Group 80

Build
Could you clone from Git and build using the README file?

I was able to clone the Git and setup the project just fine using the README file included. There was sufficient documentation to properly get the program up and running. However, documentation regarding how to use the program was limited. The documentation states that there is functionality for adding variables, constraints, etc however there is no explanation as to what each of these sections mean. I think that the variables function is fairly self explanatory. The constraints section is also rather self explanatory, although some formatting documentation may be useful. For example, from the video I noticed that you use math.sin() and math.cos() which may not be intuitive for the user (who may think that you just do sin(angle) and cos(angle). I am not sure what the optimization section does as there is no documentation on this part, some explanation could be useful.

Legibility
Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?

Most of the relevant code was in the Addin.cs file. From the file, most of the code was very easy to navigate through since the different sections were sectioned off into their own regions allowing me to simply open and close different regions. Code documentation was fairly sparse. Many of the comments were either really short or redundant. For example one function called “UnregisterAssembly” simply had the comment “Unregister” which was rather self-explanatory. I would have liked to see better commenting regarding what everything does since most of the comments were only a few words long and a sentence at most. Defining regions was a nice touch though and really helped with legibility.

Implementation
Is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?

One thing I noticed when playing with the add-in was the error handling. Many buttons, when you pressed ‘cancel’ often had an error message pop up. This seems to imply that the errors are not being handled or the handling is simply to re-print the error message. I don’t think that simply pressing cancel should lead to an error popup, especially one that states something like “Traceback File ‘optimization.py’, line 18 …” (this is the error message from pressing optimization then cancel). Simply telling the user “Could not load file” or not telling them anything when cancel is pressed seems to be a much better option. The “Add Variable” button also produces an error message when no SolidWorks Objects are loaded in. When I looked at your code, I could see that no errors are being handled regarding opening files. A quick look at line 18 of optimization.py shows that there is no try/except or anything when opening the csv

file. I would look through other parts of the code that requires the user to submit a file and maybe add some error handling code so less error messages pop up. Aside from that, the code looks fine. One minor nitpick is the rather large if/else clause in optimization, would a switch or enum be possible?

Maintainability
Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?

There were no unit tests as far as I could tell. Although not exactly a test case, an example folder was provided containing predefined variables and constraints. Looking at the program, I think that tests for different variable ranges, different constraints, perhaps edge cases with constraints (like constraints that are infeasible/tend to infinity) would make sense.

Requirements
Does the code fulfill the requirements?

The code fulfills the requirements. The optimization is able to automatically generate the optimal shape given the variables and constraints which is stated as the main goal of the project. Although the program is unable (as far as I can tell) to manage more complex objects, it is stated that this functionality is a stretch goal anyways.

Other
Are there other things that stand out that can be improved?

One thing, although minor, that I noticed is that some parts of the example and code (specifically MainUI.Designer.cs) are in Korean.