Criterion E Evaluation

Evaluation of Success Criteria

1. **Partially Met.** Although the GUI is not tedious, it is not intuitive. As my client suggested:
    - There should be visual indication of what graph tool is currently in use
    - There should be interactivity when using the straight line and curved line drawing tools
    - Objects should be draggable and not immutable
    - Using the keyboard for deleting, copying, and pasting would have made usage easier
    - The saving popup is confusing
2. **Met.** However the program is overly minimal as my client suggested, and lacks complexity. As my client suggested:
    - Preset functionality to save time should be added
    - A settings pane for more detailed graph object manipulation should be added
    - A recents panel should be added in the Welcome window
    - Setting a file name should be possible
    - Text size should be changeable
    - Undo and redo should be present
    - A checklist of requirements for a good IA Economics graph should be integrated (as this project was originally designed to assist in graph creation for Economics IAs)
    - Functionality see if those checklist points are met should be added
3. **Partially Met.** Although most Economics graphs can be created, some key graphs such as the business cycle due to the graph view editor being a square and in the shape of a sine wave, or the production possibilities curve cannot be created due to their circular shape.
    - A method of drawing more circular curves with ease was suggested, as although the curved line drawing tool was able to get close, it is hard and tedious to use
4. **Partially Met.** Designed graphs are able to be exported as a TIFF image file, but these images are not clear as the client had stated.
    - The resolution of the images created must be increased as suggested by the client
5. **Met.** During personal usage, and with the client, the program did not crash, or break, or act unexpectedly (providing that an understanding of the idiosyncrasies program are known, due to a lack of visual indicators as stated by the client).

Future Development

Future development should be predicated on the feedback from the client. However, both the client and I recognise that this project was ambitious, and in personal reflection, too ambitious. A more realistic development timeline should have been actioned, with less necessary functionalities being sacrificed to allow for more needed usability improvements, instead of the ultimate decision to scrap any functionality that did not directly pertain to graph drawing. Extensions to the program would be built upon quality of life improvements, such as the additional functionality described above.

A key functionality missing was the lack of saving created graphs not just as image files but also in a method that would allow for created graphs to be saved as a textual form, to allow for the client to continue to edit graphs after the application was closed.

Although Swift in theory had many benefits that would have allowed for easy file handling, or encoding and decoding data to and from text and program objects, I had failed to recognise that these implementations are easy provided that the developer has an understanding of Swift protocols, which I had grasped too late.

[523 words]