

Coding and Rotation Symmetry

Let's use coding to model rotation symmetry of 2-D shapes.

1. Click the link to access Scratch: Order of Rotation:

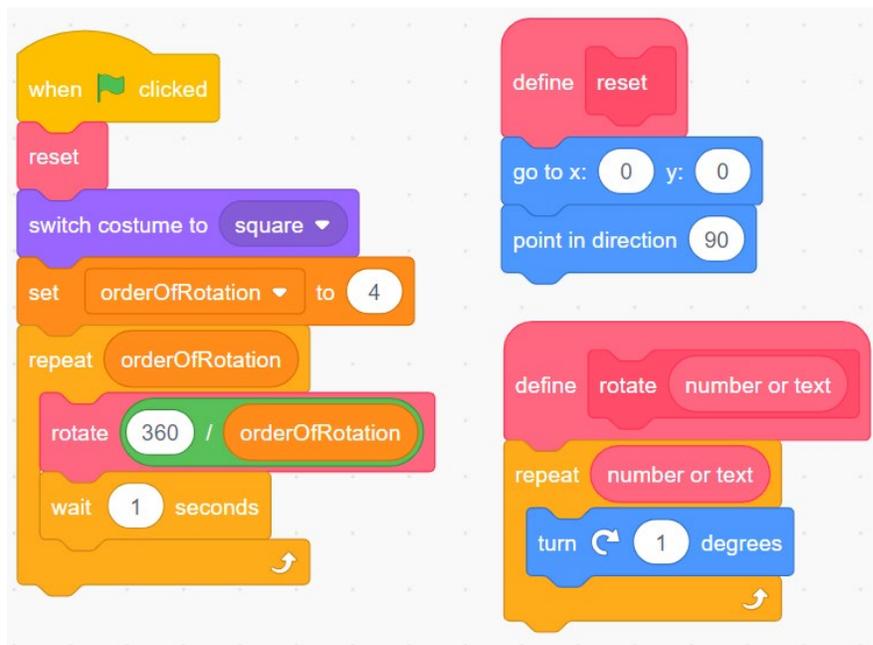
<https://scratch.mit.edu/projects/879134601/editor/>

- Click the green flag to run the application.

You will see that in one complete turn, the square is rotated 4 times since its order of rotation is 4. Each time, it looks like the original square.

2. Let's examine the code so that we understand how it works.

We will then alter the code to model order of rotation for a triangle, pentagon, and hexagon.

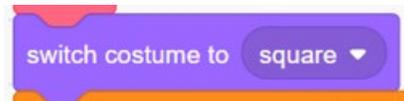


- The **reset** block has been created to ensure the shape starts in the centre of the stage and faces the correct direction before rotating.



Coding and Rotation Symmetry (cont'd)

- “Costumes” have been prepared for a triangle, square, pentagon and hexagon. To model the order of rotation for a square, the square costume is selected. You can click on the **Costumes** tab to see the other 2-D shapes that have been prepared.



- A variable called **orderOfRotation** holds the number of rotations required to model rotation symmetry and bring the shape back to its starting position. As the programmer or coder, you will need to change this value depending on the shape you are using. Since we are starting with a square, we use the value 4.

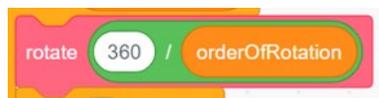


- The **repeat** block contains code that will rotate the square 4 times, since the orderOfRotation is currently set to 4.

A loop is a repetition of instructions used in code. In Scratch, the **repeat** block represents the loop.

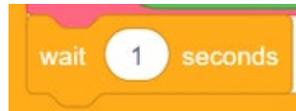


- The **rotate** block was created so that the rotation happens gradually, like an animation. To calculate the angle of rotation, we need to divide 360° by the order of rotation. So, for the square, each rotation will be $360^\circ \div 4 = 90^\circ$.



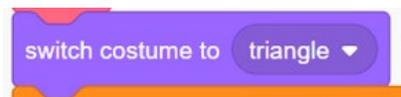
Coding and Rotation Symmetry (cont'd)

- The **wait** block is used to pause the block for 1 second before doing the next rotation. You can alter this value if you wish to pause it for a shorter or longer time.



3. Now that we have examined the code, let's alter the code so that it will model rotation symmetry for other polygons. We will start with an equilateral triangle.

- Use the pull-down menu to change the costume to a triangle.



- A triangle has order of rotation of 3, so adjust the value for the orderOfRotation variable:



That's it! Click on the green flag to run the application.

Does it rotate 3 times?

Does the triangle look the same each time?

If not, look through the code and instructions carefully to debug.

4. Go ahead and alter the code to model rotation symmetry for a pentagon and a hexagon.