Documenting Your Code for CS410J

Documentation is an essential part of software development. A well documented and well thought-out design makes programming easier and reduces the possibility of code rewrites. Good documentation makes code easier to maintain, extend, and reuse.

In CS410J we will be seeing, designing, and writing a lot of code. In order to get the most out of this code, we are going to use two mechanism to document it.

“Plan of Attack”

Starting with your Project 1, you will be required to submit a “plan of attack” (POA) for each project. A POA is a concise design document for your project. It should be in text format and should consist of 2-4 paragraphs (not code!) that describe how you will implement your project. The POA is worth one point of your project grade.

The POA for a project is due 72 hours (3 days) before the project is due (regardless of any late days you may use) and should be emailed to me (whitlock@cs.pdx.edu) with the letters “POA” in the subject. Note that I filter out all of the POAs into their own mail folder. As a result, if you have an urgent email for me, you shouldn’t put “POA” in the subject. You will not receive credit for a late POA.

Of course, the goal of the POA is to force you to think (and write!) about your project before doing it and to prevent you from starting your project at the last minute. Bwaah ha ha!

Javadoc Comments and the README

Writing well-documented code is an extremely important skill and one that requires discipline and practice. Luckily, there is a standard convention, Javadoc, for documenting Java source code. You are required to have a Javadoc comment for every class and method that you write. I reserve the right to deduct up to one point from an assignment that I deem is not well-documented. A Javadoc comment should describe the functionality of each class and method. In the case of methods the @param, @return, @throws tags should be used when appropriate. More information about Javadoc can be found at:

http://www.oracle.com/technetwork/java/javase/documentation/index-jsp-135444.html

To ensure that your code is readable by others², your code should adhere to the “standard” Java coding conventions described at:

https://google.github.io/styleguide/javaguide.html

It is important to be able to put the functionality of your software into words. To that end, each of your projects has a −README switch that should print a brief, concise, and nicely-formatted³ textual description of what your project does. The README should include your name as well as the name of the assignment and should be more than just the command line usage. When the −README switch is specified, your program should not do any work (including error checking) other than printing the README⁴.

¹Not Microsoft Word, not HTML, ...
²And by “others” I mean “me”
³Not just one big line of text!
⁴Note that your README must be “hard coded” into your program. It cannot reside in a text file that is read by your program.