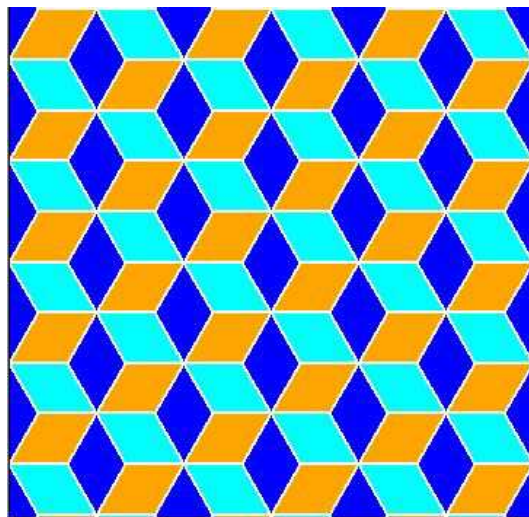


Réalisation d'un pavage avec le mode Turtle de Python

Partie du programme concernée	Langages et programmation Constructions élémentaires. Mise au point de programmes.
Objectif	Appliquer les connaissances acquises sur les Boucles, Fonctions et Listes en Python pour construire un pavage
Pré-requis	Boucle for, Fonctions, Listes, Fonctions de base de la bibliothèque turtle
Durée	2 heures

L'objectif de cette séance est de réaliser l'image suivante en utilisant la tortue sous Python.

Les difficultés attendues lors de cette séance sont autant d'ordre algorithmique que géométrique.

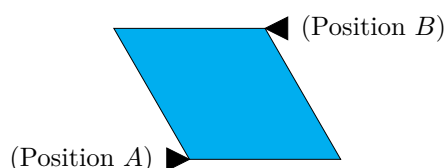


I. Récupération du fichier original

1. Le fichier *ActiviteEleve* sur lequel vous allez travailler se trouve dans le répertoire Devoir/... Sélectionnez le fichier, coupez-le (ctrl+X) et collez-le (ctrl+V) dans votre répertoire NSI.
2. Ouvrez le fichier depuis un environnement Python.

II. Construction d'un losange

Voici l'un des losanges formant le pavage que nous devons réaliser :



On précise que les côtés du losange mesure 40 unités, que ses deux angles aigus mesures 60° et que ses deux angles obtus mesurent 120° . On suppose également qu'au départ la tortue est en position *A* (orientée vers la droite comme indiqué sur la figure).

1. Saisir les quatre instructions permettant de déplacer la tortue de la position *A* à la position *B* (en respectant bien l'orientation finale).

Tester votre programme.

```
fd(40)
lt(120)
fd(40)
lt(60)
```

2. A l'aide d'une boucle *for*, achever la construction du losange.

Tester votre programme.

```
for i in range(2):
    fd(40)
    lt(120)
    fd(40)
    lt(60)
```

3. Saisir les instructions permettant de colorer le bord de votre losange en blanc et l'intérieur en cyan.

Tester votre programme.

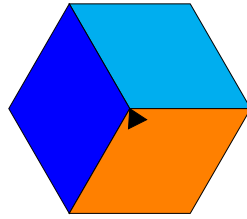
```
color('white', 'cyan')
begin_fill()
for i in range(2):
    fd(40)
    lt(120)
    fd(40)
    lt(60)
end_fill()
```

4. Récupérer les instructions saisies précédemment afin de créer une fonction **TracerLosange(CouleurBord, CouleurRemplissage)** qui permet de dessiner un losange dont les couleurs sont choisies en paramètre.

Tester votre fonction.

```
def TracerLosange(CouleurBord, CouleurRemplissage):
    color(CouleurBord, CouleurRemplissage)
    begin_fill()
    for i in range(2):
        fd(40)
        lt(120)
        fd(40)
        lt(60)
    end_fill()
```

III. Construction d'un motif



1. Une fois le premier losange (cyan) tracé avec la fonction **TracerLosange**, quelle instruction doit-on saisir pour orienter correctement la tortue afin de pouvoir commencer le tracé du losange de gauche (bleu) avec la fonction **TracerLosange** à nouveau ?

Tester l'exécution successive de **TracerLosange('white','cyan')** et de votre instruction de rotation.

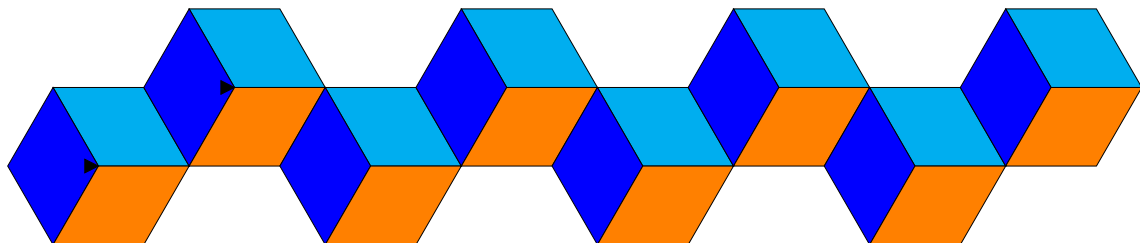
```
TracerLosange('white', 'cyan')  
lt(120)
```

2. A l'aide d'une boucle *for* et de votre fonction **TracerLosange**, construire une fonction **TracerMotif(ListeCouleurs)** ayant en paramètre une liste de quatre couleurs (la première désignant la couleur du bord des losanges et les trois autres celles de remplissage) permettant de construire le motif précédent.

