

Ryne Bell

CSSE376

Lab8

1. Number of Parameters is out of range at a maximum value of 7, where 5 is considered the max there should be. This occurs in the constructor for `TwitterClient`. The `run` method in `TwitterProcessor` is right on the edge with exactly 50 lines of code, and a McCabe Cyclomatic Complexity of 10, the max for each of these categories (in the default values).

2. For `TwitterClient`, it wouldn't be difficult to have the `maxFollowIdsPerCredetials`, `maxTrackKeywordsPerCredentials`, and `processForMillis` to have default values that can be changed with methods rather than all passed into the constructor. For `run` the code doesn't actually look too bad, I think the high number is just due to the calls to `Logger` with strings, so nothing needs to be done (besides, `run` isn't over the limit in the first place).

3. Yes, this code meets the Heuristic, the maximum Cyclomatic Complexity is 10, in the `run` method, which is below the 15 required to meet the given Heuristic.

4. The `backOff` method has a cyclomatic complexity of 3, looking at the if statements, three test cases would be to have `backOffMillis` equal to 0, equal to `capMillis`, and greater than `capMillis`. This tests all 3 of the possible independent paths.

5. Afferent Coupling is the number of other packages which have this package as a dependency, it is a measure of the responsibility of this package, for example, `com.gist.twitter` is depended on by example, so it has Afferent Coupling of 1, example has 0 as no other packages depend on it. A high value indicates high responsibility.

Efferent Coupling is a measure of the number of packages that classes within a package depend upon, it is a measure of the independence of a given package, a low value indicates high independence.

Both of these metrics can be used to ensure no package has too much responsibility, nor that one is overly independent of the rest of the project to the point that is not necessary

$$6. \text{ found/existing} * 100\% = (53+753)/(154+928) * 100\% = 74.5\%$$

$$7. (3+65+693)/(154+928-806+948) * 100\% = 62.2\%$$

$$8. 1 - \% \text{ found in field} = 1 - 126/3526 = 96.4\%$$