

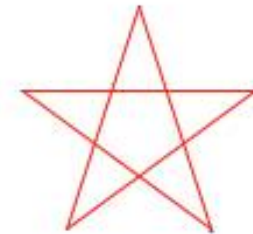
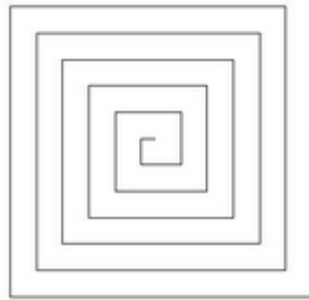
Le module TURTLE:

- ✓ C'est un module qui permet de réaliser des graphiques.
- ✓ Le principe:

Une tortue virtuelle se déplace laissant derrière elle une trace qui représente le dessin.

Les déplacements de la tortue sont contrôlés par des instructions simples du genre : avance, recule.... et qui laisse une trace si le crayon est baissé....

Exemple de réalisation:



Exemples de fonctions utilisées dans TURTLE:

up () : lève le crayon

down (): baisse le crayon

forward (d): fait avancer la tortue de d, le trait est dessiné si le crayon est baissé

backward (d): fait reculer la tortue de d, le trait est dessiné si le crayon est baissé : les distances sont en pixels

left(a): fait pivoter la tortue d'un angle de a degrés vers la gauche

right (a): fait pivoter la tortue d'un angle de a degrés vers la droite

remarque: les angles sont en degrés

reset (): nettoie la fenêtre de dessin, réinitialise la tortue: elle est située alors au centre de l'écran de dessin tournée vers la droite.

color(c): la couleur par défaut est le noir. On peut changer en mettant une couleur: c doit être une chaîne prédéfinie "red", "green", "blue"

bgcolor(c): modifie la couleur de fond (blanc par défaut).

width(e): fixe la largeur du trait (en pixels)

write (ch): permet d'écrire la chaîne ch là où est la tortue

dot(d,c): dessine un disque de diamètre d et de couleur c là où est la tortue.

circle(r): trace un cercle de rayon r, le point de départ de la tortue appartient au cercle (attention il n'est pas centré sur la position de la tortue)

circle(r,s): trace une portion du cercle correspondant à s degrés.

shape(f) : où f est une forme permet de changer la forme du curseur ; f peut prendre les valeurs "turtle", "classic" (valeur par défaut), "circle", "square", "triangle", "arrow".

goto(x,y): la tortue va se positionner au point de coordonnées (x,y) (le dessin est tracé si le crayon est baissé).

setheading(a): où a est en degré permet de fixer un cap absolu à la tortue

speed (vitesse): permet de fixer une vitesse à la souris. L'argument peut être 'fastest' 'fast' 'normal' 'slow' ou 'slowest' .

position() : retourne les coordonnées de votre crayon

setx(integer) : se déplace à la coordonnée x spécifiée

sety(integer) : se déplace à la coordonnée y spécifiée

hideturtle() et **showturtle()** permettent de supprimer le curseur ou de le remettre

Pour plus d'info : <http://docs.python.org/library/turtle.html>

s.loutfi

Applications:

1. Dans un fichier nommé *tortue.py* reprendre le script suivant :

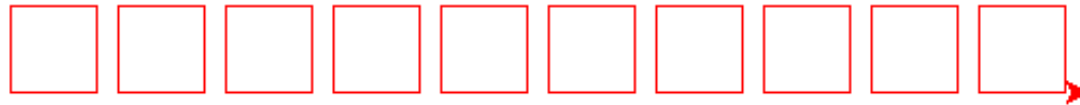
```
from turtle import* # importation du module turtle
color('red')
forward(100)
left(150)
color('blue')
forward(120)
left(120)
color('green')
forward(100)
right(90)
color('yellow')
forward(100)
```

2. Réaliser un script pour obtenir un carré. Enregistrer sous *carre.py*

3. Réaliser un script pour obtenir un triangle équilatéral. Enregistrer sous *triangle.py*

Applications:

1. Définissez une fonction *TraceCarre(taille, position, couleur)* qui permet de dessiner un carré. Cette fonction prend en arguments la taille du carré à dessiner, sa position sur l'écran et sa couleur.



2. Modifiez votre script pour que le choix de la couleur se fasse aléatoirement.



Applications:

1. Définissez une fonction *TraceTriangle(taille, position, couleur)* qui permet de dessiner un triangle équilatéral. Cette fonction prend en arguments la taille du triangle à dessiner, sa position sur l'écran et sa couleur.



2. Modifiez votre script pour que le choix de la couleur se fasse aléatoirement.



Pour les Lièvres!

Réaliser des scripts permettant de réaliser les figures suivantes :

