Project Selection

To achieve a representative sample of projects, we chose a set of ``notification-heavy" and ``notification-light" projects to analyze; we will refer to these project sets as ``heavy" and ``light" respectively. Heavy projects got points taken off for having FindBugs notifications left in their code; light projects received no point deduction. We assumed that if students left FindBugs notifications, they would be more likely to have priority notifications from the compiler and EclEmma.

We analyzed 18 heavy projects and 20 light projects. Only 18 projects submitted by consenting students had points taken off for priority FindBugs notifications so we could only select those. We randomly selected the 20 light projects due to the abundance of projects that fit into this set.

Although students could have submissions in both light and heavy data sets, student could not show up twice in one data set. To show that our data sets can be viewed as representative samples of the lower (light) and upper (heavy) level performers in the course, we compared our groupings with students' project grades. With the exception of a few outliers, project grades supported our decisions; the spread was 22 points, or two letter grades.

Analyzing Projects

Extracting data from the students' projects took approximately 10 person--hours. The professor provided code to the students to get them started on each project, which we will call ``base code"; we did not include notifications found in the base code. The students only had to write tests cases for their code, excluding the GUI classes and interfaces, so we did not analyze the base code using EclEmma. We also did not include notifications pertaining to concepts not covered in the course, which the professor told students they did not have to resolve.