Ethan Kittell
COMP.5460
3/30/25

Program 5 Report

I didn't face many issues while completing this assignment. Most supplied code did the heavy lifting of point-generation algorithms for me. The only real issue was getting the Midpoint line generation function working. I had to implement some features to this function to handle negative slope and work for all quadrants (bottom left to top right, top right to bottom left, bottom right to top left, and top left to bottom right). I also had to implement handling vertical and horizontal lines. Since sometimes they would not appear depending on the quadrant for this function. But despite all of that this was the only function that gave me issues. Implementing everything with primitives was also pretty straightforward. This is because 5 out of the 7 shapes/algorithms we were supposed to implement without primitives were already in the canvas API (so it was only about 4 to 5 lines of code each to implement with primitives). Generating the shapes with and without primitives gave the same output.

I learned a few lessons while completing this assignment. I mostly learned how to handle testing these functions and correcting them. As I mentioned previously I had to correct some issues with the midpoint line generation not working with all different methods of generation. This mainly was trying to generate the lines to and from different quadrants and also how I was generating them (horizontally or vertically). I had to do quite a bit of testing to make sure every function was working correctly so I comprehensively tested each as I did with the Midpoint line generation. I also learned to handle the UI better for this assignment. Different functions had different inputs so I learned to hide certain unused elements depending on what shape/algorithm was being generated.

There aren't any remaining bugs to report. The only thing I could consider a bug is the primitive generation. While generating a line/shape/curve with single points gives off a "pixel-perfect" look the shape primitives seem to do a little extra (more pixels drawn) when drawing the selected shape. Other than that I was unable to find any cases where a shape/line/curve couldn't be generated properly.

It is also worth noting I did add an additional feature if you could call it "additional". It was not specified how to do this part "Draw the same shapes using Canvas primitives and compare if they are identical or not; analyze and write in your report". I assumed we had to generate these with primitives just for our report maybe as a function we would call once. I added the choice to generate a shape with or without primitives on the program. This allows you to see for yourself just how similar the output is on the canvas using points or the canvas API.