

13.4

Amusons-nous avec turtle


MATHS 2NDE 7 - JB DUTHOIT

13.4.1 Un peu de doc !

Pour importer la bibliothèque :

```
from turtle import *
```

Fonctionnalités principales :

- Il est possible de commander le paramétrage du crayon par
 - `down()` qui abaisse le stylo
 - `up()` qui relève le stylo
 - `pensize(width)` qui change l'épaisseur du trait
 - `pencolor(color)` qui change la couleur ("red", "green", "blue"... ou un triplet de paramètres (r, g, b))
- On déplace la tortue avec
 - `forward(length)` qui avance d'un nombre de pas donné
 - `backward(length)` qui recule
 - `right(angle)` qui tourne vers la droite d'un angle donné (en degrés)
 - `left(angle)` qui tourne vers la gauche.
- On peut également déplacer la tortue à un point donné ou modifier son orientation avec
 - `goto(x,y)` qui déplace la tortue jusqu'au point (x,y)
- Au départ, la tortue est en (0,0), orientée à 0 degré.
- La fenêtre par défaut est 950 pixels de large et 800 pixels de haut.  Le point (0,0) est au centre de l'écran.
- Il est possible de modifier cette fenêtre avec :
 - `setup(width, height)` qui définit les tailles en pixels de la largeur et hauteur de la fenêtre
 - `clear()` qui efface tout ce qui a été tracé ou écrit dans la fenêtre.

13.4.2 Exercices

Le code suivant crée une fonction `avance(n)` qui va déplacer la tortue de n pas :

```
def avance(n):
    forward(n)
```

Exercice 13.65

Ecrire une fonction `carre(n)` qui crée un carré de côté n et remet le curseur dans sa position initiale.

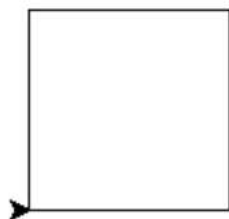


FIGURE 13.1 – Figure obtenue avec $n=100$

Exercice 13.66

Réaliser ceci :

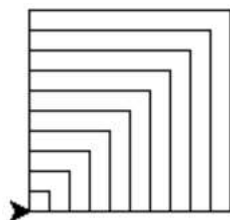
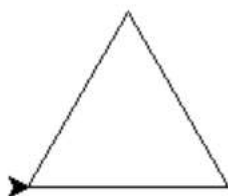


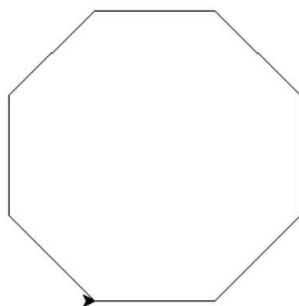
FIGURE 13.2 – Les 11 carrés ont pour côtés 10,20,30 ..etc..

Exercice 13.67

Ecrire une fonction `triangle(n)` qui crée un triangle équilatéral de côté n et remet le curseur dans sa position initiale.

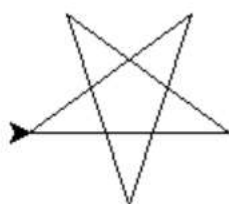
**Exercice 13.68**

Ecrire une fonction `octogone(n)` qui crée un octogone régulier de côté n et remet le curseur dans sa position initiale.



Exercice 13.69

Ecrire une fonction `etoile(n)` qui crée une jolie étoile à 5 branches de côté `n`.



Exercice 13.70

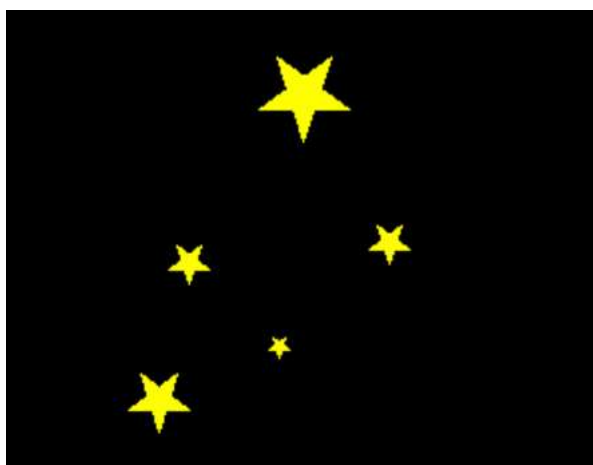
Ecrire une fonction `etoile_j(n)` qui crée une jolie étoile de couleur jaune à 5 branches de côté `n`.



☛ Utiliser pour cela `color('yellow')` ainsi que `begin_fill()` et `end_fill()`.

Exercice 13.71

Créer un ciel étoilé :-)



- Utiliser pour cela `goto(x,y)` ainsi que `bgcolor("black")` pour la couleur de fond.