

Module 2

Drawing Shapes and Repeating

Lesson 1 Think Like a Computer

Students explore algorithms in more detail. Students will write and design algorithms to draw pictures. Students have the opportunity to work with a partner to make sure their algorithms are correct.



Printable Resource



Group Work

Lesson 2 Could You Repeat That Please?

Computers are much better than humans at performing repetitive tasks. Students are introduced to the idea of iteration/repetition in Scratch, by drawing different shapes such as squares, triangles and circles.



Computer



Scratch



Projector



Printable Resource

Lesson 3 Over and Over Again

Students know how to use Scratch to perform repetitive tasks and draw different shapes. Repetition is explored in more detail in this lesson.



Printable Resource



Scratch



Projector



CD Resource



Lesson 1 – Think Like a Computer¹

Resources:

Pen and Paper for drawing pictures, Handout with Pictures 1 & 2 and Algorithm 1 with Algorithm 1 Solution (Resource 1)

Key Vocabulary:

Algorithm, Testing

Description:

Students were introduced to algorithms in Module 1 Lesson 4. This lesson explores computer algorithms in more detail. Two volunteers from the class are asked to write the algorithm for drawing a traffic light and a snowman on the board. The class will follow the algorithm and compare their results. All students should draw the same images. Algorithm 1 is presented to students. Again the class will follow the algorithm and compare their results. All students should draw the same image. Students are asked to work in pairs to design and write an algorithm for drawing a picture of their choice. Students are asked to work in pairs to design and write an algorithm that moves from the front door of the school to a mystery location.

Learning Objectives:

1. To understand how computers interpret instructions even if this means that the outcome is unexpected.
2. To show that the person writing a computer program must have a very clear understanding of what needs to be accomplished before writing the computer program.
3. Once a computer program has been written it can be tested to make sure it functions as expected.

Lesson Introduction:

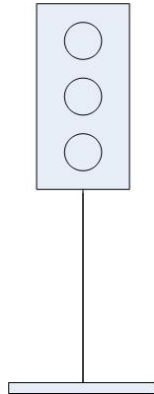
Discuss whether it would be good if people followed instructions exactly. E.g. what would happen if you pointed to a closed door and said “Go through that door?” Explain that computers work by following lists of instructions, and that they do exactly what the instructions say, even if they are incorrect (or nonsensical). This lesson explores algorithms for drawing images. This lesson shows the importance of writing and interpreting algorithms correctly.

Lesson Breakdown:

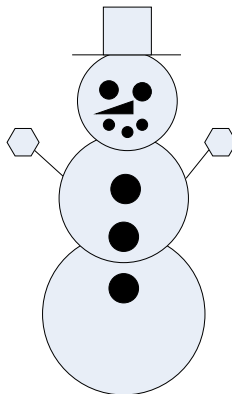
1. Students are given 3 blank sheets of paper each and asked to take out a pen or pencil for a drawing exercise.

¹ This lesson is based on Computer Science Unplugged Activity 12 - Programming Languages <http://csunplugged.com>

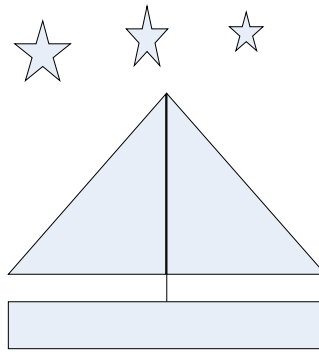
2. Choose a student and give them the following image (Resource 1, Picture 1). Ask them to write a set of instructions on the board to describe how to draw the image.



3. Once pupils have attempted to draw the image from the description ask them to hold up their drawings for one another to compare. Reflect on the problems involving size and shape when drawing the shape.
4. Choose a student and give them the following image (Resource 1, Picture 2). Ask them to write a set of instructions on the board to describe how to draw the image.



5. Once pupils have attempted to draw the image from the description ask them to hold up their drawings for one another to compare. Reflect on the problems involving size and shape when drawing the shape.
6. Present students with the Algorithm 1 (Resource 1, Algorithm 1) and ask them to draw the image following the set of instructions.
7. Once students have completed the task, ask them to hold up their images for one another to compare.
8. Did all students draw the following image “Sailing by night”? Could the algorithm be improved?



9. Students are paired for the next activity. Each student must write an algorithm for a drawing of their choice. They must not tell their partner what their algorithm draws.
10. Students swap algorithms. When both students have completed drawing they must compare the results and make sure they followed the instructions correctly and drew the correct picture. If necessary the algorithm can be refined until the most clear and efficient set of instructions is produced.
11. Students are paired for the next activity. Ask students to write instructions for each other about how to get to a mystery location in the school by starting at the front door.
12. Students then test their partner's algorithm by trying to follow their instructions to the mystery location. If necessary the algorithm can be refined until the most clear and efficient set of instructions is produced.

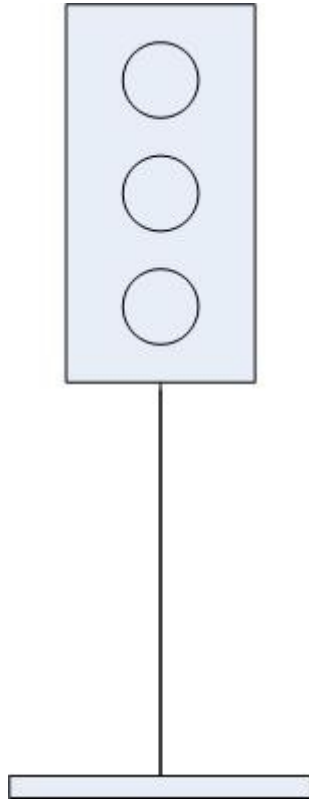
Resource 1

Handout with Pictures 1 & 2 and Algorithm 1 with Algorithm 1 Solution

Students are asked to write an algorithm on the board for the class for the pictures in the handouts.

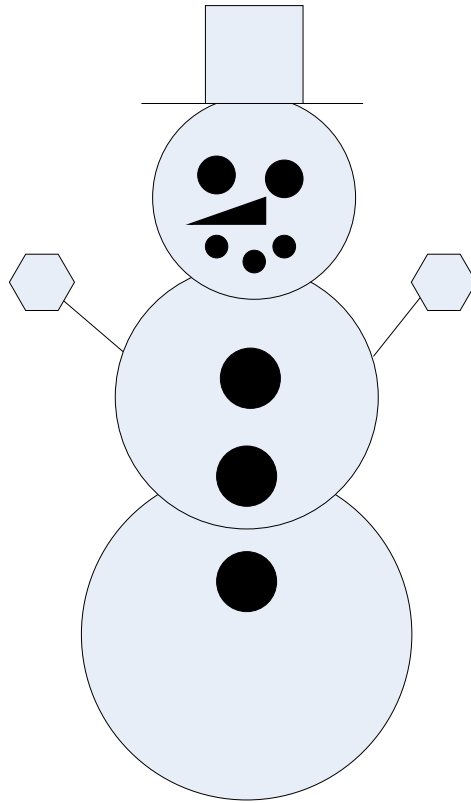
Picture 1

Describe how to draw this picture using written instructions only.



Picture 2

Describe how to draw this picture using written instructions only.

