

# Le module Turtle de Python

Une *tortue* est disponible en standard sous Python. Elle n'est pas très rapide, même pour une tortue, mais permet de réaliser des figures intéressantes.

Voici comment utiliser le module `turtle` de Python, en mode interactif (avec Idle par exemple). La documentation complète est ici : [🐢 turtle.html](https://docs.python.org/3/library/turtle.html)

```
from turtle import *

fd(50)
rt(90)
fd(50)
reset()
speed()
for i in range(6) :
    fd(100)
    rt(360/6)
```

## Liste (non exhaustive) des fonctions du module Turtle

Fonction	Effet
<code>fd(n)</code>	avance de n
<code>bk(n)</code>	recule de n
<code>rt(n)</code>	tourne à droite de n degrés
<code>lt(n)</code>	tourne à gauche de n degrés
<code>clear()</code>	efface l'écran
<code>penup()</code>	lève le stylo
<code>pendown()</code>	baisse le stylo
<code>reset()</code>	efface l'écran, remet la tortue au centre et réinitialise ses paramètres
<code>showturtle()</code>	montre la tortue
<code>hideturtle()</code>	cache la tortue
<code>speed(n)</code>	Change la vitesse de 1(lent) à 10 (rapide). La valeur spéciale 0 est la plus rapide.
<code>tracer(n,d)</code>	
<code>update()</code>	Force l'affichage des graphismes en attente
<code>bye()</code>	Referme la fenêtre
<code>setup(w,h)</code>	Ouvre une fenêtre de taille wxh

## Exemples de programmes

### Frise

[turtlesample.py](#)

```
from turtle import *

def polygone(long,nbcotes) :
    for i in range(nbcotes) :
        fd(long)
        rt(360/nbcotes)

def frise(ang,long,nbcotes) :
    for i in range(720//ang) :
        polygone(long,nbcotes)
        lt(ang)

def main() :
    setup(400,400) # Facultatif
    reset()
    speed()
    tracer(50,)
    frise(3,80,5)
    update()

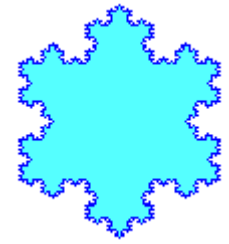
if __name__=='__main__' : main()
```

## Construction du flocon de Von Koch

```
from turtle import *
def flocon(l,n) :
    if n== : fd(l)
    else :
        flocon(l/3,n-1)
        lt(60)
        flocon(l/3,n-1)
        right(120)
        flocon(l/3,n-1)
        lt(60)
        flocon(l/3,n-1)

#clearscreen()
#speed(0)
#tracer(5,0)
flocon(243,1)
update()
penup()
back(243)
pendown()
```

On peut aussi utiliser la couleur comme dans le flocon de von-Koch suivant.



```
from turtle import *
def floc(l) :
    if l<3 :
        fd(l)
        return
    floc(l/3)
    lt(60)
    floc(l/3)
    rt(120)
    floc(l/3)
    lt(60)
    floc(l/3)

def flocon(l) :
    speed()
    color('#0000ff', '#55ffff')
    begin_fill()
    for i in range(3) :
        floc(l)
        rt(120)
    end_fill()

flocon(100)
```

## Divers

Tracez d'autres figures, comme des arbres, le triangle de Sierpinsky, la courbe du C...

La tortue Python peut être accélérée de deux manières



1. en définissant sa vitesse : `turtle.speed(0)`
2. en n'affichant pas toutes les images intermédiaires : `turtle.tracer(100,0)`

## Quelques applications des fractales

- Modélisation de plantes et de paysages
- Modélisation biologique (vascularisation des poumons par ex)
- Antennes de téléphones portables (réduction de la surface et augmentation de la longueur)
- Murs anti-bruits (multiples réflexions pour augmenter l'absorption)

From:

<https://deptinfo-ensip.univ-poitiers.fr/ENS/doku/> - **Informatique, Programmation, Python, Enseignement...**

Permanent link:

<https://deptinfo-ensip.univ-poitiers.fr/ENS/doku/doku.php/stu:python:turtle>

Last update: **2014/03/31 18:45**

