Automatic Program Feedback

Team Members:

Andre Leone          Ryan Hartman          Calvin Winget
aleone2013@my.fit.edu    rhartman2014@my.fit.edu    cwinget2014@my.fit.edu

Faculty sponsors:

Dr. Philip Chan       Dr. Ryan Stansifer
pkc@cs.fit.edu       ryan@cs.fit.edu

Goal

The overall goal for this project is to investigate the feasibility of creating a tool that instructors can use as a tool with which they can provide students with better feedback for their programming assignments. Currently, only simple feedback can be given automatically. This takes the form of letting students know whether they have passed test cases that check if the student’s output matches the expected output for the test case. For more detailed feedback, the instructor or a GSA needs to manually examine the programs to let students know which portion, if any, of their program was incorrect and the potential fixes for this bug. This project seeks to create a tool which can provide some automated feedback to let the students know what bugs were in the program and how they can fix them.

Motivation

The motivation for this project is twofold. First, with the growing number of students continuing to higher education, and the growing trend of people taking online courses, it is becoming harder for professors to grade and provide feedback on all student assignments. This also means that the feedback students receive often isn’t much more than what test cases they have passed. By investigating a tool of this sort, professors would be able to grade and provide feedback much easier to students, and students would benefit from the increase in feedback they would receive.

Approach

The approach for this project consists of two parts. The first part of the project is dynamic analysis of the student’s programs. Here, we will instrument the students program and record information from them that describe the programs execution. The student programs will be analyzed
dynamically, meaning that different inputs will be fed into the student program and the same inputs will be fed into a reference program. Features, such as get and set operations, will be recorded while the student program and reference program are running. The recorded features will be provided to the second part of the program. This part will use data mining techniques to try and identify where student’s programs went wrong and what the possible bug is that caused their program to be incorrect.

Novel Features
The project can be considered as novel since no tool currently exists that can provide automated feedback to student’s programs.

Technical Challenges
The project itself is a technical challenge. Since no tool currently exists to fill this space, and a lot of the papers published that are related to this topic deal mainly with static analysis, much research will need to be done to attempt to create the best possible tool.

With feature selection, we need to learn the tools that can instrument Java programs so that features can be extracted from them to be used in the data mining portion.

With the data mining, we need to learn and read about methods for computing program similarity, and read to see if there is any related work in the field too help provide us a starting point for the project.

Milestones
Milestone 1
- Investigate what packages are available for instrumenting Java programs
- Become familiar with the work already done by Dr. Stansifer and Dr. Chan
- Research current papers in the field to see what work is currently being done both with dynamic analysis, data mining of students programs, and program similarity.
- Design initial algorithm/system in document form
- System will consist of three main portions
  - Recording behaviors
  - Generating feedback
  - Evaluating performance feedback

Milestone 2
- Build version 1 of the feedback program
- Evaluate
- Analyze incorrect feedback

Milestone 3
- Build version 2 of the feedback program based on the analysis of version 1
- Keep iterating through the feedback program versions until the feedback analysis is yielding acceptable results
Task Matrix – Milestone 1

<table>
<thead>
<tr>
<th>Task</th>
<th>Andre Leone</th>
<th>Ryan Hartman</th>
<th>Calvin Winget</th>
</tr>
</thead>
<tbody>
<tr>
<td>Research Existing Tools</td>
<td>33%</td>
<td>33%</td>
<td>33%</td>
</tr>
<tr>
<td>Read/Understand Dr. Stansifer’s Code</td>
<td>33%</td>
<td>33%</td>
<td>33%</td>
</tr>
<tr>
<td>Design Initial Algorithm</td>
<td>33%</td>
<td>33%</td>
<td>33%</td>
</tr>
</tbody>
</table>

“I have discussed with the team and approve this project plan. I will evaluate the progress and assign a grade for each of the three milestones.”

Signature: _____________________________ Date: ________