

Lesson 4.1: Spirals Come to Life

Objectives

In this lesson, students will:

- ❖ Learn to differentiate between an algorithm to draw a circle and a spiral.
- ❖ Practice adapting comments to match the code
- ❖ Practice programmatic concepts of loops, variables, procedures (blocks), random numbers, and comments

Agenda

1. From Circle to Spiral	10 mins
2. Student Activity: Spirals Come to Life - Exercise 1	15 mins
3. Student Activity: Spirals Come to Life - Exercise 2	20 mins
4. Wrap Up and Reflections	5 mins

Preparation

- Projector for demonstrations
- Student activity worksheet printouts (one per students or one per student pair)

Resources & Links

- Solution project:
<https://scratch.mit.edu/projects/281795021/>
- Starter project:
<https://scratch.mit.edu/projects/281795139/>

1. From Circle to Spiral



In this lesson students will draw spirals creating a beautiful canvas.

Open the Spirals Solution project and click on the pencil sprite called **circle**.

<https://scratch.mit.edu/projects/281795021>

Go into fullscreen mode for the stage (upper right corner, click on ).

Display your screen and engage students in an interactive demonstration and instruction:

Click on the letter **c**. This draws a circle. Display the script for the sprite called **circle**.

Draw students' attention to the repeat loop in the procedure called circle. Point out that to draw a circle, the code draws a line, turns a little, draws a line and turns a little. This is done many times in the repeat loop. Draw students' attention to the variable named **distance**. This variable holds the number of steps we move each time before turning.

What do you think would happen if we change the number stored in distance to a larger number? A smaller number?

Change the value in distance to 12 or 14. Clear the stage by clicking the space bar and then click the letter **c** again to observe what happens when the distance is longer. The circle is bigger.

Now click on the green flag and click somewhere on the stage. A spiral will be drawn where the mouse is clicked. Click on a different location on the stage to demonstrate how wherever you click the mouse, a spiral is drawn. Clicking on the space bar clears the stage.



Tell students to draw a circle in their design journal and then a spiral and observe what they do differently to draw a circle versus a spiral.

Prompt students to share their observations.

If we look at the algorithm for each:

The algorithm to draw a **circle** is:

Draw a small line,

Turn a little,

Draw a small line,

Turn a little,.....

To draw a spiral we have to move outward a little at a time to create the spiral effect. The algorithm is:

Draw a small line,
 Turn a little,
 Draw a bigger line,
 Turn a little,
 Draw a bigger line,
 Turn a little,

We can conclude that a spiral winds continuously in a gradually widening curve.

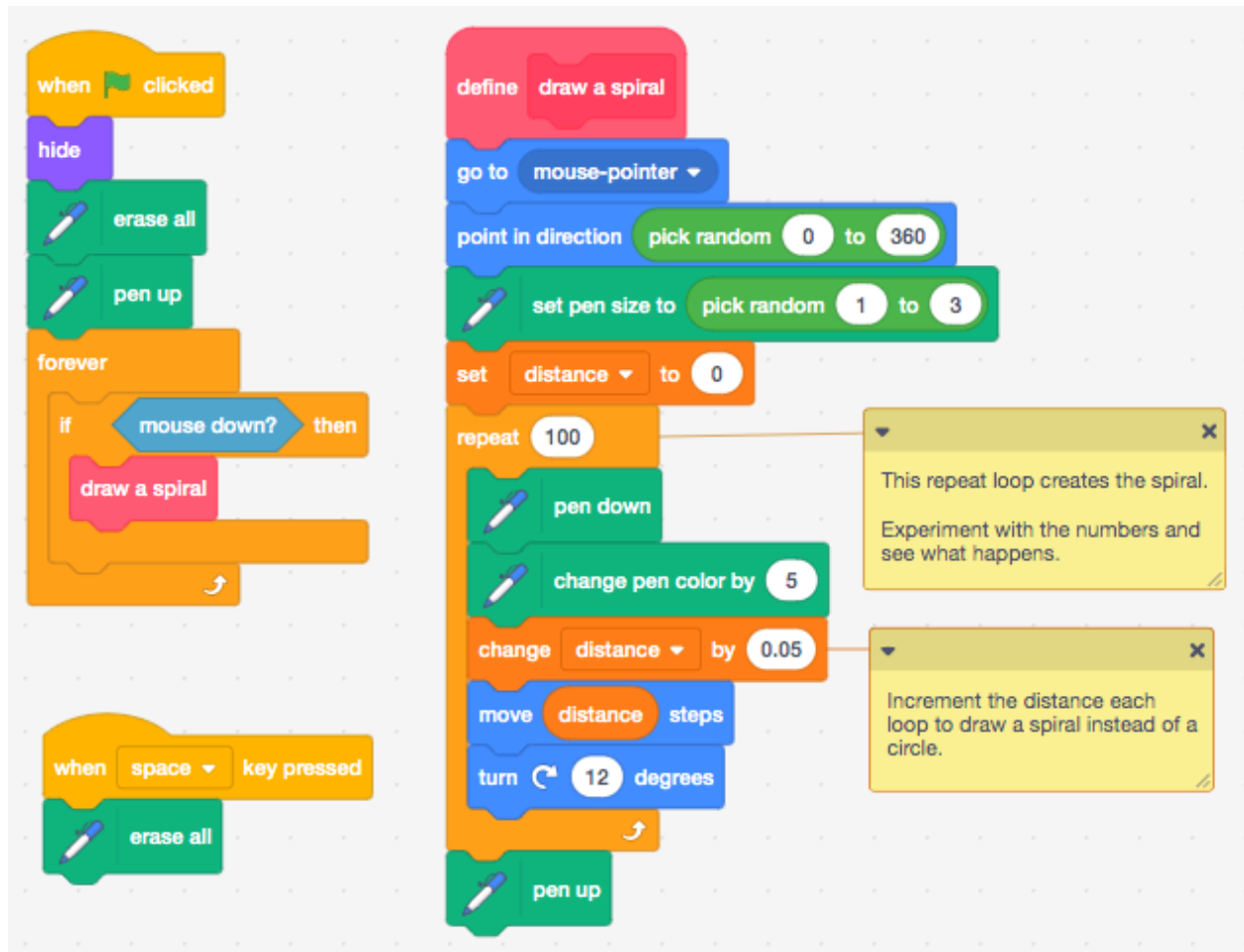
2. Student Activity: Spirals Come to Life - Exercise 1