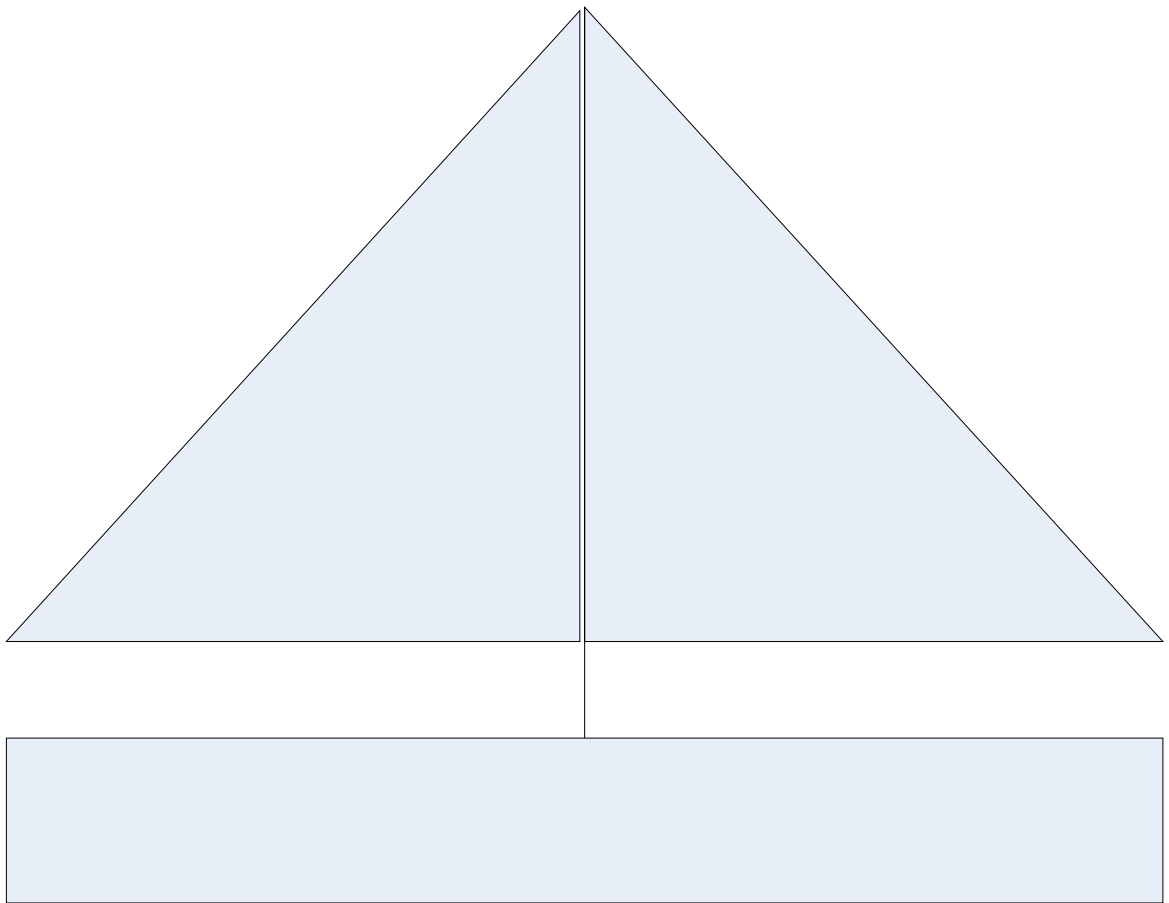
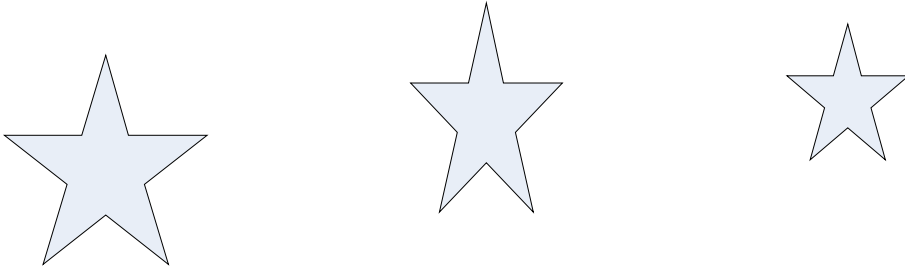


Algorithm 1

Follow the instructions in this algorithm to draw something.

1. Draw a rectangle.
2. From the centre of the rectangle's uppermost longer side, draw a straight line upwards perpendicular to the rectangle. This straight line should be the same length as the longer side of the rectangle.
3. Draw a right-angled triangle on the right hand-side of the perpendicular line.
4. Draw another triangle on the left hand-side of the perpendicular line. This triangle should be a mirror image of the triangle drawn in step 3.
5. On the top of the page, draw three stars.

Solution to Algorithm 1



Lesson 2 – Could You Repeat That Please?

Resources:

Draw a Square Program (Resource 1)

Key Vocabulary:

Debugger, Iteration, Random Numbers, Script, Sprite

Description:

Students are introduced to the idea of iteration/repetition and loops in Scratch. The students are given a set of commands and units of measurement on the board. Students give instructions to a student to walk around the class, using various commands. The Scratch program to draw a square is presented to students and they must guess what it does before trying the script themselves. The debugger is introduced for the square program and students experiment by writing their own scripts to create different shapes in Scratch.

Learning Objectives:

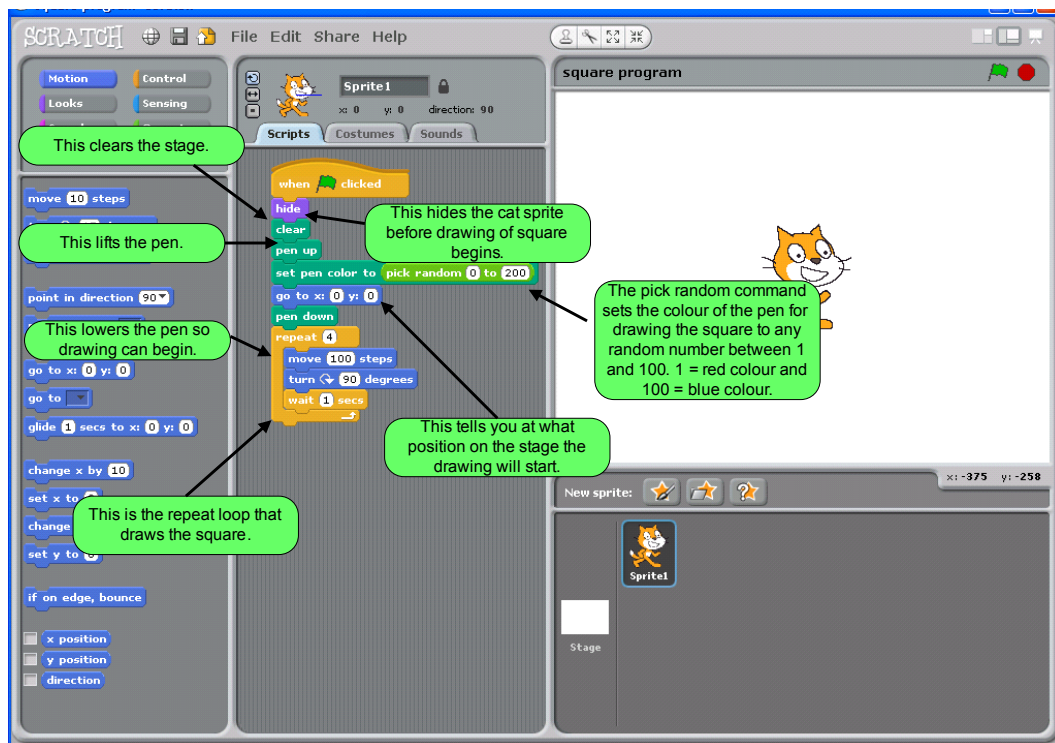
1. To introduce students to the concept of iteration and loops in Scratch.
2. To allow students to experiment with drawing shapes in Scratch.

