

Lab 02: Extension

# Course Learning Outcomes

CLO 1 Describe the fundamental structures of an agent-based programming language

CLO 2 Solve a problem by using an agent-based programming language

CLO 3 Compose logical structures (algorithms) to produce an adequate solution for a problem

CLO 5 Develop and enhance problem-solving skills by applying computational thinking methodologies

# Module Learning Objectives:

1. Demonstrate usage of basic commands by creating a NetLogo program that draws their initials. (CO 1, 2, 3, 5)
2. Demonstrate positioning the turtle in the NetLogo world. (CO 1)
3. Demonstrate the usage of colors in NetLogo. (CO 1)
4. Practice utilizing turtle and observer context in NetLogo. (CO 1)
5. Utilize procedures in NetLogo. (CO 1, 3)
6. Design a NetLogo program to display the student’s initials. (CO 1, 2, 3, 5)

# ASSIGNMENT:

To complete the Lab02 Extension the student should add another button to the interface labeled “extension”. The extension button should execute code that incorporates the student’s name or initials into a design for a ‘brand’ or ‘logo’.

This is the student’s chance to be creative with Netlogo. The logo should be visually appealing and capture something that is unique about the student or something they care about.

Include multiple shapes and colors in the design. Additionally, include (at least) 2 of the following 3 features in the program:

* Change the color of the turtle’s line as it moves! In Netlogo, colors can be represented as numbers. This means a color gradient can be using simple mathematical operators (e.g., addition, subtraction, multiplication, etc.) and/or the “random” function. The color dictionary may be viewed here: <https://ccl.northwestern.edu/netlogo/docs/programming.html#colors>

For example, let’s say we made a turtle and asked it to set color red. Red corresponds to the number 15. To change its color to orange, we must add 10. We can ask the turtle to set color color + 10. For even more fun, use the random command. Try set color color + random 10, then try other values too!

* Use the “sprout” command to spawn multiple turtles so that part of the visual appeal of the logo is in the dynamics of multiple turtles drawing the logo. The program can ask patches to sprout 1 or more turtles. Use these commands to create a dynamic scene where parts of the logo are drawn either simultaneously or sequentially.
* Use equations to generate images, lines or shapes. For example, the sin, cos or log functions may generate interesting patterns. If the command reset-ticks is included in the setup procedure, the following make-circle procedure will draw a circle.

to make-circle

ask turtle 0 [setxy (10 \* cos ticks) (10 \* sin ticks)]

tick

end

Experiment with other mathematical functions and values to generate interesting shapes.

# Grading Rubric [26 points of Extra Credit]:

[A: 5 points]: Implement two of the three features listed above. Implement two of the three features: change line color, sprout multiple turtles, use an equation.

[B: 1 point]: Change the pen width.

[C: 5 points]: The program uses only one turtle and has separate buttons for each letter.

[D: 5 points]: The program includes a third button labeled “Extra Credit” which clears the screen and draws a house.

[E: 5 points]: Use multiple colors and shapes.

[F: 5 points]: The design should be clean, visually appealing, and unique.