SECU0057: Simulation for Research – Component 1

30 MARKS

Purpose

This component aims to assess your ability to create simple Netlogo codes. For this, eight tasks have been designed that require selecting appropriate programming concepts and implementing them using Netlogo primitives. You must complete all the tasks.

Instructions

Download the files called "template.nlogo" and "spiral1.csv", and save them in the same folder.

Write eight procedures (go1, go2, go3, go4, go5, go6, go7 and go8) in the same Netlogo file.

All these procedures must be coded according to the instructions below.

Marking criteria:

Each task contains a number of requirements, including:

- *major requirements* that are designed to test important learning objectives (e.g., "being able to change the colour of a patch") and
- *minor requirements* that do not correspond to important learning objectives (e.g., the specific colour of the spiral in Tasks 1- 5)

Your answers will be assessed against the following marking criteria:

| Mark		Description
0/3	0/5	No evidence that the student has met any of the learning objectives tested in the task.
1/3	1/5	None of the major requirements have been met in the code.
	2/5	Multiple major requirements have not been met in the code.
2/3	3/5	One major requirement has not been met in the code.
	4/5	All requirements have been met in the code, except a few minor ones.
3/3	5/5	The student has managed to successfully complete the task (e.g., generate the spiral depicted on the figure) and all the requirements of the tasks have been met.

Questions?

Questions about the assessment must be posted on the Moodle forum by the 7th November. We will aim to answer them within 48 hours. Note: If you post a message on a Friday 5pm, you may not receive an answer before Tuesday 5pm.

Task 1. Procedure go1 [3 marks]

The file "spiral1.csv" includes a table with:

- As many rows as patches in the spiral represented in Figure 1.
- Three columns containing to the x-coordinates, y-coordinates and colour codes of the spiral's patches.

Write a Netlogo code that extracts the coordinates of the patches from the csv file, and set their colours according to the third column. You must not create turtles in this procedure.
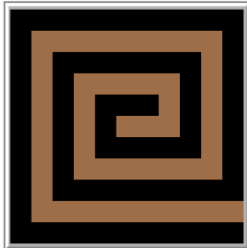
Background colour: black

Spiral colour: see spiral1.csv

Figure 1: Anti-clockwise spiral generated from the data in spiral1.csv

Task 2. Procedure go2 [3 marks]

Create a turtle at the centre of the interface (0,0). Make the turtle draw a spiral, as shown in Figure 2. For this, set the colour of the spiral patches as the turtle moves along the spiral, using the command "ask turtles [set pcolor...]". Destroy the turtle when it reaches the end of the spiral.

Trajectory: The trajectory of the turtle must be recorded in a list called "instructions". The list must contain two parameters for each segment of the spiral:

- Bearing: use the code ("N", "E", "S", "W") to represent North, East, South, West; and
- The number of patches that the turtle must cover before the next turn.

The list – E, 2, N, 2, … – must be read using "foreach". You must use the "(ifelse" command to convert the codes to directions.

Background colour: black

Spiral colour: red

Figure 2: Anti-clockwise spiral starting Eastward

Task 3. Procedure go3 [3 marks]

Create a turtle at the centre of the interface (0,0), pointing upward. Make the turtle draw a spiral, as shown in Figure 3. For this, set the colour of the spiral patches as the turtle moves along the spiral. Destroy the turtle when it reaches the end of the spiral.

Trajectory: The trajectory of the turtle must be recorded in a list called "instructions". The list must contain two parameters for each segment of the spiral:

- The number of patches that the turtle must cover before the next turn (segment length).
- Rotation at the end of the segment: use the code ("L", "S", "R", "U") to represent a 90° turn on the left, no rotation, 90° turn on the right, and 180° turn, respectively.

The list – 2, L, 2, L, … – must be read using the command "foreach".
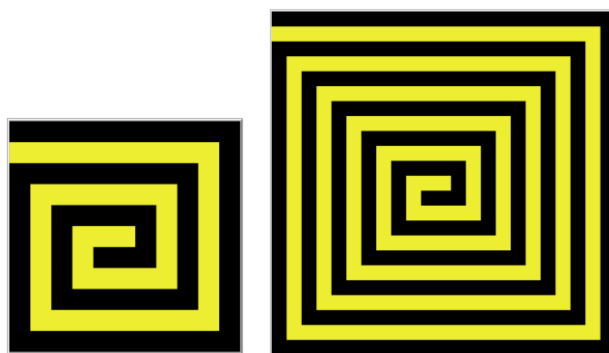

Background: black

Colour of the spiral: blue

Figure 3: Anti-clockwise spiral starting Northward

Task 4. Procedure go4 [5 marks]

Create a turtle at the centre of the interface (0,0), pointing to the left. Make the turtle draw a spiral, as shown in Figure 4. For this, set the colour of the spiral patches as the turtle moves along the spiral. Destroy the turtle when it reaches the end of the spiral.

