

Duy Tran

1002190361

Summary and Discussion

The experience of this project is extremely teeth-clenching stressful and intense given the time frame and my overall coding expertise or lack of. The CFG task is very tedious and mundane because after 1 or 2 methods have been completed, the knowledge to know how to finish one is already memorized. But then there were 15 more to complete which then became time consuming and a nuisance. What was felt when creating the CFG.pdf was also felt into making test cases especially for the end-to-end. One part that was truly confusing was finding input to unveil failures. Besides using the “Type of Tokens” in the project’s slides for input to try, there were just too many possibilities to consider what would fall under the Error category.

But besides mentioning the horrible time I had, experience is gained, that is for certain. Using Eclipse and Jacoco and seeing how it can show what was covered or not just made the project more interesting. If more time was available, there would be a strive to complete the coverage to 100% and find more errors to test.

If I had the chance to do it again, I would not know what I would do differently unless another efficient approach were to be presented. The main solution to the many obstacles was to lock in and spend as much time needed because this project seemed very manageable from the start, but time was the main issue. An example of a challenge was creating an end-to-end test path, it was every easy to get lost because of how difficult it was to view and track the nested test paths. And to be said once again, after creating 1 or 2 paths knowing how to create them was there and completing the rest was annoying and time consuming.

The best part of this project was using Jacoco and testing and the worst part was the test preparation. My suggestion to make a project a better experience for future students is to maybe give less line of code to evaluate but a more logical challenging project.