Lesson Introduction:

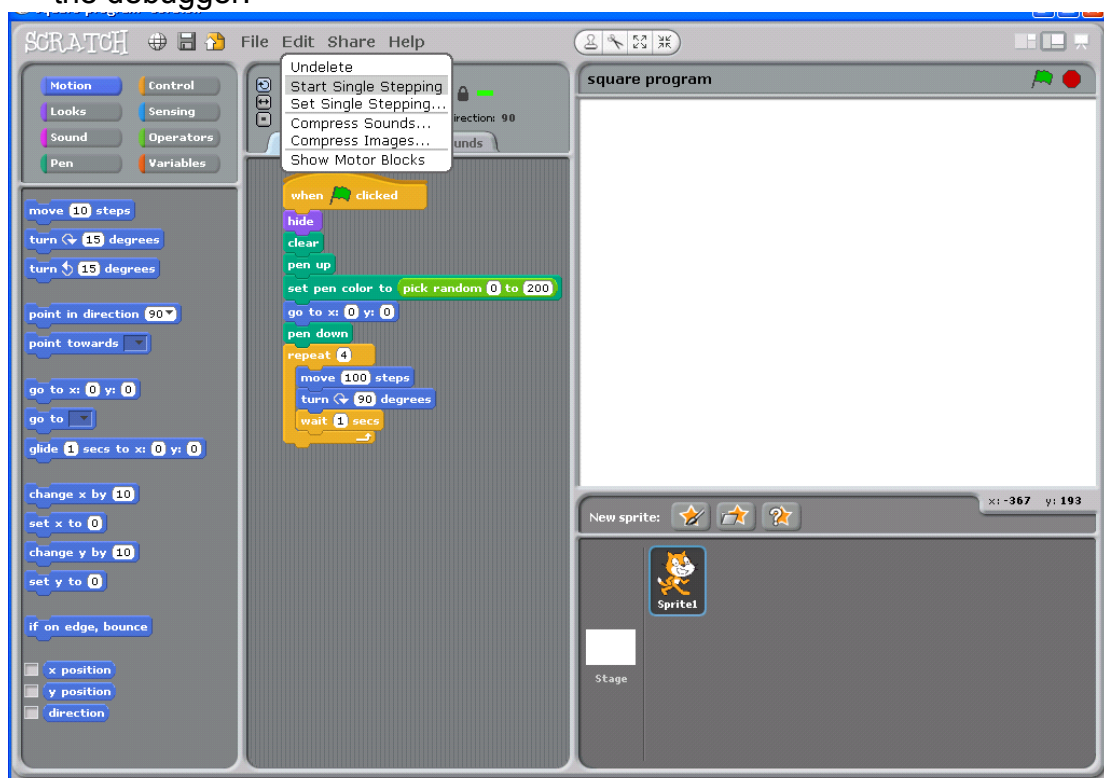
Reiterate the idea that computers need clear instructions to perform tasks. Tell students they will follow certain instructions given by other students first. They will then use Scratch to give the computer instructions to draw different things.

Lesson Breakdown:

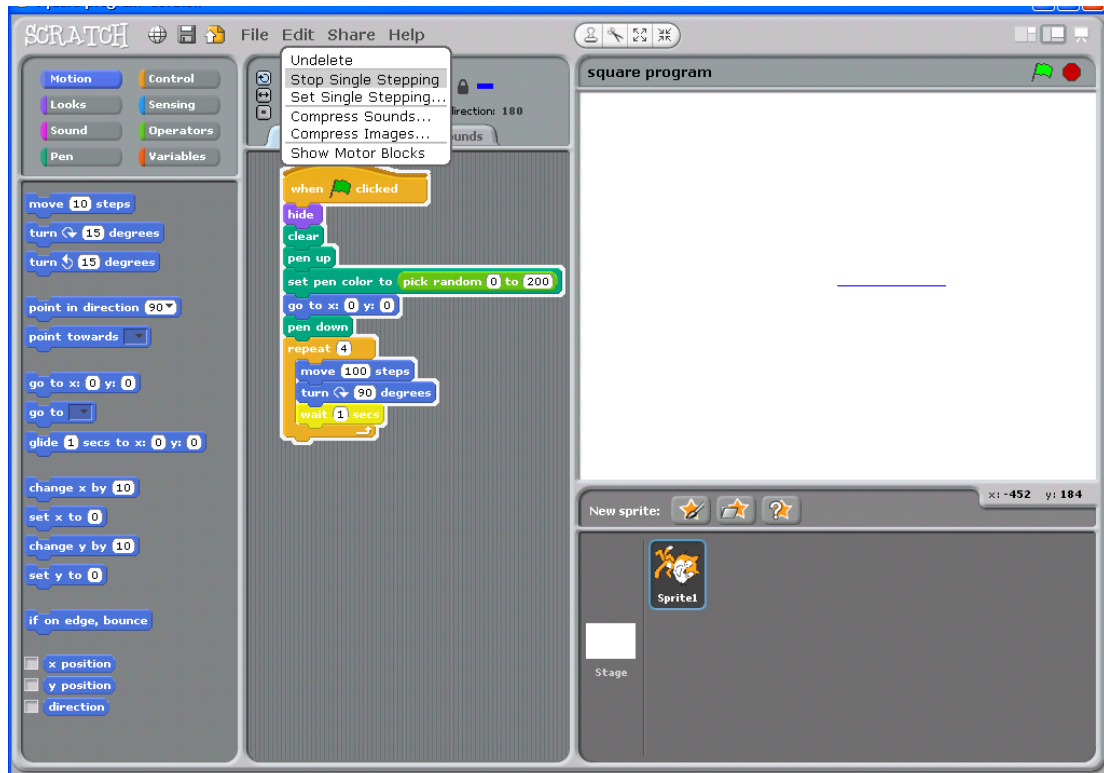
1. Place a chair in an area of the room where there is sufficient space to move around and choose a student to sit in the chair.
2. Write a series of commands on the board e.g. stand, sit, walk, turn as well as units such as steps and degrees.
3. Start by giving instructions to the student to stand, turn a number of degrees, walk a number of steps etc. Ask some of the other students to give similar commands.
4. Once the previous activity is finished, give students a copy of the Draw a Square Program from Scratch (Resource 1) and ask them to figure out what it does.
5. Ask the students to create the script themselves in Scratch to see what it does.
6. Explain the commands hide, clear, pen up, pen down, set pen colour to, and pick random number as described in the following diagram.