In this activity students are given a starter project. Students will convert code that draws a circle to code that draws a spiral. They add some code to the starter project to draw a spiral wherever the mouse is clicked on the stage. Students will also practice giving credit for the starter project in the project page.

Explain the activity for exercise 1 and distribute the activity worksheet.

Review the solution with students if necessary:



3. Student Activity: Spirals Come to Life - Exercise 2



In this activity students are instructed to be creative by changing the look of the spirals. They will also change the project to draw spirals at random locations instead of at the mouse click location.

They are instructed to observe the code comment and change any comments necessary to match the code. And finally, students will share their project. If you have particular instructions for project sharing, you will want to let students know.

Explain the activity to students and distribute the activity worksheet.

[Review the solution with students:](#)

The code to change the size and width of the spirals is in the solution project.

To draw spirals at random locations, the script for **When green flag clicked** and the **go to** block is changed to the following:



The comment for the **go to** block also has changed.

4. Wrap Up and Reflections

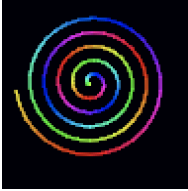



Reflection Points:


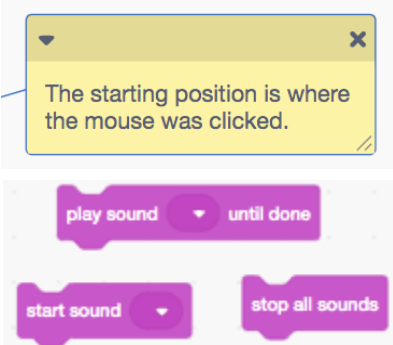
- What did you like about today’s activity?
- What was challenging about today’s activity?
- What did you notice when you increased the distance incrementally?
- What is the difference between drawing a circle and a spiral?

Student Activity: Spirals Come to Life

Exercise 1:

What to do:	Using/Details:
 <p>Remix and save</p>	<p>281795139</p>
<p>Give credit to the starter project</p>	<p>Notes and Credits</p> <p><i>How did you make this project? Did you use ideas, scripts or artwork from other people? Thank them here.</i></p>
<p>Use these blocks to complete the main script so that the sprite draws a spiral when the mouse is down</p>	
<p>Run the program.</p> <p>The spiral procedure draws a circle and not a spiral. Explore changing it so it draws a spiral</p>	<p>Hint: After turning, the sprite should draw a longer line. Find where the value of distance is and make sure it increases inside the loop</p>

Exercise 2:

What to do:	Using/Details:
<p>Explore changing what the spirals look like. Try making the spirals:</p> <ul style="list-style-type: none"> • A different width by changing the pen size • Different sizes 	
<p>Change the code to draw spirals at random locations instead of where the mouse is clicked.</p>	
<p>Make sure all code comments match the code if you made changes</p> <p>Explore adding some sounds to your project to make it really cool!</p>	
<p>Describe the changes you made to the starter project and any useful instructions in the project page and share your project.</p>	<p>Instructions</p> <p><i>Tell people how to use your project (such as which keys to press).</i></p>