

Project8

In this project, we continued working on the same material we began last week in Project 7. Once again, we use our knowledge of L-Systems to combine our `lssystem.py` code and `turtle_interpreter.py` code to make trees and fractals. In this weeks proect, we are building off of this idea. To do this, we do things like use multi-rule L-Systems, add leaves/berries/etc to the branches of our trees, edit the colors on the tree, write our own LSystem text code, and create a scene with varying trees of different colors, sizes, iterations and angles. Overall, project 8 is helping us become more proficient with LSystem coding.

In **task 1**, we were asked to add new elements to our `drawString` method in our `TurtleInterpreter` class. The purpose of this task was to be able to change the colors of leaves/berries/etc without changing the color of the along with it branches. To do this, I just had to add a number of `elif` statements to my `drawString` method that supported the possibility of new characters. Task 1 asked for 5 new characters: '`<`' for storing color, '`>`' for restoring color, and '`g`' '`y`' and '`r`' for three different leaf colors. Here is a snippet of my code for task 1:

```
elif c == '<':
    color = turtle.color()[0]
    cstack.append(color)
elif c == '>':
    turtle.color(cstack.pop())
elif c == 'g':
    #makes the leaf color medium orchid
    turtle.color("Medium Orchid")
elif c == 'y':
    #makes the leaf color turquoise
    turtle.color("Turquoise")
elif c == 'r':
    #makes the leaf color salmon
    turtle.color("Salmon")
```

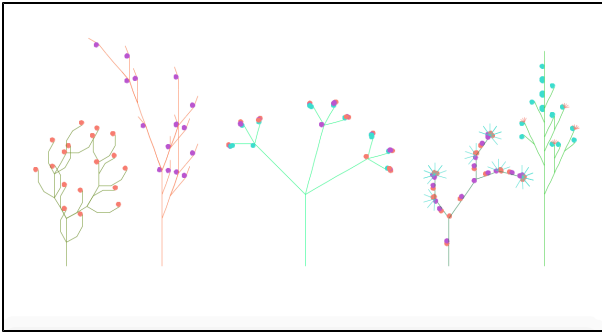
Task 2 asked me to create a new file called `arrangement.py` that imported my `lssystem.py` and `turtle_interpreter.py` files. The purpose of this file was to write a function that makes a collection of trees with different leaf colors, and branch colors. I decided to use all 5 of the given lsystems on the project 8 page, but I edited them to have different colors and different leaf colors. To do this, I just made minor edits to the given `Lsystem.txt` by adding my own colors. Here is a snippet of my code from my `arrangement.py` file:

```
# Tree 1 (systemCL.txt)
turtle.pencolor('OliveDrab')
'''tree with stem color of olivedrab'''
turtle.up()
turtle.setposition(-500,-200)
turtle.setheading(90)
turtle.down()
ti.drawString(s1,25,angle)

# Tree 2 (systemDL.txt)
turtle.pencolor('Coral')
'''tree with stem color of coral'''
turtle.up()
turtle.setposition(-300,-200)
turtle.setheading(90)
turtle.down()
ti.drawString(s2,25,angle2)
```

Here is the image my `arrangement.py` file created:

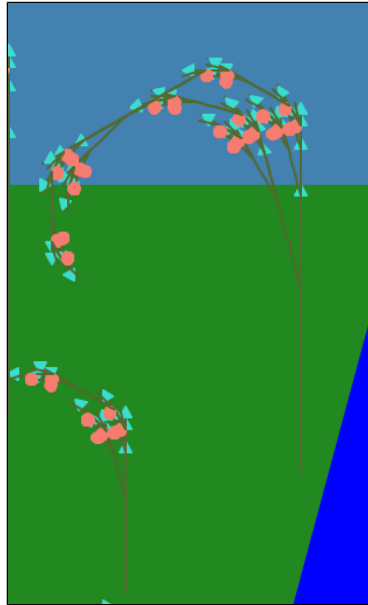
Required Image 1:



Task 3 required me to create my own LSystem text files. To do this, I used two of the given lsystem.txt files as templates and changed different parts to create my own LSystems. Here is what my mylsystem1.txt looked like compared to its original template:



original:

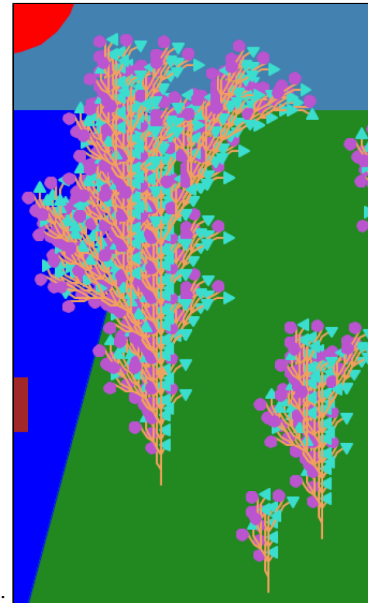


new:

Here is what my mylsystem2.txt looked like compared to its original template:



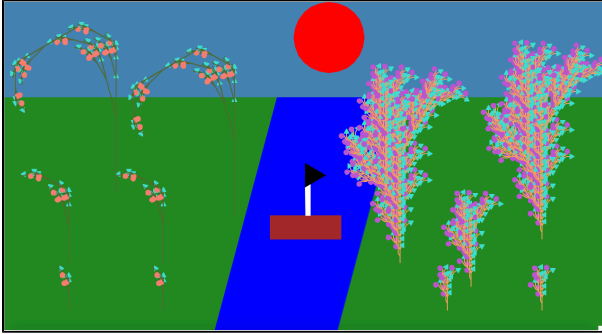
original:



new:

