Assignment 1 was very hard for me. I went into it with the mindset I could simply do a for loop around the number of points, increase them 5 times, then decrease them 5 times; same for changing the shape. I spent a lot of time trying to figure out what to do along with understanding what the code did. It was only in the last couple of days I was able to make significant program.

I tried looking in the book on how to change the color with every iteration, which was the hardest aspect of the assignment for me. Eventually with the light help of a varsity tutor, I was able to create a server.js file to put the shader code in. I heard in class about putting them in variables in the gasket1.js file, but I just couldn’t figure out how I would be able to change the colors.

The first solution I had was with the number of points. I simply made a random function with the help of the built-in random function to vary the number of points the Sierpinski triangle had from 600-15000. The next solution came easily, as I had used the same random function to randomize the viewport, where I had originally done so with the vertices array.

Last, was changing the colors. With the light help from the varsity tutor I had created a server.js file where I could change the colors, but I couldn’t use the random function in the glsl file, of which I had placed the shaders in from the html file. After searching for many hours on the internet to see how I could change the colors from the gasket1.js file and send the input to the glsl file, I started looking at how the position shader got its input from the gasket1.js file. After some more searching, I finally found out I had to create a uniform variable in the fragment shader, and get the uniform location of that variable in gasket1.js before I was able to use the random function to change the color of the triangle.

With this, I believe the second assignment will be much easier to accomplish.

NOTE: To run the program, cd into the folder and do: **node server.js**, then in a web browser, type **localhost**.

SAMPLE OUTPUT:

