Nom_____ Date_____

La géométrie
Unité 1, Fiche 7a

Codage et symétrie de rotation

Utilisons le codage pour modéliser la symétrie de rotation des figures à 2D.

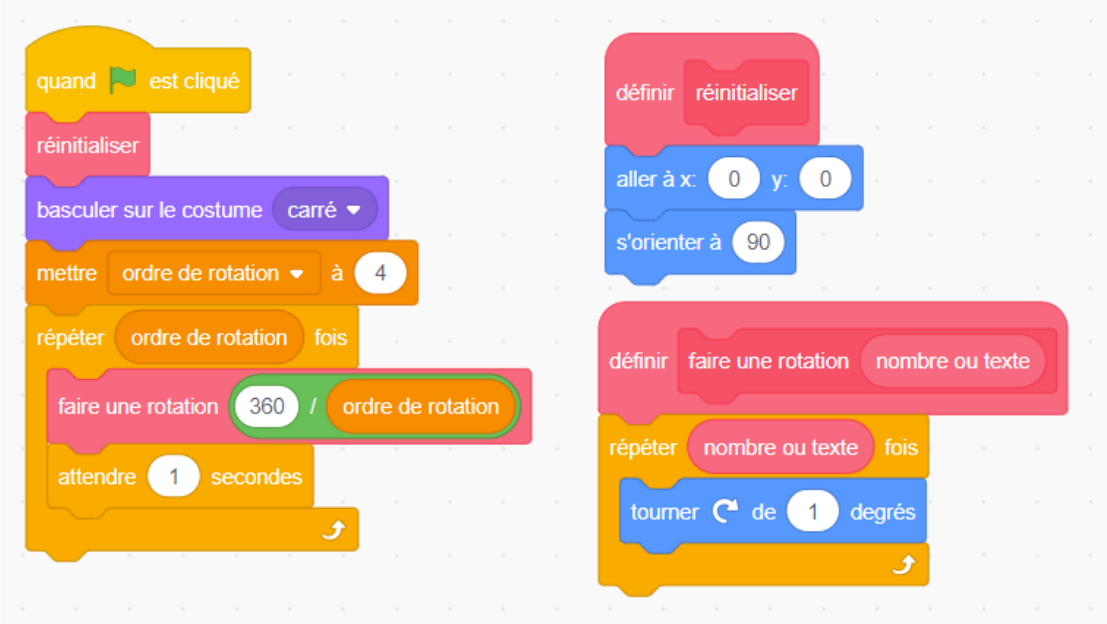
1. Clique sur le lien pour accéder à Scratch : L'ordre de rotation :

   https://scratch.mit.edu/projects/926509772/editor/

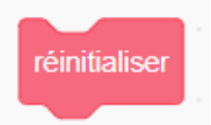
   - Clique sur le drapeau vert pour lancer l'application.

     Tu verras qu'en un tour complet, le carré est tourné 4 fois puisque son ordre de rotation est 4. À chaque fois, il ressemble au carré d'origine.

2. Examinons le code afin d'en comprendre le fonctionnement.
   Nous allons ensuite modifier le code pour modéliser l'ordre de rotation d'un triangle, d'un pentagone et d'un hexagone.



   - Le bloc de **réinitialisation** a été créé pour s'assurer que la figure commence au centre de la scène et qu'elle est orientée dans la bonne direction avant de faire une rotation.

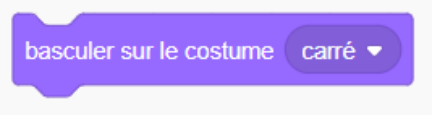


Nom_____ Date_____

La géométrie
Unité 1, Fiche 7b

Codage et symétrie de rotation (suite)

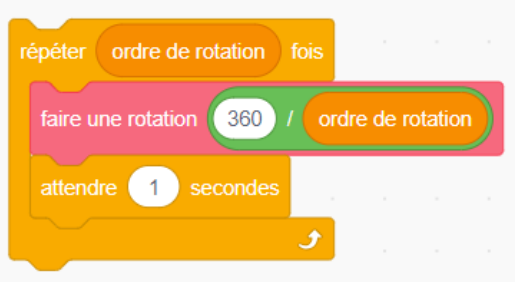
- Des « costumes » ont été préparés pour un triangle, un carré, un pentagone et un hexagone. Pour modéliser l'ordre de rotation d'un carré, le costume carré est sélectionné. Tu peux cliquer sur l'onglet **Costumes** pour voir les autres figures à 2D qui ont été préparées.



- Une variable appelée « ordre de rotation » contient le nombre de rotations nécessaires pour modéliser la symétrie de rotation et ramener la figure à sa position de départ. En tant que programmeur ou codeur, tu devras modifier cette valeur en fonction de la figure que tu utilises. Comme nous commençons par un carré, nous utilisons la valeur 4.

- Le bloc de **répétition** contient le code qui fera subir une rotation du carré 4 fois, puisque l'ordre de rotation est actuellement fixé à 4.

Une boucle est une répétition d'instructions utilisées dans un code. Dans Scratch, le bloc de **répétition** représente la boucle.

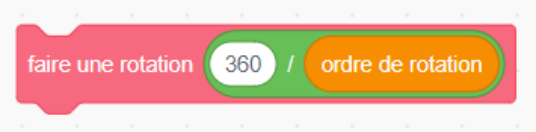


Nom_____ Date_____

La géométrie
Unité 1, Fiche 7c

Codage et symétrie de rotation (suite)

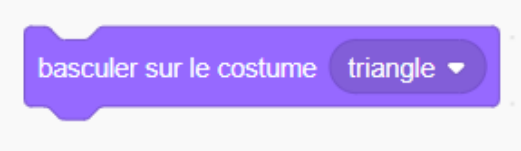
- Le bloc de **rotation** a été créé pour que la rotation se fasse progressivement, comme une animation. Pour calculer l'angle de rotation, il faut diviser 360° par l'ordre de rotation. Ainsi, pour le carré, chaque rotation sera de 360° ÷ 4 = 90°.



- Le bloc d'**attente** est utilisé pour mettre le bloc en pause pendant 1 seconde avant d'effectuer la rotation suivante. Tu peux modifier cette valeur si tu souhaites une pause plus courte ou plus longue.

3. Maintenant que nous avons examiné le code, modifions-le pour qu'il modélise la symétrie de rotation pour d'autres polygones. Nous commencerons par un triangle équilatéral.

   - Utilise le menu déroulant pour transformer le costume en triangle.



   - Un triangle a un ordre de rotation de 3, il faut donc ajuster la valeur de la variable « ordre de rotation » :

Nom_____ Date_____

La géométrie
Unité 1, Fiche 7d

Codage et symétrie de rotation (suite)

Voilà, c'est fait ! Clique sur le drapeau vert pour lancer l'application.
L'application exécute-t-elle trois rotations ?
Le triangle a-t-il la même apparence à chaque fois ?
Si ce n'est pas le cas, lis attentivement le code et les instructions pour déboguer.

4. Vas-y et modifie le code pour modéliser la symétrie de rotation d'un pentagone et d'un hexagone.