Module 8

CS108L Computer Science for All
Module 8: Recursion and the Fractal Tree

Model Overview:

The goal for this model is to create a recursive program that draws a tree-like fractal pattern. We define a recursive program as one that contains a procedure that calls itself! Recursion is a powerful concept that frequently stretches our brains! For more details on recursion, see this week’s videos.

Setup:

- Create the following buttons: Setup and Go

- Create the following sliders: initialBranchLength
growthFactor
branchAngleParent
branchAngleChild
branchThickness

- In Settings, set the location of origin to edge and bottom.
- Set the following coordinates: min-pxcor = -23
  max-pxcor = 23
  max-pycor = 33
The Basic Structure:

- For this module you will need a minimum of three procedures: Setup, Go, and Grow.
- Structure your Grow procedure in a recursive manner.
- Move your turtle a certain distance and “hatch” a new turtle with its heading set to branchAngleChild to grow your tree.
- Create a global variable that keeps track of the branch thickness of your fractal tree. This value should decrease with every call to the recursion.

The overall approach to creating this model:

1) Build the basic model that creates a simple branching tree
2) Add changing the branch angle of the parent
3) Add in changing the branch thickness
4) Change the branch color
5) Change the branch thickness

Module 8: Recursion and the Fractal Tree Grading Rubric (20 Points Total)

<table>
<thead>
<tr>
<th>Done</th>
<th>Points</th>
<th>Task</th>
</tr>
</thead>
</table>
| 1    | A:     | • Submit a NetLogo source code with the file name: M1.firstname.lastname.nlogo.  
• The first few lines of your Code tab are comments including the following:  
  ;Student’s Name:  
  ;School:  
  ;Teacher's Name:  
  ;Date: |
| 2    | B:     | • The code in the Code tab of your program is appropriately documented with “in-line comments.” |
| 2    | C:     | • You include a description of your program in the Info section. See Coding Standards Guidelines for more information. |
5  | D:  
|   | • Clicking “Setup” and “Go” will run your program and create a fractal tree on the interface.  

5  | E:  
|   | • Clicking “Setup” and “Go” produces a treelike pattern with at least 3 different branch thicknesses for different generations – which may or may not match the choice of thicknesses shown above.  

5  | F:  
|   | • Clicking “Setup” and “Go” with the setting shown in Fig 1, produces a treelike pattern with at least 3 different branch colors for different generations – which may or may not match the choice of colors shown above.  

2  | G:  
|   | • (Extra Credit) Create a procedure that passes in two parameters sz and colr. The procedure should allow you to create a turtle of with a given size and color. You will call the procedure with code similar to this:  

```
create-my-turtle 3 red.
```

Make a button on your interface to call this procedure!