



Programmation avec Tynker

Activité 1 : Construire une rosace

Cycle 4, niveau 1, durée : environ 30 min
Thèmes : algorithmes et programmation, transformations du plan
Objectifs : boucles imbriquées répéter, déplacement, stylo, modifier un programme.

Allumer la tablette et ouvrir l'application Tynker.

Etape 1 : Construire un cercle

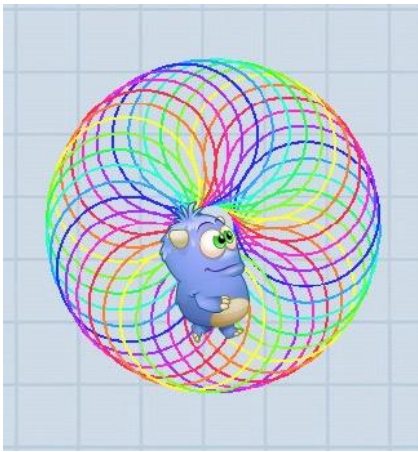
Voici un programme pour tracer un icosagone, Modifier les paramètres du script afin d'obtenir la représentation d'un cercle.

```
on start
clear
pen down
set pen color to #941100
repeat 20
  move 30 pixels
  turn 18 degrees
```

Etape 2 : Construire un ensemble de cercles

Compléter les instructions suivantes afin d'obtenir la figure ci-dessous, puis programmer et exécuter votre programme.

```
on start
clear
pen down
repeat 36
  turn 10 degrees
  set pen shade to 60
  change pen color by 20
  repeat 36
    [ ]
```



Aide étape 2 : Construire une rosace

Si vous ne savez pas répondre aux questions, recopier le programme suivant et observer le tracé.

```
rosace
Code Stage
Motion
move 10 pixels
turn 15 degrees
turn 15 degrees
point in direction 90 degrees
set angle to 0 degrees
point towards select actor
go to x: 0 y: 0
go to select actor
glide 1 secs to x: 10 y: 10
on start
clear
go to x: -134 y: 5
repeat 36
  pen up
  turn 10 degrees
  move 3 pixels
  set pen shade to 60
  change pen color by 20
  repeat 36
    pen down
    turn 10 degrees
    move 10 pixels
```

Allumer la tablette et ouvrir l'application Tynker.

1) Voici deux programmes et deux frises.

Quel programme permet d'obtenir la frise 1 ?

Quel programme permet d'obtenir la frise 2 ?

Programme 1 :

```

on start
  point in direction 90 degrees
  set size to 20 %
  go to x: -600 y: 50
  clear
  repeat 10
    pen down
    repeat 4
      move 80 pixels
      turn 90 degrees
      wait 0.2 secs
    pen up
    move 80 pixels
  
```

Programme 2 :

```

on start
  point in direction 90 degrees
  set size to 20 %
  go to x: -600 y: 50
  clear
  repeat 7
    pen down
    repeat 4
      move 80 pixels
      turn 90 degrees
      wait 0.2 secs
    pen up
    move 160 pixels
  
```

Frise 1 :

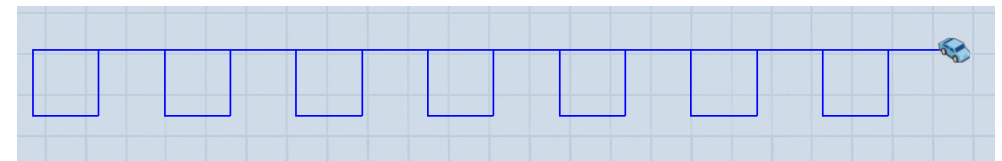


Frise 2 :



Programmer ces deux frises pour vérifier vos réponses.

2) Modifier le programme afin d'obtenir la frise suivante :



Aide question 2 : exécuter le programme suivant et vérifier que l'on obtient bien la frise voulue.

```

on start
  point in direction 90 degrees
  set size to 60 %
  go to x: -600 y: 50
  clear
  repeat 7
    pen down
    repeat 4
      move 80 pixels
      turn 90 degrees
      wait 0.2 secs
    move 160 pixels
    pen up
  
```

Ouvrir l'application Tynker et modifier le lutin (*properties, add costume, bibliothèque spooky, choisi l'araignée spider*).

1. Une araignée tisse sa toile de la manière suivante :

Etape 1 : elle va au sommet du carré,
Etape 2 : elle trace un carré de longueur « côté »,
Etape 3 : elle revient au centre.

Programmer avec Tynker la fonction « **carré de longueur côté** » ci-contre et repérer ses 3 étapes sur le programme.

```

carré côté
pen up
move côté / 2 pixels
turn 90 degrees
move côté / 2 pixels
pen down
repeat 4
  turn 90 degrees
  move côté pixels
pen up
repeat 10
  move côté / 2 pixels
  turn 90 degrees
turn 90 degrees
  
```

2. Voici le **programme principal** :

- a) Quelle est la longueur du côté du carré initial ?
- b) Combien de rotations de carrés sont effectuées dans le programme ?
- c) Quel est l'angle de la rotation ?
- d) Justifier que le carré est invariant par 45 rotations d'angle 2° .

