CodeScore: Automating Checks for Code Quality

Project Summary
Code smells are easily recognized design weaknesses that may indicate more significant problems within a system. CodeScore recognizes code smells using a set of specific rules. It then uses these results to compute a CodeScore—a single value that reflects the maintainability and understandability of a piece of software.

Motivation
The technology industry lacks automated tools for evaluating software quality. Such tools would be helpful for individuals who desire to improve, recruiters who seek top programmers, and educators who assess student performance.

System Architecture

Eclipse AST

CodeScore Calculation

\[ \text{Score} = \sum_{i=1}^{n} \frac{100 \cdot w_i}{1 - \frac{\text{detector} \cdot \text{loc} \cdot \text{pc}}{n}} \]

Detectors
- Comment to code ratio
- Method chaining
- Absence of Javadocs
- Method length
- Line Length
- Hard coding
- Nested conditionals
- Number of parameters
- Catch clauses

Sample Report Page with pc = .2

Project BadCode
CodeScore: 74

Results
The algorithm accurately detects code smells and presents its findings to the user. Through rigorous testing, it was determined that 8 of 10 detectors found at least 80% of the errors with time linear to the number of total lines in the input files.

Accuracy by Detector

CodeScore Runtime