In this project, I learned allot about determining reachable, infectable and code would propagate to desirable outcome. In task 4, I learned how to design jUnit test cases for each individual function provided in the correlated .java files. I also learned how to handle the exceptions that was “thrown” in these test cases. I learned how to throw errors and exceptions based on the prospected failures.

The lesson I have learned from this task was to acquire the habit of designing unit test case to check the quality of the application in the initial stages of development. Unit test cases can also be utilized to debug errors found in the code and provide quality software. In task 4, the unit test were performed in a systematic manner and the process of the application was performed in a systematic manner as well.

I liked that fact that jUnit defines the structure of a test case and provides the tools to run them. I also like how it enables exception handling. I like the fact that jUnit provides a platform to display all failures with a complete description of the failure. jUnit is an essential tool to conduct software quality assessment on Java code. I do not like the fact that jUnit is not able to determine if a particular segment of the code is reachable.