Unit 3 Status Report

Date: May 5, 2022
To: Mr. Fulk
From: Anish Lakkapragada, Hank Hsu, Abhishek Nambiar
Subject: Week 2

Accomplishments:

Abhishek and Anish made progress on exploring the code (and the Swing library to draw the tiles that are required in the grid). Anish has written some proof-of-concept code to test our basic idea of creating a tile grid that users can move around with (through keyboard input.)

Abhishek and Anish have also devised a method to create the appearance of tile selection/movement on a grid of tiles. A selected tile by the user is highlighted a certain color (blue for now), and then when the user indicates they want to move in a certain direction (through arrow keys) the current tile is toggled to be in black color (indicating unselected) and the new tile is drawn to be blue. This gives the illusion of a user moving along a grid with the keyboard.

Hank has nearly completed the core method of being able to differentiate functions in String form. The basic functionality for this has been coded so that it correctly computes and displays the derivative for most functions in string form. Functions that work so far include any polynomial (e.g. "3x^2 + 2x") ± cos(x) or sin(x). Currently Hank is working on debugging this method (one of the most tedious parts of our application.)

While Hank has done this, Anish and Abhishek have structured the classes and designed the methods for each of our classes; all available in our Project Specification.

Problems/Risks:

Because the complexity of the functions our game can handle is dependent on our ability to differentiate on them, the main challenge so far is whether we can get our String differentiator function to be fully functional for all corner cases.

Hank had to account for edge cases such as differentiating constants, recognizing x instead of x^1, getting rid of double signs (e.g. sin(x) - cos(x) should return cos(x) + sin(x), NOT cos(x) - - sin(x), + - should be simplified to -). We are sure we will discover more of these edge cases (and will need to solve them) as we continue to work farther.

Another minor risk is the fact that in Anish's proof-of-concept code, changing the color of a current tile could lead to slightly more blotched and thicker text than before. We think this could be because we are putting another line (now in a different color) on top of a pre-existing one.

Next Steps:

Now that we have figured out most of the calculation and differentiation-related aspects of our project, next week Hank, Abhishek, and Anish all plan to divide the classes and write their methods respectively. It is likely that Anish will be working on the *Game* and *TileManager* class while Abhishek works on the *Tile* and *FunctionsList* class. We plan to only beautify our application with fonts/styling after the bare application is fully functional.

Next week, Hank also plans to account for coefficients in front of sin(x) and cos(x) functions (e.g. "5sin(x)") and inside them (e.g. sin(5x)).