Algorithm Design

Our design consists of three main parts: Recording behaviors, generating feedback, and evaluating feedback. Before the steps can be discussed, the features used need to be given.

1 Features Used

To allow for comparison between programs, seven different features were selected to be used. The features are as follows:

- Edit Distance: number of operations to change one trace files feedback to match another trace file
- Line Distance: difference in number of lines of source code
- Method Distance: count of different calls to Java List methods(add, get, remove, insert)
- Exit Status Distance: difference between operating system exit codes
- Exception Type Distance: difference between Java exception types
- Program Output Distance: Difference between output of programs
- Timeout Distance: Difference between timeout value of programs

2 Recording Behavior

- Compile all reference and submitted programs
- For each compiled class file we then instrument to record the features that are used
- All submitted and reference programs are then ran for each test case and trace files are recorded
- For all programs all features we use are also extracted during runtime

3 Generate Feedback

- For every test case, feature vectors are constructed comparing each submitted program to each reference program
- To aggregate over multiple testcases, each feature vector is then summed to give one final feature vector that compares a submitted to a reference program
- The magnitude of each feature vector is then taken to give a final distance value between a submitted and reference program
- Each submitted program is then assigned the feedback of it's closest reference program.

4 Evaluate Feedback

- For each submitted program check to see if the feedback assigned to it matches our manually assigned feedback.
- If the feedback does not match, try and identify what went wrong and why