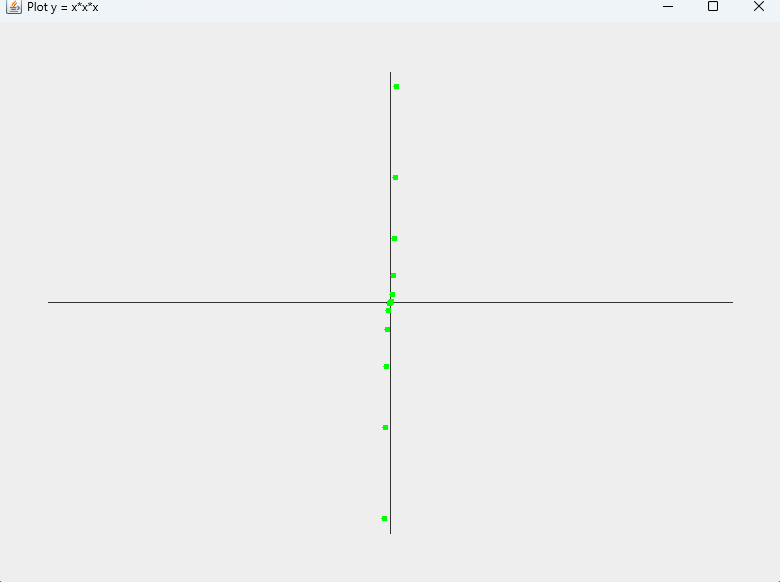
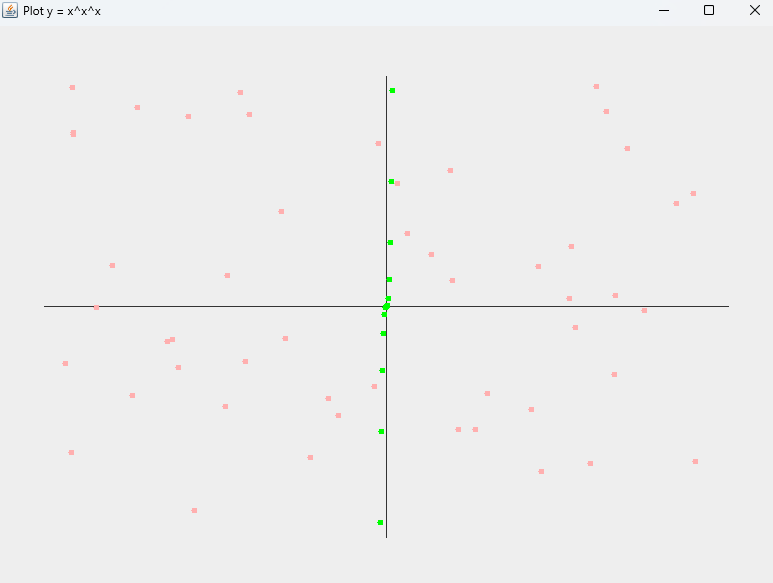
Justin Murphy

PSS 1 Write Up

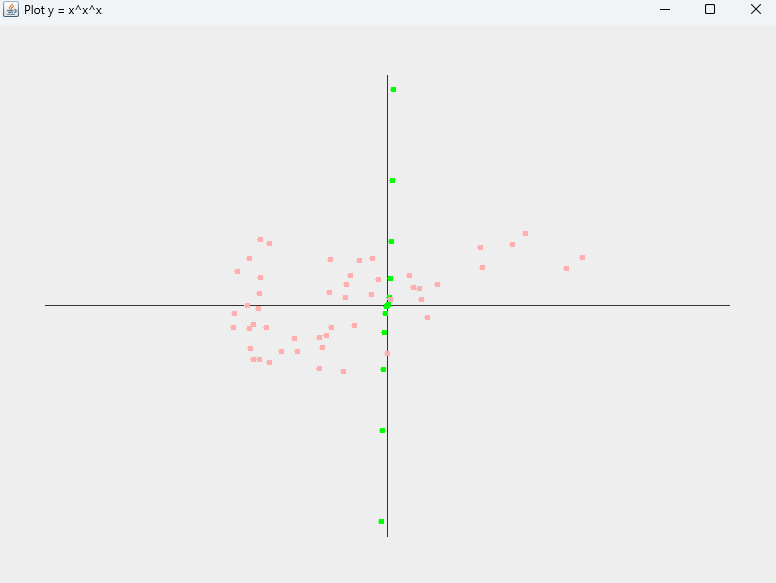
The beginning of the code has an Override that takes over the Painting Component for the plotter. It then asks for the Width and Height that is supplied in the tester, and initially draws the X and Y axes. It then sets the Color of the plot points to Green, using the setColor method and goes into making the rest of the graph by scaling everything from -350 to 350 on the graph to one. It makes the Graph convert itself to be able to be viewed as a Java output, for both X and Y, and then goes into the equation. Finally it checks to make sure the points are plotted only if they are on the screen allotted and plots the points on the finished graph.



We then add the salted points by importing the Random function, getting a random variable along the plot points that are already there, and changing the color to Pink to see the differences between the Main Function and the Salted points.



Then we add the Smoother for the data for specifically the Salted points. We add a whole new section into the Salted Points to generate the Salted points, then to Smooth the Salted Points we take the Average of the points plotted, take their index, shift their index closer to the main points of the Equation, and then make the graph entirely. Taking the average of each smoother for both X and Y, and then filling out the portion of the graph that's necessary.



This data would be a full graph of closer points, almost filling out the whole graph, if we didn’t force it to only place the points in the places we could render and see in the main portion. However, going forward I will be picking a different equation that fills up a little bit more space, since it is difficult to see how long these plot points are.