- Introduce students to the debugger or single stepping in Scratch. The debugger highlights the code step by step as things happen on the stage. Explain that this is useful if students do not understand a script or if there is a problem with it. The diagram below shows how to start the debugger:



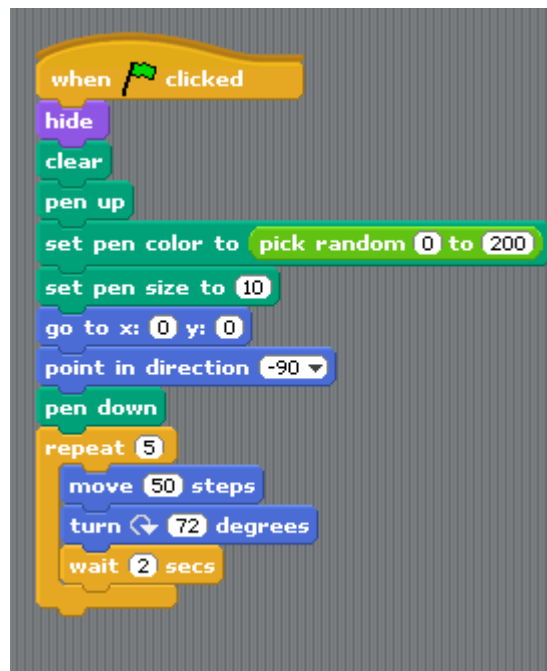
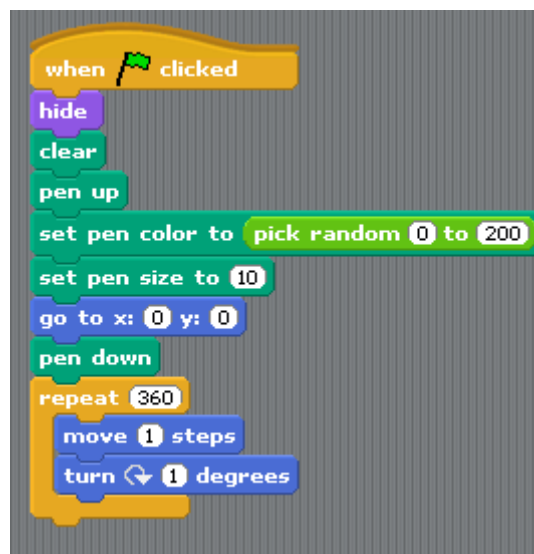
- The diagram below shows how to stop the debugger:



9. Ask students to draw a triangle, a pentagon and a circle. See following diagrams for solutions:

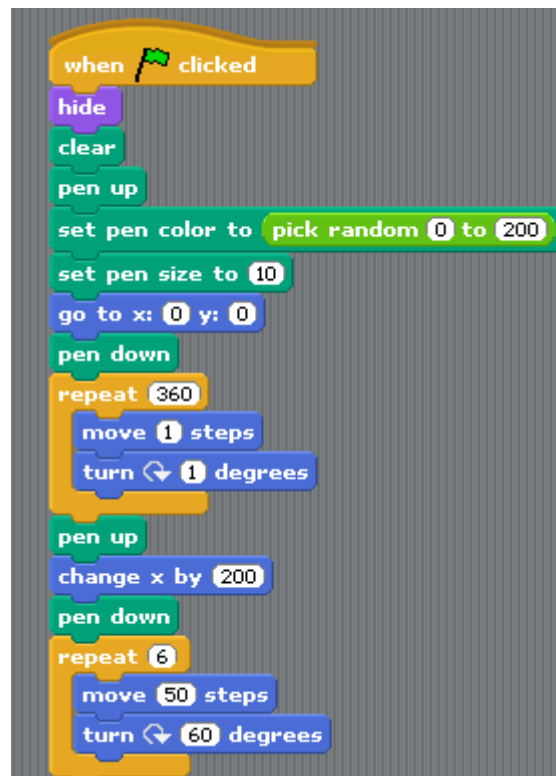
Triangle:



Pentagon:**Circle:**

10. Ask students to draw a circle and hexagon side by side.

Circle and hexagon side by side



The script above changes the x value by 200 while the pen is up so that the hexagon shape is drawn beside the circle. Encourage students to experiment with these shapes.



Extension activity

Ask students to draw something interesting using 4 or 5 shapes. They can use different pen sizes and different colours e.g. a house, a car or a smiley face.

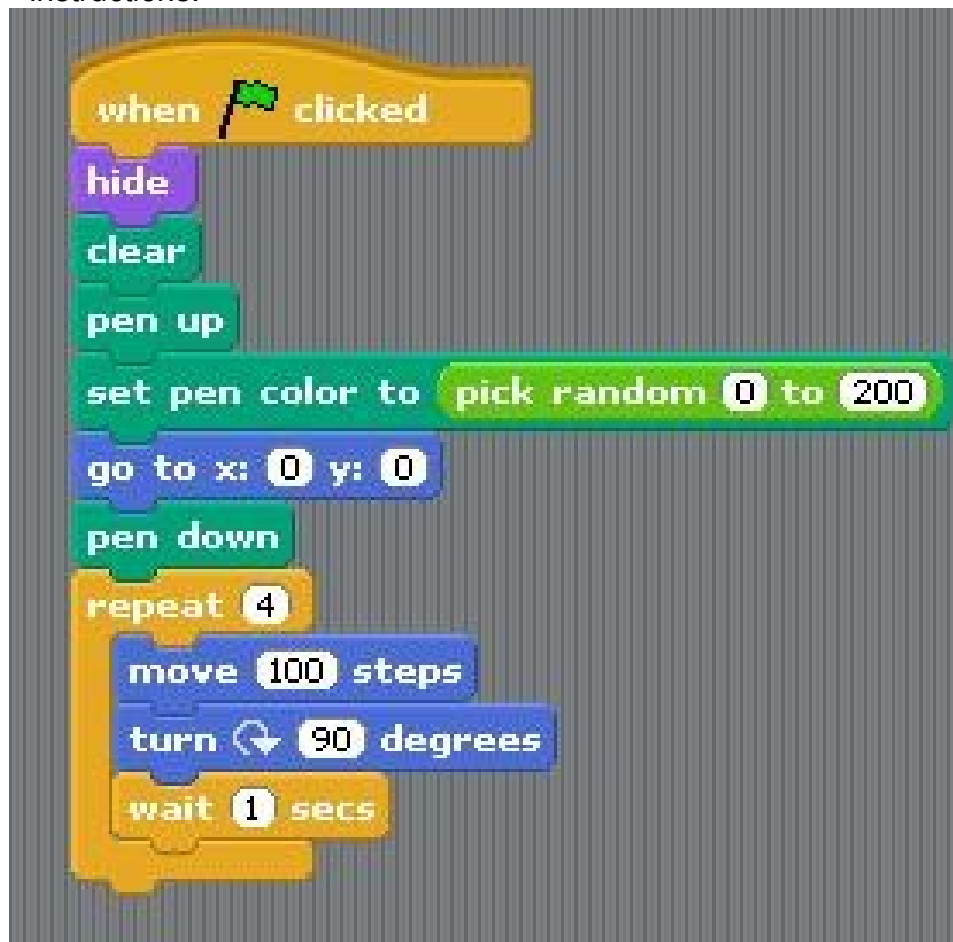
Resource 1

Draw a Square Program

Students are asked to figure out what actions the Scratch script performs. They then build the script using Scratch.

Look at the script from Scratch below. Can you figure out what it does?

Hint: Imagine you are a sprite walking around the stage receiving these instructions.



Lesson 3 - Over and Over Again

Resources:

Nested loops worksheet (Resource 1), Solution for drawing the 4 Shapes Pattern (Resource 2), Solution for Coloured Nested Squares (Resource 3), Solution for drawing Flower (Resource 4)

Key Vocabulary:

Iteration, Loops, Nested Loops

Description:

Students are introduced to the idea of a nested loop or a loop inside a loop. This lesson is a follow-on lesson from Lesson 2, where iteration and the idea of repeat loops were introduced. Students will create a simple script containing nested loops and will then create various scripts to draw patterns with different coloured shapes.

Learning Objectives:

1. To familiarise students with the notion of nested loops.
2. To allow students to experiment with drawing different shapes and patterns using nested loops.

