

11 Code Quality

Programs are written to be read by others, (and, by the way, they run on computers). How can we measure whether a program is well written, whether it is readable, well tested?

When the program is well written it does not contain convoluted complicated code that is hard to understand. That means that the programmer, to the best of their ability, followed our *one task — one method* rule. When the program is structured that way, it is easy to design tests for all helper methods and make sure that all possible paths through the program are covered by tests. Tools that measure **test coverage** verify this.

Other measure of code quality is based on the analysis of the source code, looking at the complexity of possible paths through the program. We will use one such tool in this lab.

The *Metrics* tool is designed as a *plugin* for Eclipse. The programmer can select to compute the code quality metrics for a specific project and the *Metrics plugin* will examine the source code and provide a comprehensive report.

Measures of code quality

Some of the values the *Metrics* computes are the following values. In most cases it reports the maximum, the average, and allows you to see the values for every class in your project.

- Number of attributes: how many fields have you defined in the class
- Number of classes
- Number of interfaces
- Number of lines of code per method
- Nested block depth
- Total lines of code
- McCabe Cyclomatic Complexity

We find the last one to be the most revealing piece of information for our student's code. It measures the complexity of the most poorly designed method and will hone in on it. The complexity of the method increases with

the addition of each new path through the code that needs to be tested. We will not explain the details here, but will show examples of code where the index value exceeds 10. What we see is that these methods are very complex and nearly impossible to test.

We encourage you to read more about the McCabe Cyclomatic Complexity, think about the code quality, and use tools like this to pinpoint the problems in student code.