

Tâche n°1 : Le module Turtle

Le module *Turtle* regroupe un ensemble d'instructions qui permettent de faire des dessins.

1. Tapez le programme ci-dessous, l'exécutez et observez le dessin obtenu.

```

from turtle import *
forward(120)
left(90)
forward(50)
done()
  
```

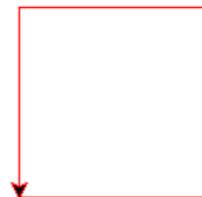


On obtient la moitié d'un rectangle.

2. Modifiez votre programme pour dessiner un carré.

```

from turtle import *
forward(120)
left(90)
forward(120)
left(90)
forward(120)
left(90)
forward(120)
done()
  
```



3. Tapez le programme ci-dessous et comparez avec la question précédente. Selon vous, à quoi sert l'instruction sur la 2^e ligne du programme ci-dessous qui s'appelle **une boucle**.

```

from turtle import *
for i in range(4):
    forward(120)
    left(90)
done()
  
```

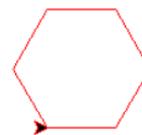
L'instruction **for i in range(4)** : permet de répéter 4 fois la même instruction.

4. Modifiez le programme ci-dessus pour dessiner un hexagone régulier.

Programme

```

from turtle import *
for i in range(6):
    forward(120)
    left(60)
done()
  
```

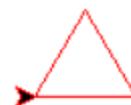


5. Modifiez votre programme pour dessiner un triangle équilatéral.

Programme

```

from turtle import *
for i in range(3):
    forward(120)
    left(120)
done()
  
```



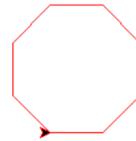
6. Modifiez votre programme pour dessiner un octogone régulier.

Programme

```

from turtle import *
for i in range(8):
    forward(120)
    left(45)
done()

```



Tâche n°2 : Fonctions

Lorsque l'on veut utiliser une partie de code plusieurs fois, on crée une « fonction » à l'aide de l'instruction `def`. Dans le programme ci-dessous, on explique à l'ordinateur comment il doit tracer un triangle équilatéral en précisant sa longueur. Puis dans le programme principal, on fait tracer 2 triangles à l'ordinateur en changeant la longueur de son côté.

1. Tapez puis exécutez le programme ci-dessous. Détaillez l'action de chaque instruction en s'aidant de l'annexe.

<pre> from turtle import * def triangle(longueur): for i in range(3): forward(longueur) left(120) up() goto(100,100) down() fillcolor('red') begin_fill() triangle(60) end_fill() up() goto(-100,-100) pensize(10) pencolor('green') down() triangle(120) done() </pre>	<p>Importe le module Turtle</p> <p>Définition du triangle de longueur donnée</p> <p>Répète 3 fois :</p> <p>Avance de la longueur donnée</p> <p>Tourne à gauche de 120°</p> <p>Lève le crayon</p> <p>Va au point (100,100)</p> <p>Baisse le crayon</p> <p>Remplir en rouge</p> <p>Commence à remplir un triangle de longueur 60 pixels, termine de remplir</p> <p>Lève le crayon</p> <p>Va au point (-100,-100)</p> <p>Épaisseur du crayon 10 px</p> <p>Couleur du crayon vert</p> <p>Baisse le crayon</p> <p>Trace un triangle de longueur 120 px</p> <p>Fin du dessin</p>	 
---	--	--

2. Créer des fonctions qui dessine un carré, un parallélogramme, un hexagone et un octogone.

Tâche n°3 : Compléter et apprendre le bilan

Module Turtle	<code>from turtle import *</code>
Avancer	<code>forward()</code>
Tourner à droite	<code>right()</code>
Tourner à gauche	<code>left()</code>
Fermer la fenêtre Turtle	<code>done()</code>
Répéter des instructions	<code>for i in range() :</code>
<code>goto(x,y)</code>	Déplace le curseur en (x;y)
<code>up()</code> <code>down()</code>	Lève ou baisse le crayon
<code>fillcolor, begin_fill, end_fill</code>	Colorie une figure
<code>pensize</code>	Épaisseur du crayon
<code>pencolor</code>	Couleur du crayon