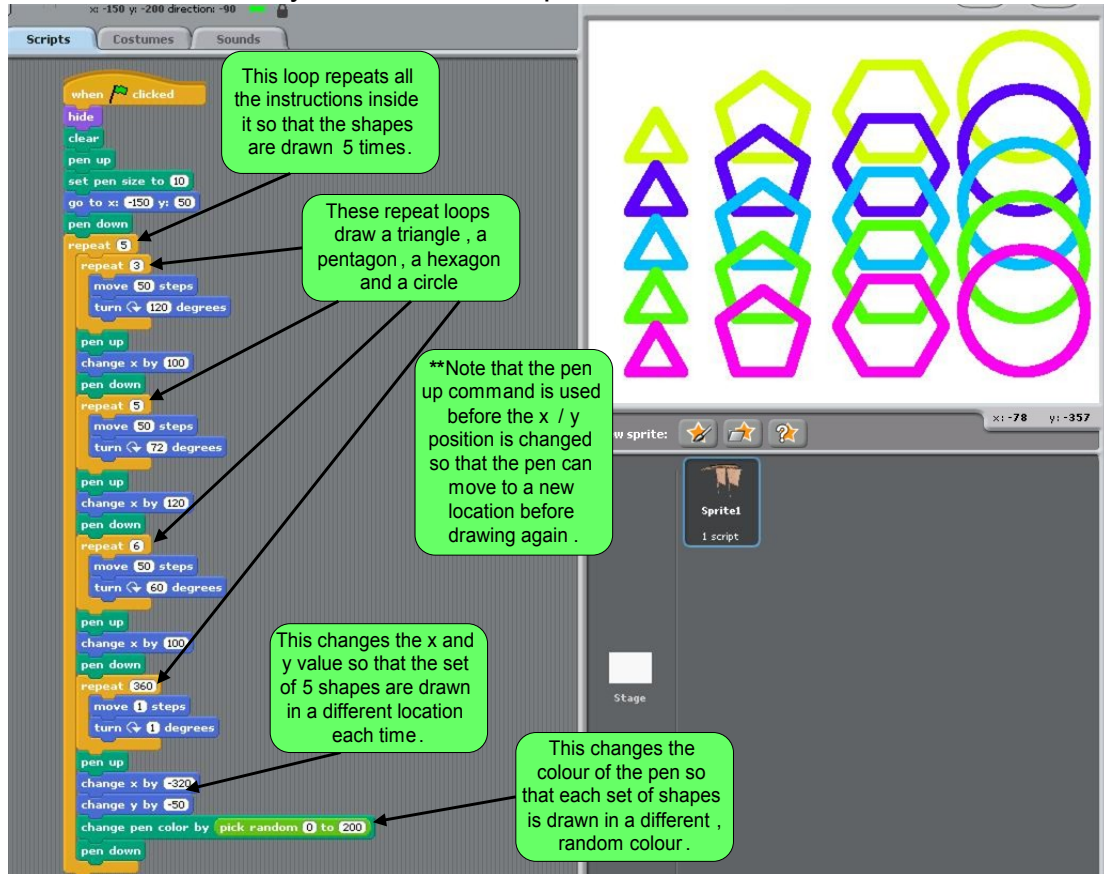
Lesson Introduction:

- Reiterate the idea of iteration as being repetition from the last lesson, Module 2 Lesson 2.
- Tell students that today they will be using loops nested inside other loops to create interesting shapes.

Lesson Breakdown:

1. Present students with Exercise 1 on the Nested Loops worksheet (Resource 1). Ask the students to try the script to see what happens.
2. Ask them to use the debugger/single stepping to slow things down if they are unsure of what is happening in the script. They can do this by clicking on the extras button, and choosing Start Single Stepping from the drop-down menu, before clicking the green flag.
3. Discuss the idea of nested loops or a loop within a loop from the triangle and pentagon example on the sheet. Students should realise that the outer loop removes the need to rewrite all code inside the loop many times. This reduces the amount of code overall.
4. Ask students to modify the program by changing the colour of the shapes in the loop and to change the number of repetitions of the outer loop to 3. They should notice that the triangle and pentagon are drawn 3 times.

5. Show students the Scratch file “Solution for drawing 4 Shapes Pattern” (Resource 2) (See screenshot of this solution below). Ask students to draw this pattern using 4 shapes: triangle, pentagon, hexagon and circle. Ask them to make the shapes overlap as in the solution. All shapes have been drawn individually in the previous lesson, Lesson 2.

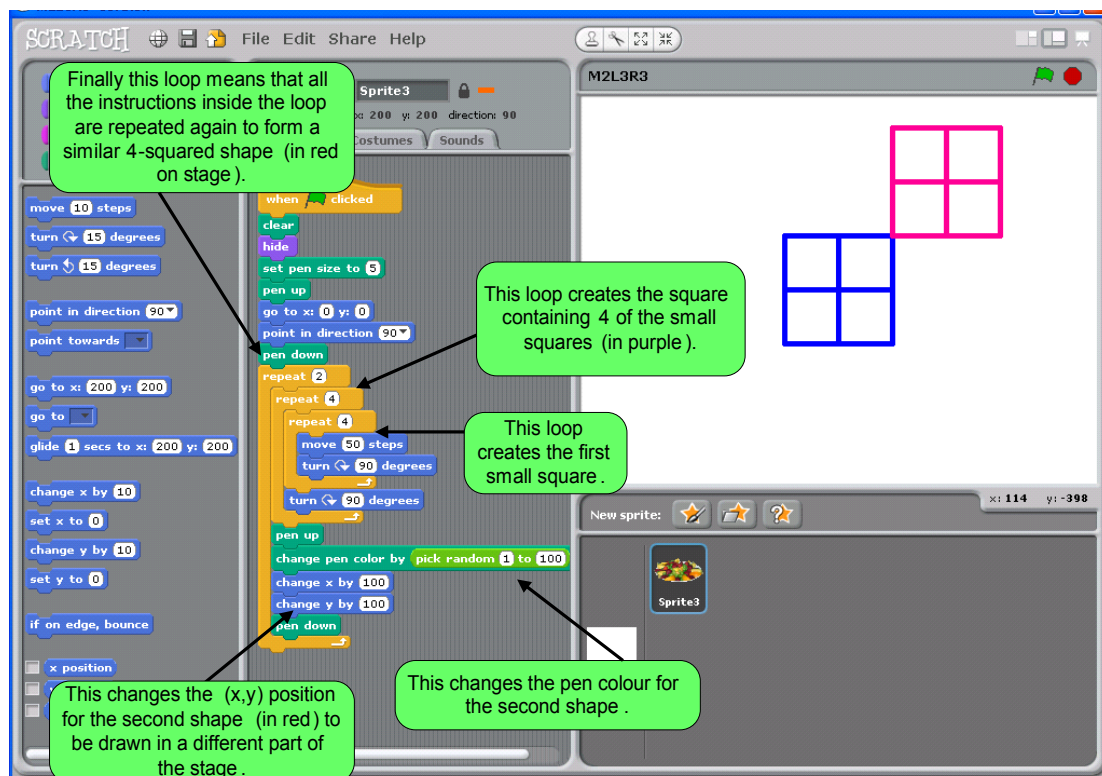


Students may be given hints as follows:

- Tell students they should clear the screen and hide the current sprite each time the green flag is clicked at the start.
- Tell students they must have an outer loop to repeat the drawing of each shape 5 times.
- Give hints about sides and angles if necessary i.e. the angle for each shape = 360 divided by the number of sides e.g. For a triangle, the number of sides = 3 and the angle = 120 degrees (3×120), for a pentagon, the number of sides = 5 and the angle = 72 degrees (5×72), for a hexagon the number of sides = 6 and the angle = 60 degrees (6×60) and a circle = 360 degrees (360×1). Also remember not to move too many steps for each side e.g. 50 steps is a good start so all shapes will fit on stage.
- Tell students that before drawing the first triangle on the first row, the x and y position needs to be set. Before drawing the pentagon on the first row, the x position needs to be changed. Before drawing the hexagon on the first row, the x position needs to be changed again and before drawing the circle on the first row, the x position