Tester votre fonction avec la liste de couleurs **Couleurs=['white','cyan','blue','orange']**.

```
def TracerMotif(ListeCouleurs):  
    for i in range(3):  
        TracerLosange(ListeCouleurs[0], ListeCouleurs[1+i])  
    lt(120)
```

IV. Construction d'une ligne de motifs



1. Une fois le premier motif (le plus à gauche) tracé avec la fonction **TracerMotif**, quelles instructions doit-on saisir pour déplacer, sans tracer, la tortue au centre du prochain motif ?

Tester l'exécution successive de **TracerMotif(Couleurs)** et de vos instructions de déplacement.

```
TracerMotif(Couleurs)  
penup()  
fd(40)  
lt(60)  
fd(40)  
rt(60)  
pendown()
```

2. Une fois le second motif tracé, quelles instructions doit-on saisir pour déplacer, sans tracer, la tortue au centre du prochain motif ?

Tester les instructions permettant de dessiner les deux premiers motifs et de placer la tortue dans la position permettant de tracer le troisième motif avec la fonction **TracerMotif**.

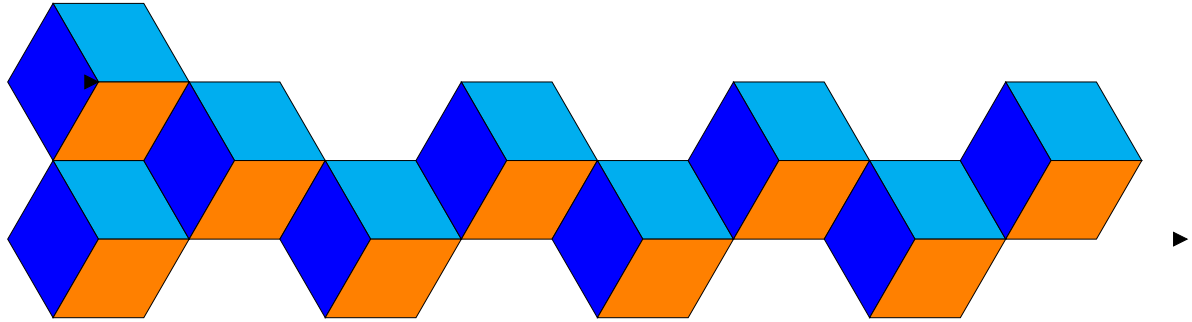
```
TracerMotif(Couleurs)
penup()
fd(40)
lt(60)
fd(40)
rt(60)
pendown()
TracerMotif(Couleurs)
penup()
fd(40)
rt(60)
fd(40)
lt(60)
pendown()
```

3. A l'aide d'une boucle *for* et de votre fonction **TracerMotif**, construire une fonction **TracerLigne(ListeCouleurs)** ayant en paramètre une liste de quatre couleurs permettant de construire la ligne de motifs précédente.

Tester votre fonction avec la même liste de couleurs.

```
def TracerLigne(ListeCouleurs):
    for i in range(4):
        TracerMotif(ListeCouleurs)
        penup()
        fd(40)
        lt(60)
        fd(40)
        rt(60)
        pendown()
        TracerMotif(ListeCouleurs)
        penup()
        fd(40)
        rt(60)
        fd(40)
        lt(60)
        pendown()
```

V. Construction du pavage



1. Une fois la première ligne de motifs (la plus basse) tracée avec la fonction **TracerLigne**, quelles instructions doit-on saisir pour déplacer, sans tracer, la tortue au point de départ de la prochaine ligne ?

Tester l'exécution successive de **TracerLigne(Couleurs)** et de vos instructions de déplacement.

```
TracerLigne(Couleurs)
penup()
bk(520)
lt(60)
fd(80)
rt(60)
pendown()
```

2. A l'aide d'une boucle *for* et de votre fonction **TracerLigne**, construire une fonction **TracerPavage(ListeCouleurs)** ayant en paramètre une liste de quatre couleurs permettant de construire le pavage souhaité.

Tester votre fonction avec la liste de couleurs de votre choix.

```
def TracerPavage(ListeCouleurs):
    penup()
    goto(-200,-200)
    pendown()
    for i in range(7):
        TracerLigne(ListeCouleurs)
    penup()
    bk(520)
    lt(60)
    fd(80)
    rt(60)
    pendown()
    hideturtle()
```

VI. Rappel des commandes de la tortue

Fonction	Effet
setup(w,h)	Ouvre une fenêtre de dessin de largeur w et de hauteur h
showturtle()	Montre la tortue
hideturtle()	Cache la tortue
reset()	Efface l'écran, remet la tortue au centre et réinitialise ses paramètres
fd(n)	Avance de n unités
bk(n)	Reculé de n unités
rt(n)	Tourne à droite de n degrés
lt(n)	Tourne à gauche de n degrés
goto(x,y)	Déplace la tortue au point de coordonnées (x,y)
penup()	Lève le stylo
pendown()	Baisse le stylo
color('...', '...')	Change la couleur (en anglais) de la tortue (bien mettre la couleur entre guillemets, par exemple : 'red'). La première couleur est celle du bord, la seconde celle de remplissage.
begin_fill()	Commence le remplissage couleur
end_fill()	Termine le remplissage couleur