Criterion A: Planning

Problem Statement

My client has difficulty teaching mathematical graphing transformations and would like a way to help his students visualize them¹.

Description of Scenario

The graphing programs my Client, Mr. Macky, uses do not have the desired features that would allow him to show graphing transformations well because new numbers need to be entered every time the graph changes and it is very inefficient. To actually show the transformation it needs to be smooth. Mr. Macky would use this to help him teach his class².

Rationale for Proposed Product

A specialized computer program made by myself will work well for the situation as it can save time and automate a process for my client which will allow his students to learn better. A GUI Java application works perfectly as I can display information but also have features like a slider for the number input that my client specifically requested³. I will use Java because I am familiar with it, it is free, it has a well built development environment, and it can be used on many platforms.

Success Criteria

What will the program do?

The user will be able to enter and graph their own functions (linear, trigonometric, exponential, logarithmic, random automatically connected points, etc.). The user will then be able to mathematically transform (stretch, shrink, reflect, etc.) the graph through variables inputted through sliders and see these transformations as they occur with automatic refreshing of the graph⁴.

¹ Cameron Macky, initial interview by author, Bangkok, November 18, 2019, audio recording 10:00, Appendix 1

² Cameron Macky, initial interview by author, Bangkok, November 18, 2019, audio recording 10:00, Appendix 1

³ Cameron Macky, initial interview by author, Bangkok, November 18, 2019, audio recording 11:00, Appendix 1

⁴ Cameron Macky and Author, Annotated Prototype, Bangkok, December 6, 2019, Appendix 3

User Friendly Features

It will also have various other features such as having a table display all of the points, searching for points, zooming on the graph, displaying multiple graphs at once (such as pre and post transformation). The user may also change the graph window, point density (the amount of points calculated changing the quality of the graph), and each functions color. All of these features along with the sliders and help notes will all be user friendly⁵.

Error/Exception Handling

The program will only calculate the points within the window to avoid all situations where a function is approaching infinity and be able account for asymptotes or places without points. It will also check for various mathematical errors such as dividing by zero⁶.

Program Efficiency

The program will be simpler, more streamlined, and more efficient to use compared to similar graphing programs. It will do this by automatically refreshing the graph and avoiding all possible unnecessary calculations. It will also have a feature where point density can be adjusted to change the amount of points being calculated which can be lowered to make the program run faster but may decrease the quality of the graph.

Word Count: 435

⁵ Cameron Macky and Author, Annotated Prototype, Bangkok, December 6, 2019, Appendix 3

⁶ Cameron Macky and Author, Annotated Prototype, Bangkok, December 6, 2019, Appendix 3

Appendix 1: Initial Interview



Link to audio file recording of initial interview.

Appendix 2: Second Interview



Link to audio file recording of second interview.



Appendix 3: Annotated Prototype (Annotated by Mr. Macky and Author)