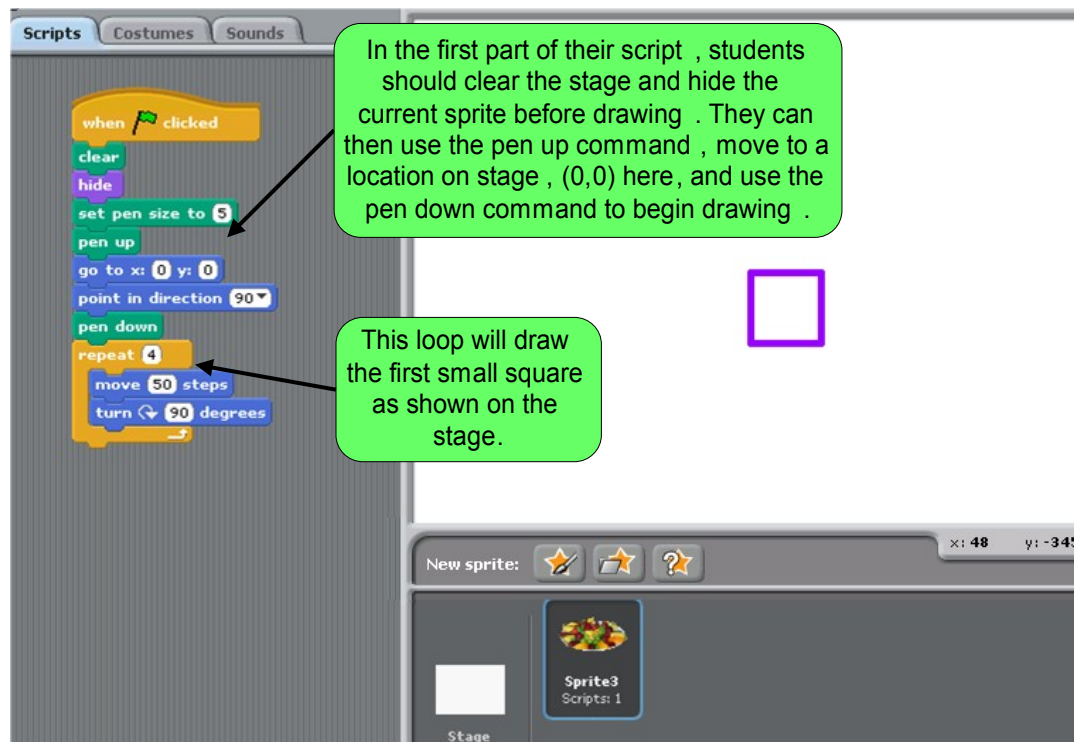
needs to be changed once again. Changing the x position each time makes room for the next shape to be drawn.

- Tell students to remember to use the pen up command before changing the x position when moving to draw their next shape and not to forget to use the pen down command again before they begin to draw that next shape.
 - Tell students that once they have drawn the first row of 4 shapes, they must change the x value to return to the start of the next row of shapes. They must also change the y value to move down the page before they begin to draw the first shape in the next row of shapes. Students will also change the pen colour when drawing a new row of shapes.
6. Ask students to draw the shape below, using 3 loops as illustrated in the solution below. This drawing is in Exercise 2 on the nested loops worksheet (Resource 1).

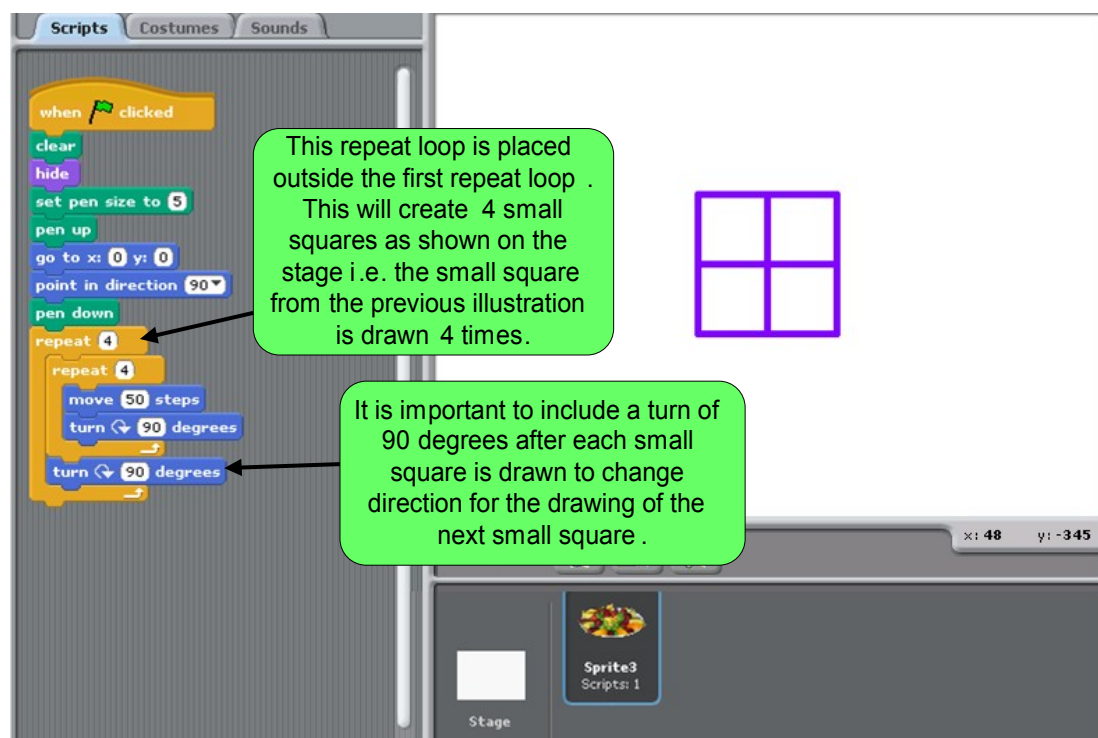


Students may be given hints as follows:

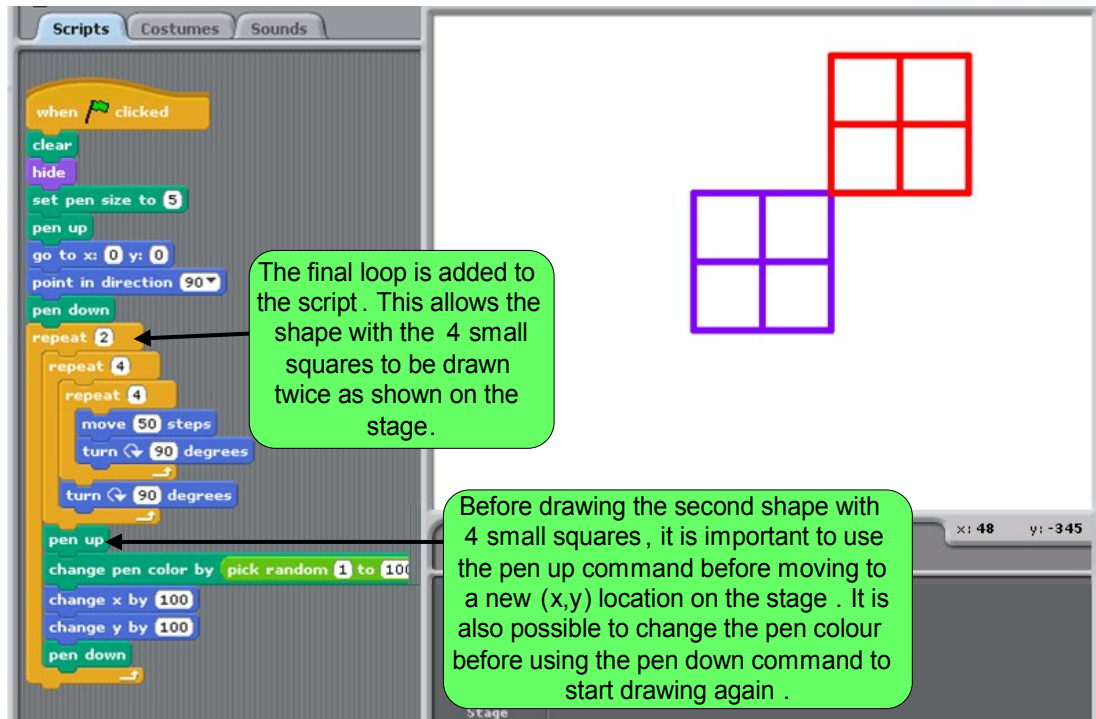
- Ask students to draw one of the small squares first. The following image illustrates how to create a script for this.



