

Utilisation du Module Turtle

I Premier exemple

1. Exécuter le programme suivant et décrire son rôle :

```
from turtle import *

forward(200)
right(70)
forward(200)
right(110)
forward(200)
right(70)
forward(200)
```

2. Ecrire un programme permettant de tracer un rectangle de longueur 300 pixels et de largeur 200 pixels.
3. Ecrire un programme permettant de tracer un triangle rectangle dont les côtés de l'angle droit ont pour longueurs respectives 150 pixels et 200 pixels.
4. Exécuter et observer le programme suivant :

```
from turtle import *

def losange():# Définition de la fonction losange()
    forward(200)
    right(70)
    forward(200)
    right(110)
    forward(200)
    right(70)
    forward(200)

def rectangle():# Définition de la fonction rectangle()
    forward(300)
    left(90)
    forward(200)
    left(90)
    forward(300)
    left(90)
    forward(200)

def triangle():# Définition de la fonction triangle()
    x=position()[0]# x prend la valeur de l'abscisse de la tortue
    y=position()[1]# y prend la valeur de l'ordonnée de la tortue
    forward(150)
    left(90)
    forward(200)
    goto(x,y)

color("green")# Le tracé est défini en vert
losange()# Appel de la fonction losange()
color("blue")# Le tracé est défini en bleu
rectangle()# Appel de la fonction rectangle()
color("black")# Le tracé est défini en noir
triangle()# Appel de la fonction triangle()
```

Remarquer le rôle des symboles # pour commenter.

II Premières instructions du module Turtle

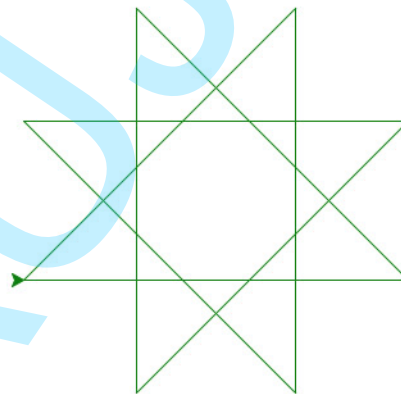
- `from turtle import *` : importe la bibliothèque turtle
- `setup(largeur , hauteur)` : définit les dimensions de la fenêtre en pixels
- `clear()` : efface la zone de dessin
- `reset()` : efface la zone de dessin et réinitialise les paramètres
- `exitonclick()` : ferme la fenêtre au clic de la souris
- `up()` : relève le crayon
- `down()` : abaisse le crayon
- `forward(n)` : avance de n pixels
- `backward(n)` : recule de n pixels
- `left(d)` : tourne de d° vers la gauche
- `right(d)` : tourne de d° vers la droite
- `goto(x , y)` : déplace la tortue au point de coordonnées (x , y)
- `position()` : retourne le couple de coordonnées de la tortue
 - `position()[0]` : abscisse de la tortue
 - `position()[1]` : ordonnée de la tortue
- `color(couleur)` : définit la couleur du tracé
- `width(épaisseur)` : définit l'épaisseur du tracé en pixels
- `write(texte)` : affiche un texte

III Boucles et fonction

1. Exécuter le programme ci-dessous :

```
from turtle import *
color('red')
for i in range(12):
    forward(300)
    left(150)
exitonclick()
```

2. Ecrire un programme permettant de tracer la figure ci-dessous :



3. (a) Exécuter le programme suivant :

```
from turtle import *
def octogone(longueur,couleur):# 1
    color(couleur)
    for i in range(8):
        forward(longueur)
        left(45)
l=int(input("Entrer la longueur d'un côté (en pixels) "))# 2
c=input("Entrer la couleur du tracé (en anglais) ")# 3
octogone(l,c)# 4
```

- (b) Compléter les commentaires et décrire le rôle du programme.
- (c) Ecrire un programme permettant de tracer un rectangle dont la longueur, la largeur et la couleur sont entrées par l'utilisateur.

IV Polygones réguliers

Définition

Un *polygone régulier* est un polygone ayant tous ses côtés de même longueur et tous ses angles de même mesure.

1. Polygone régulier à trois côtés

- Quel est le nom usuel du polygone régulier à trois côtés ?
- Quelle est la mesure de l'angle entre trois sommets consécutifs d'un polygone régulier à trois côtés ?
- En utilisant le module Turtle, tracer un polygone régulier à trois côtés de côté 100 pixels.

2. Polygone régulier à quatre côtés

- Quel est le nom usuel du polygone régulier à quatre côtés ?
- Quelle est la mesure de l'angle entre trois sommets consécutifs d'un polygone régulier à quatre côtés ?
- Adapter le programme précédent pour tracer un polygone régulier à quatre côtés de côté 100 pixels.

3. Polygone régulier à six côtés

- Quel est le nom usuel du polygone régulier à six côtés ?
- Quelle est la mesure de l'angle entre trois sommets consécutifs d'un polygone régulier à six côtés ?
- Adapter le programme précédent pour tracer un polygone régulier à six côtés de côté 100 pixels.

4. Polygone régulier à n côtés

- Donner, en fonction de n , la mesure de l'angle entre trois sommets consécutifs d'un polygone régulier à n côtés.
- Adapter le programme précédent pour :
 - demander à l'utilisateur de saisir le nombre n de côtés souhaité ;
 - tracer un polygone régulier à n côtés de côté 100 pixels.
- Tester votre programme pour tracer un hendécagone régulier (comme tout le monde le sait, un hendécagone est un polygone à onze côtés).

5. Pour aller plus loin

- Pour une fenêtre de taille standard, si le nombre de côté est trop grand, le tracé sort du cadre. Déterminer à partir de combien de côtés le phénomène se produit puis adapter le programme pour que, dans ces cas, le côté du polygone soit de 25 pixels.
- Ecrire une fonction `polygone(n)` qui trace un polygone régulier à n côtés.
Utiliser cette fonction pour tracer les 10 premiers polygones réguliers sur une même figure.
- Ecrire une fonction `polygonec(n , $couleur$)` qui trace un polygone régulier à n côtés en coloris *couleur*.
Utiliser cette fonction pour tracer les 10 premiers polygones réguliers sur une même figure, chacun d'une couleur différente.