GUI Graph Plot Project

Mark Daniels and Ryne Emanuel and Colby Howard

Approved by: Laurel Hilger(LA)

Colby Approved by Professor Lanik

The purpose of this project was to use GUIs to plot a set of malleable set of data with min and max values at different colors and to create a repository using GitHub that will allow the grader to access and run our code. Following the guidelines stated as the default option on the second page of the requirements, we split our code into 3 different functions that had three different purpose but when run together, created one desirable output. It is important to note that the phase "global gui" appears as the first line under each function to ensure the gui variable being used in each part is consistent throughout.

The purpose of the first function was to create an empty plot that would eventually receive data points from the second function. An empty plot is established in the first part of the function using the "gui.fig" and "gui.plot" functions. As stated previously, the global gui command ensures that any function beginning of containing the "gui" variable uses/is altered by/plots the same value given to the gui variable. The second part of the first function generates a format for the values from function 2(F2) and gives them a position on the plot—as indicated by the numbers between the "[]" symbols inside the parentheses containing the words in pink. This second part is similarly repeated for the third but instead of referring to the values of F2, it references the min and max values determined in F2—the process in which will be discussed later. The fourth aspect creates a button that, when combined with the text boxes created by the sixth part, allows for random values to be plotted. Unlike the other five parts, the fifth part is not designed for any sort of functionality, but rather it gives the line created by the plotted values a color.

Function 2 was created to give values to function 1 to plot. The first part of the code took the inputted "x" and "y" strings and turned them into arrays. This process was repeated for the min and max values for both "x" and "y," but instead of turning these values into arrays, they were turned into numbers using the "string2num" command. The third part of the function is comprised of 4 if statements establishing the default "x" and "y" min and max values if no values are inputted into the textboxes. The last part of the function is comprised of one large if statement with two other if statements inside. This part gives the line created by the values a color, checks to see if the max values are greater than the min values, and establishes an error message to be displayed if the input strings are entered incorrectly.

While function 1 and 2 work hand in hand with each other, function 3 performs its function only when the operator decides it to be necessary. To put it simply, the third function creates a reset button that clears the values inputted in the text boxes and resets the min and max values to their default values.

The most difficult parts of the project were understanding the overall format of the MATLAB code, understanding the incorporation and syntax of GitHub, and adding textboxes to the buttons. While understanding what was desired for the project was confusing at first, once we understood what we were to do, the coding part became much simpler. We realized that this was essentially just a version of the vending machine with more parts to it. This allowed us to write the code but did not completely help us understand how to create the textboxes. We figured out how to correctly write the code for the textboxes by referring to MathWorks and experimenting with different ways of formatting the code until it performed the actions that we wanted. A great example of how GUIs are used today and how they are constantly being improved is the Tesla self-driving cars. GUIs must be used to tell the car which routes to follow, how fast to go, and how to react in certain situations. However, like many things, nothing is perfect. A GUI cannot account for every possible outcome in this case thus resulting in

Tesla not being 100% autonomous. This process is still being refined and changed and could very well be a possibility within the very near future.