- Ask students to draw 4 small squares together, using another repeat loop (nested loop). The following image illustrates how to modify the previous script.



- Ask students to draw the final shape and, if necessary, prompt them to use a final loop outside the other 2 loops to draw the previous shape twice, as shown in the following image.



The Scratch solution file (Resource 3), which contains the entire script, may be shown to the students once they have tried the problem.

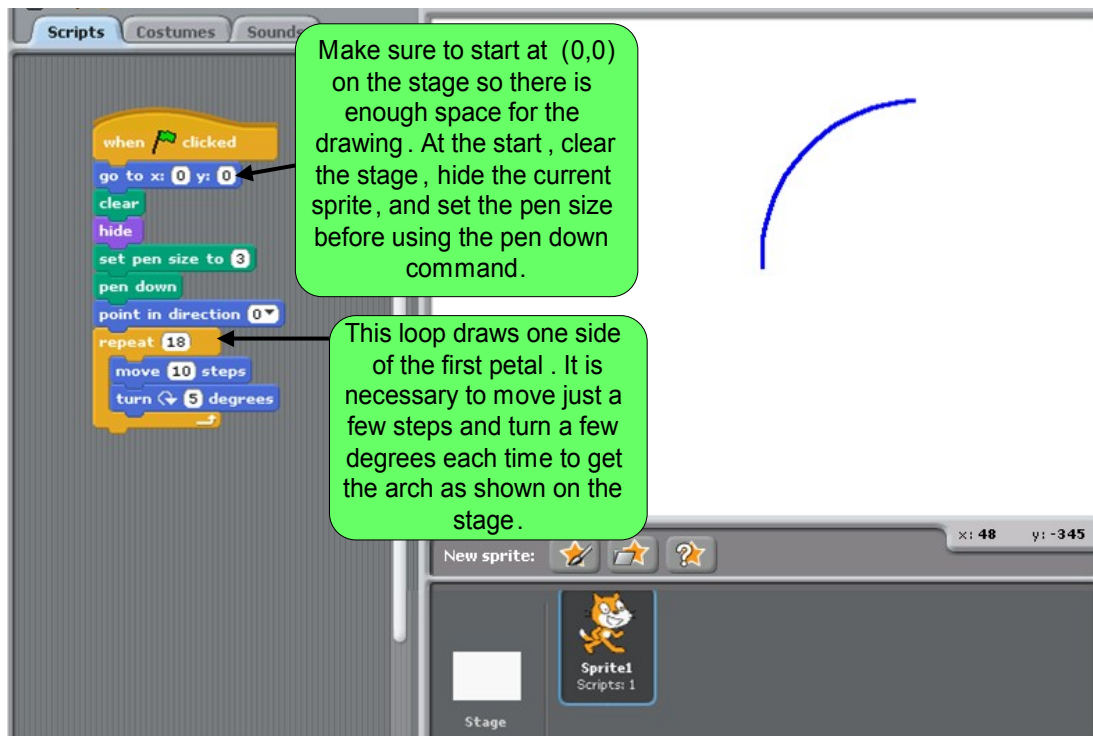
➡ Extension activity

Ask students to draw the shape in Exercise 3 on the nested loops worksheet (Resource 1). They will use 3 nested loops.

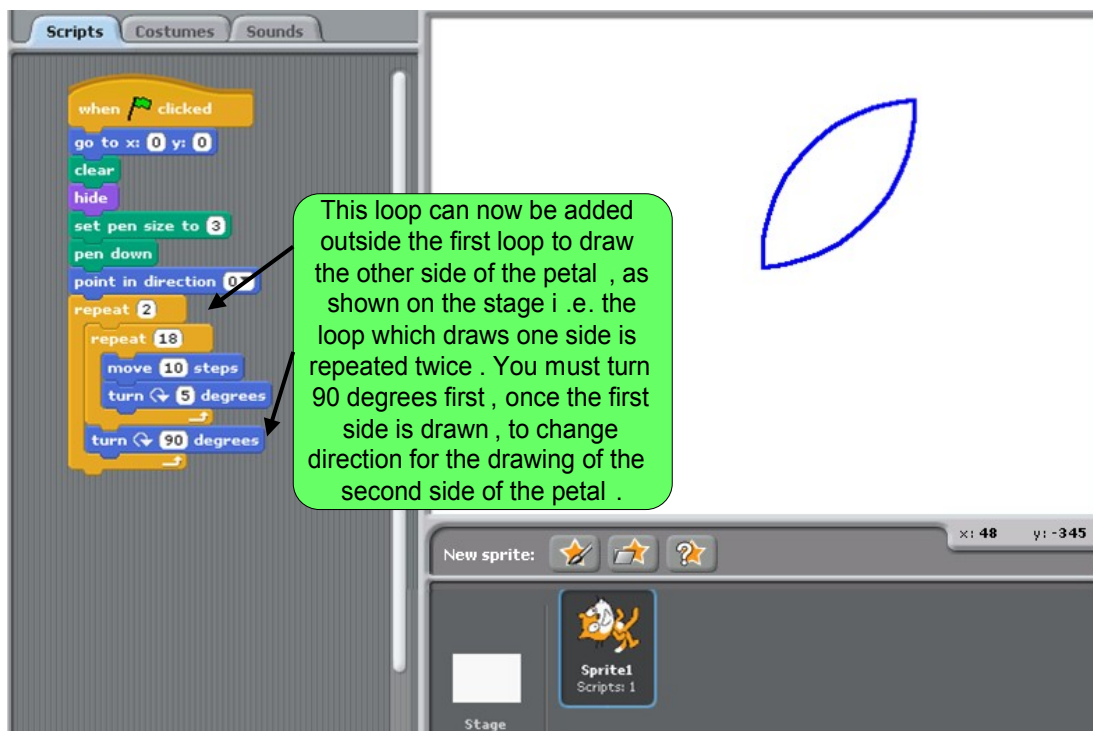
The Scratch solution file (Resource 4), which contains the entire script, may be shown to the students once they have tried the problem.

