

Turtle et les boucles

I. Turtle

1) Observer le programme ci-dessous. Il faut savoir que tout ce qui est derrière # n'est pas pris en compte lors de l'exécution du programme : ce sont des commentaires, ils aident le lecteur à comprendre ce qui se passe.

Voici le squelette d'un programme avec Turtle. Le copier dans Thonny, sans trop se préoccuper de la signification des instructions.

```
from turtle import *

import traceback
try:
    setup(800,800) #ouvre une fenêtre de dimensions 800*800
    shape("turtle") # donne une forme de tortue au curseur

    #c'est à cet endroit que l'on écrit les instructions.
    mainloop()

except:
    traceback.print_exc()
```

Voici un exemple :

```
from turtle import *
import traceback
try:
    setup(800,800)
    shape("turtle")
    speed(3) # fixe la vitesse
    fd(50) #fd signifie forward : fait avancer la tortue de 50 pixels
    right(90) #tourne de 90° vers la droite
    fd(100)
    left(60)
    fd(50)
    pu() # pen up : lève le stylo, le tracé ne se fait plus
    goto(200,0) #déplace la tortue au point de coordonnées (200 ; 0)
    pd() # pen down : descend le stylo, le tracé se fait à nouveau
    bk(100)

    mainloop()

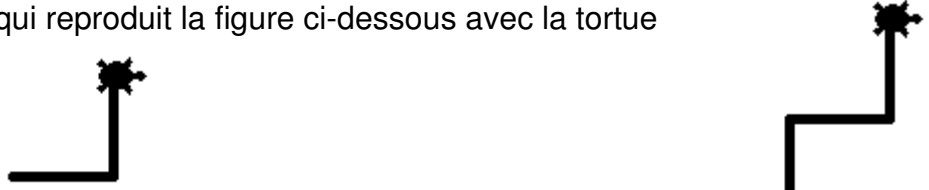
except:
    traceback.print_exc()
```

Bien sûr on peut utiliser les boucles, les if-else etc, dans les programmes avec la turtle.

2) A vous !

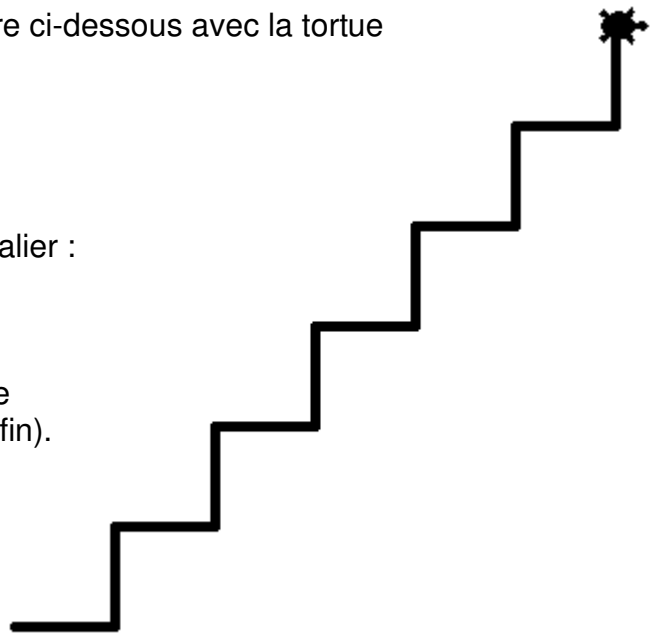
a) Ecrire un programme pour que la tortue avance de 350 pixels, puis tourne de 150° vers la gauche. Puis faire une boucle qui répète 12 fois cela.

b) Ecrire un programme qui reproduit la figure ci-dessous avec la tortue vers la droite à la fin.



c) Utiliser alors une boucle pour faire un escalier :

d) Faire un programme qui reproduit la figure ci-dessous (avec la tortue vers la droite à la fin).



Puis essayer d'obtenir :



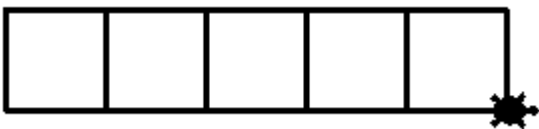
Il faudra partir du bord gauche de la fenêtre.