Then, we were asked to create a scene that incorporated our personal lsystem trees with different iterations. In a new file called growth.py, I created a scene with a river and my two different lsystem trees on opposite sides of the river. The trees were of iterations 2, 3, and 4. Here is what my scene looked like:

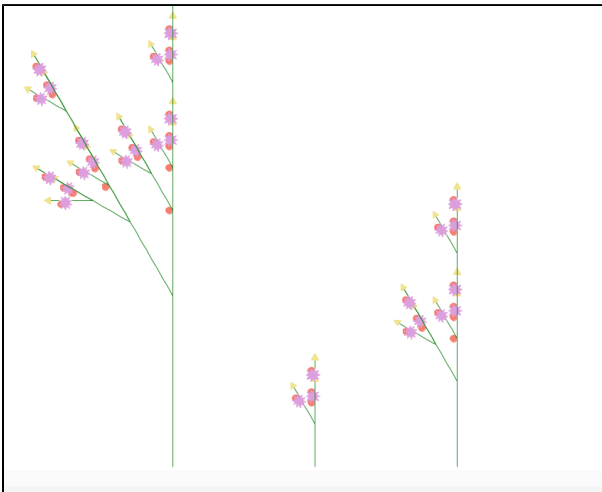
Required Image 2:



After I completed the required tasks, I worked on three extensions for this project. First off, I completed an extensions through the project by creating a new color character and a new leaf character that drew a triangle shaped leaf in my drawString method. For **extension 1**, I built off of this extension by creating a few new characters in my drawString method. I created two new color characters and one new leaf character that drew a star shaped berry. Here is a snippet of code from my first extension:

```
elif c == 'P':
    #draws a petal
    #CREATED FOR EXTENSION1
    for i in range(12):
        turtle.begin_fill()
        turtle.forward(5)
        turtle.right(108)
        turtle.forward(5)
        turtle.left(144)
        turtle.end_fill()
elif c == 'b':
    #CREATED FOR EXTENSION1
    #makes the leaf color salmon
    turtle.color("Plum")
elif c == 'o':
    #CREATED FOR EXTENSION1
    #makes the leaf color salmon
    turtle.color("Khaki")
```

Here is an image from extension 1 with different iterations:



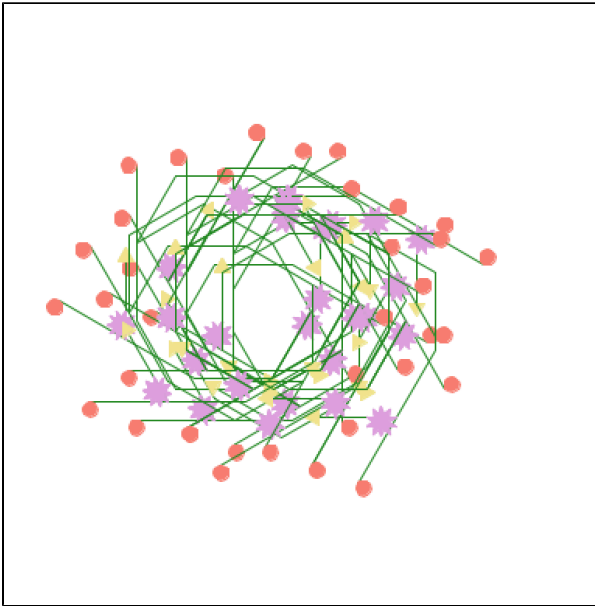
Extension 2, I wrote my own new lsystem.txt code that made a circular, bush-like tree. This file was called lsystemextension2.txt. This is what the code in that text document looked like:

```
base Z
rule Z F--F-F-([Z]F<rL>]-Z<bP>]-F<oQ>]-Z<bP>
rule F F
base Z
```

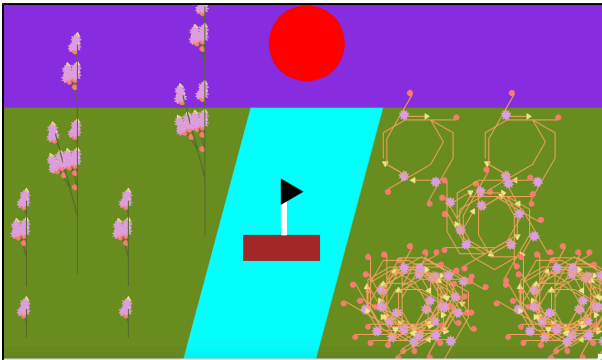
rule Z F--F-F-[[Z]F<rL>]-Z<bP>[-F<oQ>]-Z<bP>

rule F F

Here is what the resulting image from my new lsystem looked like:



And finally, for **extension 3**, I put the two previous extensions together and placed them in my scene from task 3 with the different lsystems and new colors. Here is my extension 3 image:



What I learned. I expanded my knowledge of LSystems. I was able to import LSystems in normal turtle image scenes. Also, I got some practice in writing my own lsystems. In addition, I learned how to create leaves, berries, etc and gained the ability to change the colors and shapes of specific parts of the tree. In conclusion, I became much more experienced and comfortable with LSystems.

During this project, I worked alongside fellow coder Stefan Kohli and received help from TA Mike at multiple parts during the project.