Students may be given hints as follows:

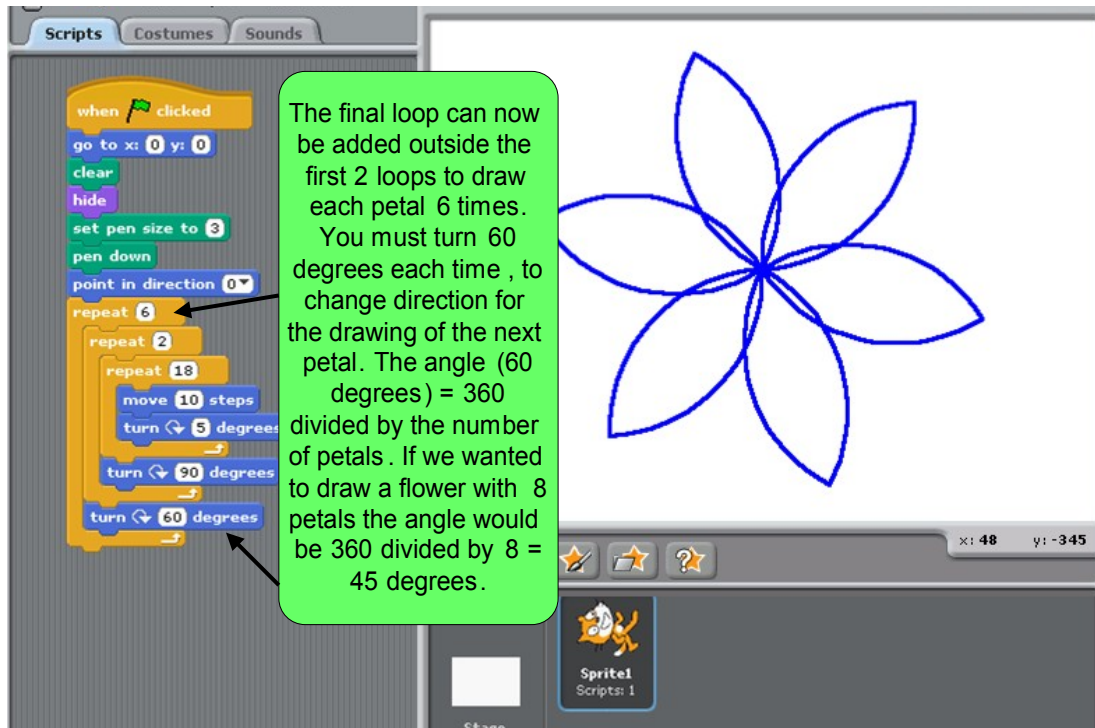
- Ask students to draw one side of the first petal as shown in the following image to get started. It is important that they start at a specific point on stage i.e. (0,0) in this example, and that they hide the current sprite before beginning.



- Ask students to use a repeat loop to draw the other side of the petal, as shown in the following image. They must remember to turn 90 degrees to change direction before drawing the other side of the petal, as shown.



- Ask students to use a final repeat loop to draw each petal 6 times, as shown in the following image. They must remember to turn 60 degrees to change direction before drawing each new petal, as shown.



The final loop can now be added outside the first 2 loops to draw each petal 6 times. You must turn 60 degrees each time, to change direction for the drawing of the next petal. The angle (60 degrees) = 360 divided by the number of petals. If we wanted to draw a flower with 8 petals the angle would be 360 divided by $8 = 45$ degrees.

The image shows a Scratch script on the left and a drawing of a flower on the right. The script is as follows:

```
when clicked
  go to x: 0 y: 0
  clear
  hide
  set pen size to 3
  pen down
  point in direction 0
  repeat 6
    repeat 2
      repeat 18
        move 10 steps
        turn 5 degrees
      turn 90 degrees
    turn 60 degrees
```

The drawing on the right is a blue line drawing of a flower with six petals. The coordinates at the bottom right of the drawing area are x: 48 y: -345. The stage area at the bottom shows a sprite named 'Sprite1' with 1 script.

Resource 1

Nested loops worksheet

Exercise 1 contains a mystery script. Students write Scratch programs to draw the shapes in Exercise 2 and 3.

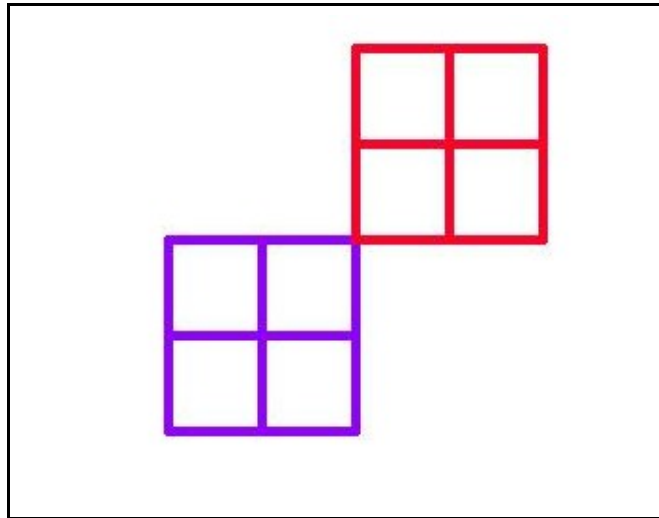
Exercise 1

Build and execute this script and see what happens...



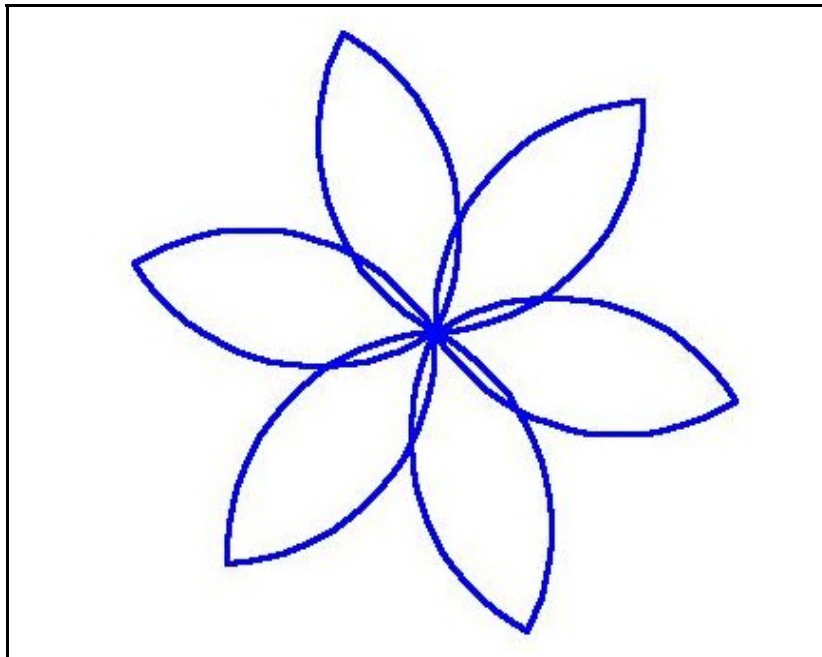
Exercise 2

Create the shape below using nested loops.



Exercise 3

Create the shape below using nested loops.



Resource 2

Solution for drawing the 4 Shapes Pattern

Scratch file with a possible solution for drawing the 4 shapes (triangle, pentagon, hexagon and circle) pattern.



CD Resource

"M2L3R2 Drawing 4 Shapes Pattern.sb"

Resource 3

Solution for “Coloured Nested Squares”

Scratch file with a possible solution for Exercise 2 from Resource 1.



CD Resource

“M2L3R3 Coloured Nested Squares.sb”

Resource 4

Solution for drawing Flower

Scratch file with a possible solution for Exercise 3 from Resource 1.



CD Resource

“M2L3R4 Flower.sb”