3) Avec des carrés et plus

a) Ecrire un programme qui dessine un carré.

b) Pourriez vous écrire ce programme avec seulement trois lignes d'instructions (à part le "squelette") ?

c) Modifier le programme pour obtenir la figure ci-dessous.



d) Décrire ce que fait le programme suivant, inséré dans le "squelette" :

```
for j in range(1,4):  
    for i in range(4):  
        fd(j*10)  
        left(90)  
        fd(j*10)
```

Pour cela, compléter :

Quand j vaut 1 on dessine un carré de côté et on va dans le coin

Quand j vaut

.....

e) Ecrire un programme qui réalise la figure suivante :

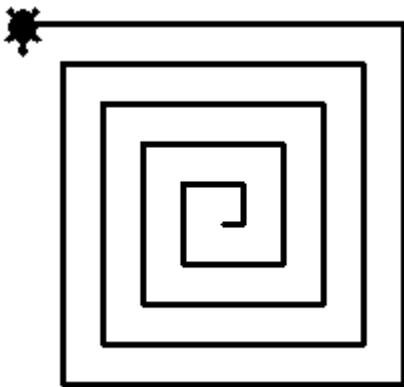


(C'est presque le même que le précédent...)

f) Ecrire un programme qui dessine un escalier, comme précédemment, mais la première marche fait 10 pixel, la deuxième 20 px, la troisième 30 etc....

Sauriez-vous le modifier, pour avoir des marches de 20, 30, 40, etc....

g) Ecrire un programme qui réalise la figure suivante :



Pour cela décrire le déplacement à partir du centre (le premier segment fait 10px, le second 20px etc...), puis écrire le programme correspondant.

II. Jeu "devine".

1) On donne le programme suivant :

```
from random import randrange
nbChoisi = eval(input("entrez un entier"))
nbOrdi = randrange(1,51)

if nbChoisi != nbOrdi :
    print("perdu")
else:
    print("gagné")
print("l'ordinateur avait choisi ", nbOrdi, " et vous ", nbChoisi)
```

A quoi correspond " != " ?

Que fait ce programme ?

2) Modifier ce programme pour que l'utilisateur puisse faire 10 essais.

3) Modifier le programme pour qu'à chaque essais le programme vous dise si le nombre que vous avez choisi est trop petit ou trop grand quand vous avez perdu.

3) Quand on a gagné, on ne veut pas continuer à faire des essais.
L'instruction break permet de sortir de la boucle. Essayer...

4) On voudrait savoir combien d'essais ont été nécessaires pour trouver. Compléter votre programme pour que ce soit affiché.

5) Une nouvelle boucle : la boucle while qui signifie "Tant que" en anglais.

Recopier le programme suivant et le tester.

```
a = 2
print ("a","|","a^3") # le "|" se trouve sous le 6
print("-----")
while a**3 <1200:
    print(a,"|",a**3)
    a = a + 1
```

Combien de fois passe-t-on dans la boucle while ?

Que vaut a au premier passage, au second, au troisième ?

Pourquoi ce programme s'arrête-t-il quand a vaut 10 ?

5) Compléter le programme suivant pour que l'utilisateur puisse jouer temps qu'il n'a pas gagné.

```
from random import randrange
nbChoisi = 100
nbEssais = 0
nbOrdi= randint(.....)
while nbOrdi != nbChoisi:
    nbChoisi = .....
    nbEssais = .....
    if nbChoisi < .....:
        print("Trop petit et c'est votre essai no", nbEssais)
    elif nbChoisi > .....: # elif signifie else if
        print("Trop grand et c'est votre esai no",nbEssais)
print ("Vous avez gagné en ", nbEssais," essais")
```

Répondre aux questions suivantes :

a) A la deuxième ligne, aurait-on pu mettre un autre nombre que 100 ?

b) Pourquoi à la dernière ligne est-on sûr que le joueur a gagné ?