Code review & CI and Automation

This document reports on the LO5 section of the portfolio.

1. Code review and issues in the codebase

The main issue with this codebase overall is the amount of redundancy with the functions. I would like to note that I had to develop this program in high school in a short amount of time and the main concern was to have a final product that was working for real world users therefore with my little to no Python experience at the time, I unfortunately could not find the time to simplify the program to be written in a shorter amount of code. However, despite this redundancy, I think that the redundant implementations here provide more than enough examples and a template to easily implement and test a new feature for calculating a new specific formula in the future, allowing for new developers to understand what is going on in the code easily overall without complicating things and therefore allowing for easy expansions.

Although the massive amount of repetition of function implementations are present in the codebase, comments are written for each function implementation in the code and in the unit test suite to provide documentation inside the code for helping to describe what is going on with the codebase.

In addition, I would also like to mention that along with the circumstances of developing this program in the past, it is also important to note that the way that the program was written has a difficulty to write tests without copying the calculation implementations due to how the functions were implemented inside the widgets and the restrictions of the Tkinter graphical user interface library such as the mainloop() function running the graphical program interrupting calls made on separate threads, leading to a difficulty with multithreading as well.

2. Construction of an appropriate CI pipeline for the software

The four stages of constructing an appropriate CI pipeline for a software is as follows: development, build, testing and deployment [1]. The first phase of this pipeline, which is development, can be handled with Git which is widely used for version control and can ensure that the integration of the code to start the pipeline goes smoothly with each new push of code changes and merging. Then in the second phase, which is the build phase, we need to build the environment using the code that is pulled by the pipeline in the first phase. After successfully building this environment, the pipeline can initiate the third phase, which is the testing phase, by using a testing loop provided by a testing pipeline using test data, test infrastructure and test cases as well as monitoring the overall testing. By the completion of this phase, we arrive at the final phase in the pipeline, which is the deployment phase, where the software after testing can be deployed to the production environment without impacting the end user.

3. Test automation

Since the implementations of various formulas in the program share the same template overall, the implementation and testing of new features as described above can be automated to ensure a reliable and repeatable continuous integration workflow. Some aspects of testing that can be automated include unit tests for input validation and calculation correctness checks as described under the functional requirements in the range of requirements of the software.

4. CI pipeline functions demonstration

Based on the construction of an appropriate CI pipeline for the software in context, we can first see any kind of change that may negatively impact the program with the implementation of a new feature such as a new formula to calculate, which can be written code that breaks the program during runtime or an incorrect implementation of the formula for example. Then in the following build phase, we can create the environment with the graphical user interface to have an appropriate environment for simulating the user behaviour in the next phase of testing, which can be the simulation of the expected workflow of clicking the button for a specific formula, calculating the variable and returning back to main menu or the unexpected but possible scenario of entering an invalid input into a field. And finally, after successfully launching and passing the testing phase, we can enter into the deployment phase with the newly added features into the software such as a new formula that can be used to calculate a new variable.

Additional Bibliography

[1] Bigelow, S.J. (2021) *CI/CD pipelines explained: Everything you need to know: TechTarget*, *Software Quality*. Available at: https://www.techtarget.com/searchsoftwarequality/CI-CD-pipelines-explained-Everything-you-need-to-know (Accessed: 21 January 2024).