CS 1632 - DELIVERABLE 3:

Systems Testing A Web Application

Hoodpopper

James Hahn and Ahmed Abdelsalam

\* +1 point for TYPO fix in repository (issue #1)

Summary

When writing the Selenium tests for the Ruby compiler web application, there were several issues that appeared.  For example, when evaluating output, it was not separated clearly enough.  If the user inputs “a=5”, and clicks the “Tokenize” button, the application redirects the user and outputs a wall of text, on three separate lines.  However, the problem with this is that it causes the problem of not being able to correctly identify output on a specific line, only with the text as a whole.  So, if a test were to input “a=5”, and wanted to make sure the = operator was identified as “on\_op”, and it did not care about the other two identifiers and r-values, the tester would have to create a separate method that parses the text for the desired value.

Another issue that popped up was some of the Selenium methods did not work.  For example, when findElements(By.tagName(“code”)) was called on the Parse page, it would only return text from the first <code> tag it came across, instead of returning the text from both <code> tags; this was fixed by changing it to findElements(By.tagName(“p”)).  For some odd reason, Selenium doesn’t like the <code> element, which is somewhat mysterious.  Other than these two, there were not anymore problems that emerged; testing this web application was fairly straightforward, since it was all user-facing tests that the user could see, instead of unit tests where the hidden implementation must be understood and fully tested.

A problem that could arise in the future is the tester does not have enough domain knowledge to fully test the system.  Thankfully, Professor Laboon was nice enough to give the students a bit of knowledge about how the three operations (tokenize, parse, compile) worked on the web application, but if the system became more complex in the future, and the testers did not have extensive knowledge of compiler design, it could lead to faulty testing.

Test Code: <https://github.com/kingsman142/CS1632/tree/master/Deliverable3>

Screenshot