Trajectory: The turtle must be moved using the commands "fd" and "left 90". In the previous procedures, the trajectory of the turtle was recorded using a pre-specified list. As a result, the spiral would not grow if we increased the size of the interface. To ensure that the spiral always fits the interface, the trajectory should be generated dynamically, using a rule-based approach. The rule should reflect the patterned geometry of the spiral. As an example, the following rule could be used to create a square frame starting from the bottom left corner: *repeat 4 [fd world-width right 90]*.

Background: black

Colour of the spiral: yellow

Figure 4: Anticlockwise spirals starting Westward - max-pxcor = 5 (left) and max-pxcor = 11 (right).

Task 5. Procedure go5 [3 marks]

In this task, you must draw a spiral starting from the centre of the interface (see Figure 5), using the command "ask patch x y [set pcolor…]". The sequence of patches must be identified using their coordinates, without using any turtles.

Trajectory: The order in which patches must be recoloured must be based on the elements of a list called "instructions". These elements are similar to those in the go2 procedure:

- Bearing ("N", "E", "S", "W") and
- The number of patches that must be covered before the next turn.

The list – S, 2, E, 2, … – must be read using the command "repeat".

Background: black

Colour of the spiral: green
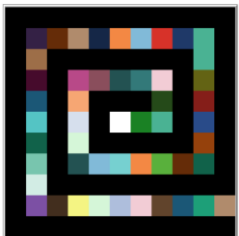
Figure 5: Anti-clockwise spiral starting Southward

Task 6. Procedure go6 [5 marks]

Create a turtle at the centre of the interface (0,0) and change the colour of its patch to white. Make the turtle draw a spiral, as shown in Figure 6. Set the colour of the spiral patches as the turtle moves along the spiral. Destroy the turtle when it reaches the end of the spiral.

Trajectory: For this task, the turtle must be given destination points and move toward them in straight line. The destination points should correspond to the corners of the spiral, and their coordinates should be recorded in a list called "destination_list". The list must start as follows: [2, 0, 2, 2, …], where the first two items correspond to the first destination point (2, 0), and the next two items correspond to the second destination point (2, 2), and so on and so forth. The iteration over the list must be done using the command "repeat". To stop the movement of the turtle, you must use a while loop that stops when the turtle reaches the destination patch.

Colour of the spiral: Except for the central patch (which must be white), all other patches should be coloured randomly. To create a more colourful spiral, you must exclude all colours whose numerical codes are less than 11 and those whose codes end with a 0 or a 1.

Background: black

Colour of the spiral: multicolour

Figure 6: Anti-clockwise spiral starting Eastward

Task 7. Procedure go7 [3 marks]

Once a spiral has been created, it can be useful to save it in a csv file. Write a Netlogo code that can iterate over all the patches of the interface (including background patches and spiral patches), and save the following information in a csv file called ("spiral2.csv"):

- Column 1: x-coordinates of the patches
- Column 2: y-coordinates of the patches
- Column 3: colour codes of the patches

Each row should correspond to a different patch

You must not create turtles in this procedure.

Task 8. Procedure go8 [5 marks]

Task 8 is similar to Task 7. Write a Netlogo code that can iterate over the patches of the spiral only, and save the following information in a csv file called ("spiral3.csv"):

- Column 1: x-coordinates of the patches
- Column 2: y-coordinates of the patches
- Column 3: colour codes of the patches

Each row should correspond to a different patch of the spiral.

You must not create turtles in this procedure. You must use the "of" and "foreach" commands in your code.

Finalising your answers and verifying your code
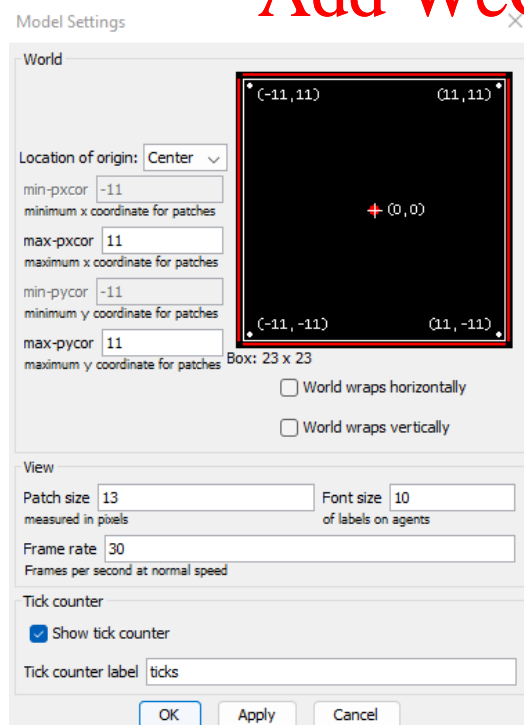
If you have managed to complete all the tasks, make the following modification to your code before uploading your code on Turn-it-In:

• At the end of the go6 procedure, add a line that calls the go7 procedure. This will save the colourful spiral in "spiral2.csv".

• Edit the go1 procedure so it extracts data from "spiral2.csv" instead of "spiral1.csv".

• Resize your interface as shown in the image.

This modification should allow you to display the colourful spiral anytime you want. It will also help us check that your code works.

END OF COMPONENT 1