

Informatique SV L2

TP 2

Python intègre un module «Turtle» permettant de dessiner, comme avec un crayon. Les fonctions principales sont

- `forward(distance)` : Déplace le crayon
- `up()` : Lève le crayon (on écrit pas quand on se déplace)
- `down()` : Pose le crayon (on écrit quand on se déplace)
- `right(angle)` : change la direction de déplacement d'un certain angle vers la droite
- `left(angle)` : change la direction de déplacement d'un certain angle vers la gauche
- `goto(x,y)` : déplace le crayon à un emplacement précis de la fenêtre
- `color()` : définit la couleur du crayon
- `speed(vitesse)` : Définit la vitesse à laquelle on dessine : Le paramètre vitesse peut être : 'fastest', 'fast', 'normal', 'slow', 'slowest'

1 Dessiner dix carrés

1.1 Fonction carre

La fonction ci-dessous dessine un carré :

```
from turtle import *
def carre(taille, couleur) :
    "fonction qui dessine un carré" #documente la fonction
    color(couleur)
    c = 0
    while c < 4 :
        forward(taille)
        right(90)
        c = c + 1
```

Ecrire cette fonction dans fichier qu'on nommera *dessins.py*

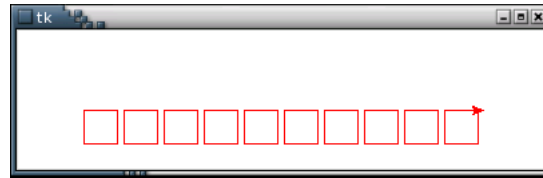
1.2 Programme principal

Dans un nouveau fichier, écrire le programme suivant permettant de dessiner 2 carrés en appelant la fonction *carre* définit précédemment

```
from dessins import * # importe les fonctions du fichier dessins.py
up() # relever le crayon
goto(-150, 50) # va en haut à gauche
# dessiner deux carrés rouges, alignés
down() # abaisser le crayon
carre(25, 'red') # tracer un carré
up() # relever le crayon
forward(30) # avancer + loin
down() # abaisser le crayon
carre(25, 'red') # tracer un carré
up() # relever le crayon
a = input() # attendre
```

Tester le programme.

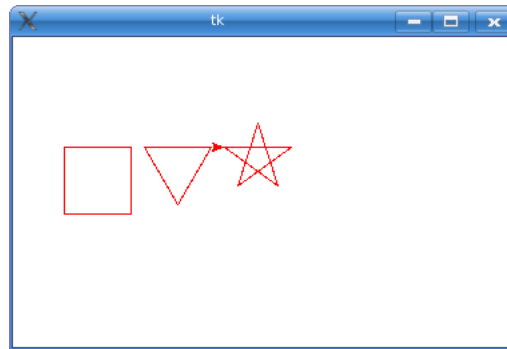
Modifier le programme pour qu'il affiche 10 carrés alignés en utilisant un boucle. On doit obtenir le résultat suivant :



2 Carrés, triangles, étoiles...

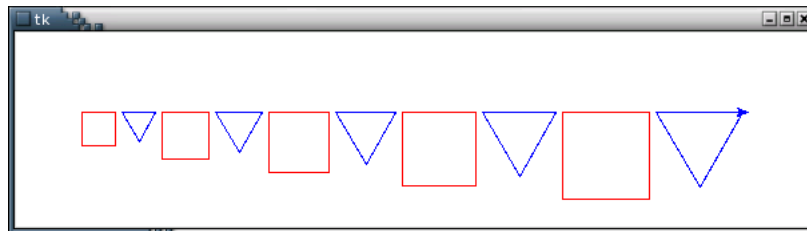
2.1 Complement du modules dessins

Dans le fichier *dessins.py* ajouter 2 fonctions `triangle(taille, couleur)` et `etoile5(taille, couleur)` qui dessinent un triangle équilatéral et une étoile à 5 branches. Dans un autre fichier, écrire un programme pour obtenir l'affichage suivant :



2.2 Varier la taille

A l'aide d'un programme qui fera appel aux fonctions `carre` et `étoiles` à plusieurs reprises, avec des arguments variés pour dessiner une série de carrés et de triangles :



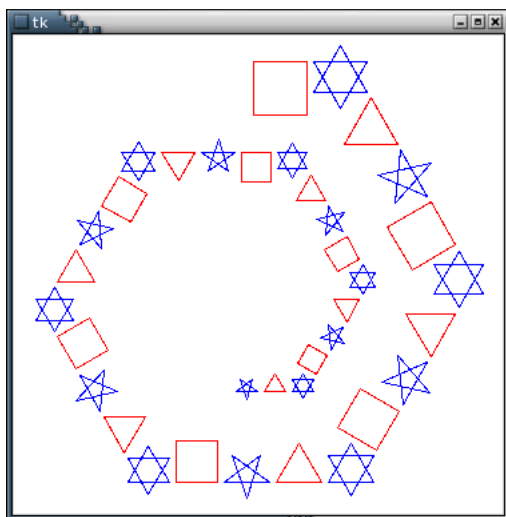
Écrire un programme qui dessine 9 étoiles de différentes taille de manière à obtenir le résultat suivant :



2.3 Étoiles à 6 branches (facultatif)

Ajoutez au module *dessins* une fonction `etoile6()` capable de dessiner une étoile à 6 branches, elle-même constituée de deux triangles équilatéraux imbriqués. Cette nouvelle fonction devra faire appel à la fonction `triangle()` et `triangle_retourne()` qu'on rajoutera au module *dessins*.

Votre programme principal dessinera une série de ces $\frac{2}{2}$ étoiles :



Merci à Gérard Swinnen pour le document « Apprendre à programmer avec Python » (sous la GNU Free Documentation License, version 1.1), dont est inspiré ce TP.