COSC 603 APG

Software Testing and Maintenance

Spring 2016

William Cosulich

**Project #1 - Reengineering Legacy System**

**Task 5**

I do have to start off with saying that the last part of the this project kind of sucked and was a bit painful. I didn’t know a single bit of Fortran when starting this project. I was able to read and understand the flow chart but reading the legacy system was a bit tough for me. I know you had said that knowing Fortran was not completely necessary, but for me it helps to at least know some of the syntax of the coding language that a program is written in, for me to be able to more easily rewrite the code. It was very hard for me to figure out how the code implemented what was in the flow chart. Some of the Fortran syntax was able to very hard to figure out as some of it not longer being used in most any newer Fortran projects. For example the Arithmetic if statement kept tripping me up. I could not for the life of me figure out what the 3 comma delimited numbers after the if meant. It took a lot of searching online to finally get figure it what and that it is not a normal if statement.

Now not having the ability to run the program to see the expected output made it nearly impossible to understand if my version of the code had the correct output. I am a bit worried that the code I write might not be outputting the correct values but I have no way a confirming that what I got is the same a what the legacy code would output if I used the same inputs. Reengineering this code was probably the hardest part. I have worked with legacy code before to do maintenance and have rewrite legacy code in another language to help make it easier to maintain. However most of those times the legacy code might have been old but I was able to at least run it to see how it was supposed to function and what it was outputting. The other times we just through out the old code if is was too old and hard to keep maintaining it and used the previous requirements to start all over again building a modernized version that was more maintainable system.

After getting past trying my best to reengineer this code, creating the web based documentation using Javadocs was fairly easy once it was set up. All I had to do was click generate Javadoc and everything was done for me. in my opinion this is a very new plugin to have. If I have to write any future project in Java I think that I use to Javadocs to help. It was nice learning about a new tool that I had not known about at all.

Now as far a getting my code to have multiple commits before it was due was hard there were a few times I would commit stuff and I would not be able to see the new commits on GitHub. Also there were a lot of time when I was working on my code I didn’t have access to any internet as I would try to work on some it during the day if I had any down time at work.