IEEE RAS @ MDC Project Setup and Description Template – *Python Graphing Calculator*

1. **Project Name:**

***Python Graphing Calculator***

1. **Project Topic/ Keywords**

Programming, Python, Mathematics

1. **Project Objective**

This project’s objective is to program a functional and portable graphing calculator which will be able to make 2-D and 3-D graphs. The purpose of this calculator is to help students in math classes to visualize complex geometries, especially in Calculus. Consequently, this project will be donated to MDC Eduardo J. Padron math professors and math department.

1. **Project Description**

1. **Overall description/ What is it about?**

This project is a program which consists of a series of functions that can be modified by the user. Each function computes the values as a set of points of a given type of graph and displays them in the screen as a graph.

1. **Technical description/ How does/will it work?**

The basic functioning principle of this Python graphing calculator is functional programming. As previously said, each function will compute the values for a specific mathematical function (Ex: logarithm, cosine, polynomial, etc.), the parameters for each function, like degree and coefficients for example. Each function computes all the values for the given mathematical function in a user defined domain and stores the x and y values corresponding to every point of the mathematical function’s graph into the ‘x\_val’ and ‘y\_val’ lists respectively and in an organized manner. Then, a library called ‘Matplotlib’ is used to plot every point into a graph. This library has a function called ‘plot(x,y)’ which takes as its arguments the x and y values and plots the corresponding point. In this program, instead of giving the function a single pair of values, we give a pair of lists which correspond to the x and y values (plot(x\_val, y\_val)). Then this function plots every point- in order of their index- which is contained between the two lists. Finally, a function called ‘show()’ displays the graph as a window which contains an image with the plot area and the corresponding graph(s).

1. **Challenges/ Difficult stuff?**

Some of the current and future challenges in this program as of *6/8/2020* include:

* Identifying graph asymptotes *(Algebra)*
* Computing Derivatives and Integrals *(Calculus I)*
* Computing and displaying areas for single graph portions *(Calculus II)*
* Computing and displaying areas for multiple graphs (shared areas) portions *(Calculus II)*
* Computing and displaying 3-D graphs *(Calculus III)*
* Improved user interface using Tkinter library
* Improved validation system

**\**Notice****: For a more detailed list and description of challenges and problems encountered in this project in its current stage, please refer to the “To Improve List and Notes” section of the program.*

1. **Stages- Categories breakdown / Electrical, Mechanical, Manufacturing, Software, etc.**

The categories for this program mainly include ‘*Software*’ since this project is mainly focused in programming. However, if the project gets very complex in math-related stuff a potential category for this project would be ‘*Mathematics*’. The Software category oversees translating the math as we humans know it into an algorithm that the computer is able to run it in order to make the computations. The Mathematics category oversees the research for effective mathematical properties, definitions and tools which can be efficiently converted into code. For example, it is hard to make a computer integrate using integration methods and rules, but in the other hand it is easy to make a computer integrate by adding up very small rectangles under a graph.

1. **Project Budget**

As of *6/8/2020* this project has a net cost of $0. The libraries and programming resources we are using are open source and thus free for now.

1. **Project as it is to date (Files, code, designs, notes, helpful information)**

The project is contained in a Python file (.py) and is included with this simple documentation in the same folder. Helpful links include:

* [*Python Package Index*](https://pypi.org/)
* [*Python Language Reference*](https://docs.python.org/3/reference/)
* [*Matplotlib Documentation*](https://matplotlib.org/)
* [*NumPy Documentation*](https://numpy.org/)
* [*Python ‘math’ Library*](https://docs.python.org/3/library/math.html)
* [*Tkinter Library Documentation*](https://docs.python.org/3/library/tk.html)

1. **Project Deadline**

The deadline for this project is scheduled to be by the *end of Summer 2020* or by *September 1st , 2020*