Jlaja Pandora Podier

Robert Bathmann

COSC 603: Software Testing and Maintenance

Spring 2016

Project 1 Task 5 **Program Modification / Reengineering**

The first task for reengineering the Fire Danger program was to understand what it does. Although the project did not call for a line by line assessment, but to understand the overall behavior of the subroutine danger, we decided to step through each line of the code and read the paper with the flow diagram provided to get a high level understanding of what was happening within the program and what information was needed to perform the calculations. The program calculates National Fire Danger Rating Indexes, fuel moisture, buildup index, drying factor, fire-load index, and various spread indexes. The program’s first step was to determine if there was snow on the ground, and depending on the response the program executed a specific path. To approach the reengineering of the Fire Danger program, we decided that we needed to break the program into components based upon what are known constants and the inputs required to be entered in by the user.

As a team we decided to frame the project by first identifying the variables that needed to be defined and what type of variables they would be defined as. We determined that we would develop two (2) classes. We defined two (2) classes, FireDanger and ForestConditions, in which the FireDanger class calculates and represents the computed National Fire Danger Rating Indexes and the Forest Conditions class encapsulates the input needed to compute National Fire Danger Rating Indexes. The ForestConditions class also has a static method that creates an instance of ForestConditions from prompts and user input.

We each had our own software development environment with configuration of Eclipse for Java EE Developers, GitHub project, and GitHub Repository set up on our personal computers, but since we were submitting the project as a team we decided to utilize one GitHub repository for submission of the project. Jlaja began to experience issues with her software development environment in which her GitHub and her Java Eclipse were not properly syncing after numerous tries of trying to point the workspaces to each other. After numerous attempts, she decided to uninstall Eclipse Java and reinstall to resolve the issues she was facing. While troubleshooting her issues, the initial framework as discussed committed to the GitHub repository. With the initial commit, we ensured that we both were added as contributors to the repository to be able to share code together.

We encountered some challenges and learned about some pitfalls when translating older programs. The portion of the original Fortran program that calculates the timber and grass spread indexes is pure spaghetti. Simply replicating that behavior was not going to work, so we created separate methods for calculating each as they are not closely related. We also found a problem with our translated program when it encountered an array index bounds exception during runtime. We assumed Fortran uses zero-based array indexes like Java, but Fortran array indexes begin at on. We corrected the program to account for the discrepancy.