Drew Smith
CEN4021
February 24, 2016
Assignment 3

Assignment 3 - Code Smells

According to inFusion, the io-commons java code directory has a 5.4 Quality Deficit Index. The directory also has 55,302 lines of code, 61 design flaws, and 57 flawed entities. After exploring around for a while, I decided to choose a random smell for each type to get a little "smell" of everything.

1. Type: Complexity
Function Name: testMultiThreadCancel() : void
Explanation: The function has a Quality Deficit ranking of 2.4. The function mostly consists of 3 try-blocks, and 5 catch blocks. The reason for the smell is because the function has duplication with a method from an external class. The external duplication impact consists of mostly Complexity but also involves Encapsulation and Coupling. Other attributes to the smell consist of a long code length and weak common attribute access. After analyzing the code, I've concluded that I do agree with this smell as methods should avoid being duplicated within a program.

2. Type: Encapsulation
Class Name: OrFileFilterTestCase
Explanation: The class has a Quality Deficit Ranking of 15 in the Encapsulation category but also suffers greatly in every other category with a quality deficit average of 8. For Encapsulation, the class contains many accessor methods and exposes a significant amount of data in its public interface via public attributes. The class consists of 11 different methods that are used mostly for accessing the results of data in other classes. I do agree with the smell of this class as the class is made mostly of just accessor methods to gather data from a variety of other classes.

3. Type: Coupling
Function Name: doNormalize(String filename, char separator, Boolean keepSeparator): String
Explanation: The method is excessively large and complex, as it contains a high cyclomatic complexity and nesting level. The method carries a Coupling Quality Deficit of 5. I do not agree with the smell of this internal method as it mostly contains a series of for loops that iterates through the data to simply perform normalization. Breaking up the method to separate, smaller methods would only make the class more complex and ambiguous.

4. Type: Inheritance
Function Name: setup() : void
Explanation: The Quality Deficit Index of this method is 8. The method is mostly affected by Sibling Duplication, which means that the method has duplication with a method from the inheritance hierarchy. The method uses the base class extensively and in the leaf class, the method is being used by other derived classes and also derives overriding. The method appears in 3 out of the 36 classes. After analyzing how the method is used in each, I agree that the function uses poor utilization of inheritance.

5. Type: Cohesion
Function Name: teardown() : void
Explanation: The method has a Quality Deficit of 2.4 as it heavily uses attributes from one or more external classes, directly or via accessor operations. Furthermore, in accessing external data, the method is intensively using data from at least one external capsule. The function is affected by Feature Envy and is weakly connected to its containing class in terms of data usage. All the function does is to reset a class but also contains a block of code that supports the resetting of instances that may be removed. I think the quality ranking of 2 suites the smell of the function as there is not much too complain about.