

Du carré au triangle

Niveau	Domaine	Modules
Seconde professionnelle	Géométrie	
		Algorithmique et programmation

Cette activité est présentée pour une mise en œuvre avec l'outil Capytale mais peut être adaptée à n'importe quel autre environnement Python (Edupython, IDE,...).

ÉNONCÉ ÉLÈVE

Le programme inscrit dans la partie SCRIPT permet de dessiner un carré de côté c (en pixel).

1. Dans la console, saisir `carre(100)` pour obtenir un carré de côté 100 pixels.
2. **forward (c)** signifie que la « tortue » avance de c pixels.
Que signifie **left(90)** dans la dernière instruction ?
3. Combien de fois la boucle (**for i...**) est-elle effectuée ?
4. Modifier le programme pour qu'il puisse dessiner un triangle équilatéral de côté c .

SCRIPT proposé aux élèves :

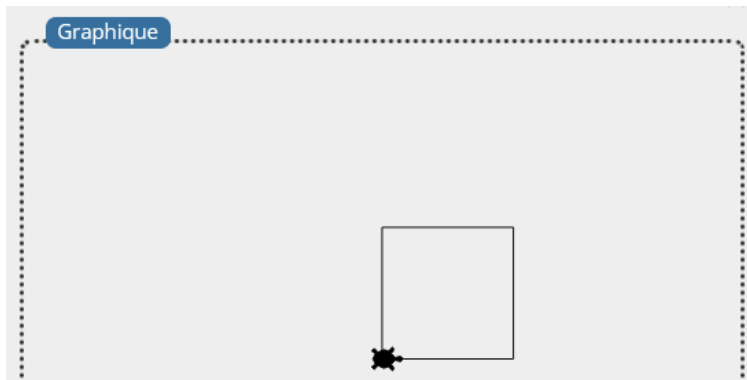
```
1 from turtle import*
2 def carre(c):
3     for i in range(0,4):
4         forward(c)
5         left(90)
6     turtle.done()
```

SCRIPT pour l'enseignant en version à « copier-coller » pour gagner du temps :

```
from turtle import*
def carre(c):
    for i in range(0,4):
        forward(c)
        left(90)
    turtle.done()
```

PROPOSITION DE CORRIGÉ

1. Extrait de la console :



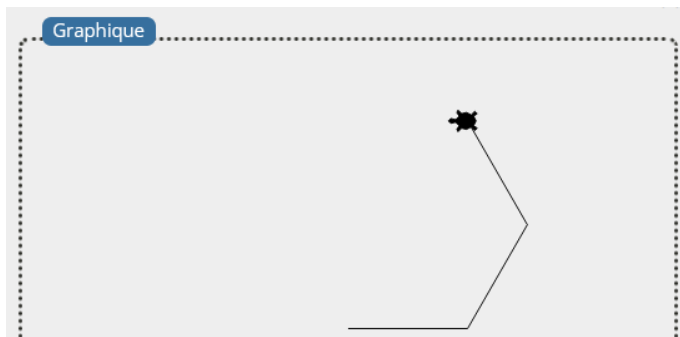
2. **left(90)** signifie que la tortue tourne de 90° vers la gauche (c'est-à-dire dans le sens inverse des aiguilles d'une montre).

3. La boucle est effectuée 4 fois.

4. Un premier « réflexe » est de transformer le programme de la façon suivante :

```
1 from turtle import*
2 def triangle(c):
3     for i in range(0,3):
4         forward(c)
5         left(60)
6     turtle.done()
```

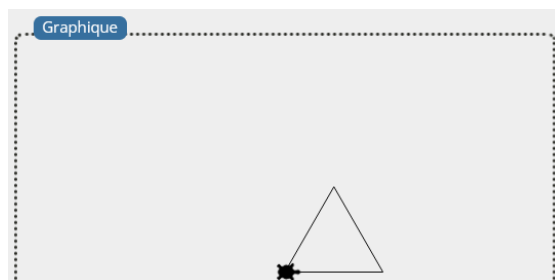
Or, cela donne le graphique ci-dessous :



Il est attendu que l'élève comprenne que son éventuelle erreur est dans l'instruction **left**.

Une proposition de corrigé est la suivante :

```
1 from turtle import*
2 def triangle(c):
3     for i in range(0,3):
4         forward(c)
5         left(120)
6     turtle.done()
```



Il est attendu que l'élève explique pourquoi il faut choisir 120° au lieu de 60°.