fractales

Dessiner avec une tortue

Pour pouvoir utiliser ce module, il faut l'importer au début du programme :

```
1 from turtle import *
```

Ensuite tu peux donner un ordre à ta tortue :

```
1 forward(200)
```

Cette commande fait avancer la tortue de 200 pixels. Une commande pour contrôler la tortue a la forme suivante :

- une commande (forward, backward, left, right, etc.)
- des parenthèses ()
- un argument numériques (distance, angle)

Par exemple, pour faire reculer la tortue de 200 pixels, écris ceci :

```
1 backward(200)
```

Pour faire tourner la tortue de 90 degrés vers la gauche :

```
1 left(90)
```

Pour faire tourner la tortue de 45 degrés vers la droite :

```
1 right(45)
```

Le programme suivant fait dessiner un triangle à la tortue.

```
1 forward(200)
2 left(120)
3 forward(200)
4 left(120)
5 forward(200)
6 left(120)
```

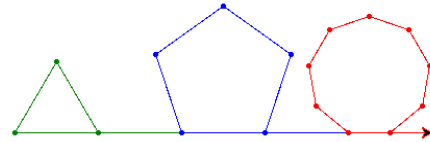
Nous avons vu les 4 commandes de base pour déplacer la tortue : forward, backward, left et right. Mais il est également possible de contrôler d'autres aspects du dessin :

téléporter la tortue	goto(x, y)
épaisseur du trait	width(3)
couleur du trait	pencolor('blue')
monter le stylo	up()
descendre le stylo	down()

Pour s'entraîner

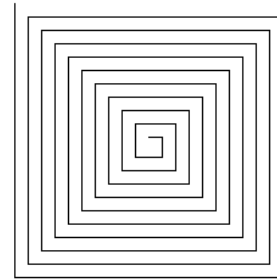
À l'aide de la bibliothèque turtle

1. Dessiner un carré de côté 100 pixels.
2. Écrire une fonction `polygon(n, a)` qui prend en paramètres le nombre de sommets `n` et la longueur d'un côté `a`.
3. Dessiner à la suite trois polygones ayant 3, 5 et 9 cotés.



polygone spirale

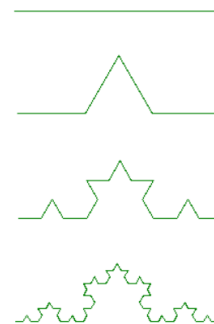
Écrire une fonction récursive python `spirale(cote, delta, angle)` utilisant la tortue. Par exemple, `spirale(10, 5, 90)` devrait donner une figure qui commence ainsi :



Qu'obtient-on avec `spirale(2, 3, 120)` ?

La courbe de Von Koch

Proposer un code permettant de dessiner récursivement :



Le flocon

En déduire la fonction pour dessiner le flocon de Koch ci-dessous