3. Recopier et exécuter le programme. Quelle est la forme de la toile tissée par l'araignée ?

```

on start
set size to 50 %
point in direction 90 degrees
go to x: 0 y: 0
clear
repeat 45
  change pen color by 10
  carré 200
  turn 2 degrees
  
```

4. Modifier tour à tour les 3 paramètres et observer les modifications sur la toile d'araignée :
 - a) la longueur du carré
 - b) le nombre de rotations
 - c) l'angle de rotation

Aide question 3 : Construire la toile

Voici la forme tissée par l'araignée que l'on doit obtenir :



Programmation avec Tynker
Activité 4 : Un pavage de spirales

Cycle 4, niveau 4, durée : environ 50 min

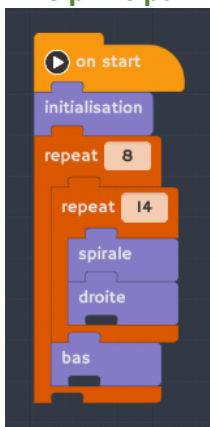
Thème : Algorithme et programmation, pavage.

Objectifs : fonctions (bloc), variable, boucle répéter, déplacement, stylo, écrire un programme.

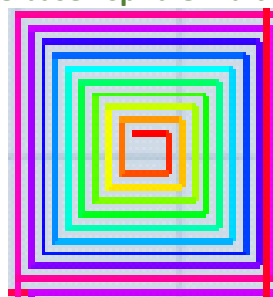
Allumer la tablette et ouvrir l'application Tynker.

Voici un programme principal, composé de 4 fonctions, qui permet de tracer un pavage de 112 spirales.

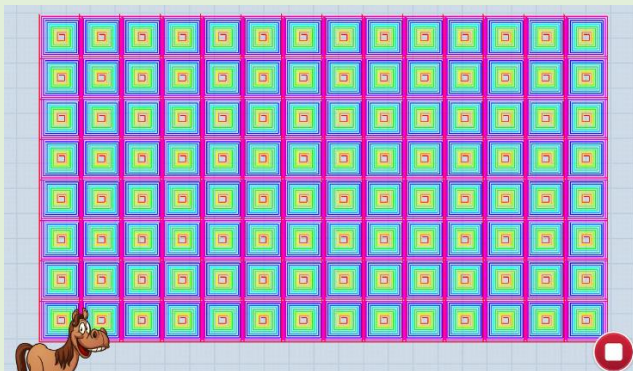
Programme principal :



Motif de base : spirale multicolore :



Pavage final à obtenir sur l'écran :



Ecrire le programme des 4 fonctions, décrites ci-dessous, puis exécuter le programme principal pour voir si le pavage est réussi !

Fonction « Initialisation » :

_ Positionner le lutin au départ à $x = -500$ et $y = 300$.
_ Effacer les précédentes constructions.

Fonction spirale :

Utiliser une variable pas afin de construire la spirale. Cette variable sera incrémentée de 2 à chaque construction de côté.

Fonction « droite » :


Cette fonction sert à se positionner au centre d'une nouvelle spirale afin de continuer la ligne de 14 spirales.

Fonction « bas » :

Cette fonction sert à revenir à gauche de l'écran, au centre d'une nouvelle spirale, afin de recommencer une nouvelle ligne de spirales.

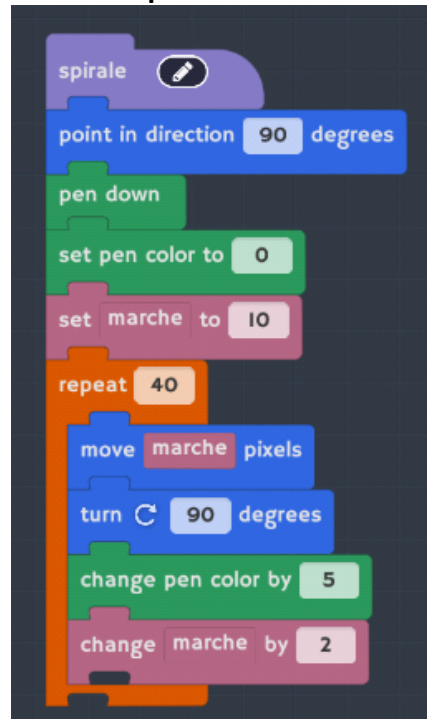
Aide : programmer les fonctions suivantes et exécuter le programme principal, le pavage devrait apparaître de toutes les couleurs !

Fonction « Initialisation » :



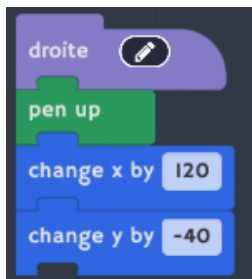
```
initialisation
go to x: -500 y: 300
clear
```

Fonction spirale :



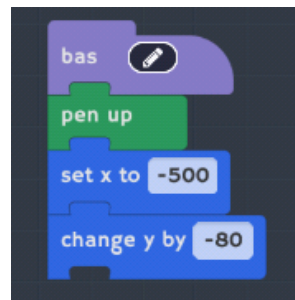
```
spirale
point in direction 90 degrees
pen down
set pen color to 0
set marche to 10
repeat 40
  move marche pixels
  turn 90 degrees
  change pen color by 5
  change marche by 2
```

Fonction « droite » :



```
droite
pen up
change x by 120
change y by -40
```

Fonction « bas » :



```
bas
pen up
set x to -500
change y by -80
```