Name: Braden Maillet
Email: braden_maillet@student.uml.edu
Professor: Krishnan Seetharaman
Class: Computer Graphics (COMP.4270)

Assignment 2

This assignment submission includes a .html as well as a .js file. Together they compile within a web browser to create an interactive sierpinski gasket. This assignment is an extension of assignment 1. A lot of the code that is viewed within this assignment has been reused from that. The assignment given required us to create HTML UI input elements to adjust various aspects of the sierpinski gasket. Including but not point amount, port size and color. It also required us to adjust other UI elements based on what was happening within the program. For example when the program is running there is text that notifies the user of that. The same happens when the program is stopped.

The program that I created does everything that was required within this project. I found this Assignment not too difficult overall. It was a little bit tedious working through the HTML. I had some trouble formatting due to my lack of experience with HTML. But once everything was set up on that end things went quickly. After all of the front end stuff I worked through how to grab information from the HTML. This was rather easy. I simply used the .getElementbyid() function to pin my input fields to variables within my .js file. This meant I could simply use variable.value to find the value of any of the input fields in my program. If the input fields were empty I would default to the automatic value found in the slider. I notified the user of this with a text field at the top of my web page. In order to operate the buttons I used the addeventlistener() method and pinned them to variables that were referencing the element id. This way I could run an animate() or stop() function every time either one of the buttons was pressed.

Do any bugs remain within the program? None that I know of currently. I tested all the input fields confirming that they default to the sliders. I do wish that I was able to implement a pause button but due to time restraints, I was not able to. I did fix a bug that existed from the previous assignment. The gasket used repeatedly creates a new gasket within the render function. Now it creates a gasket every time the render function is called. This makes for a much cleaner animation. It's what I should have done in the first place.