CMSC 315

Programming Project 1 – Matching Delimiters

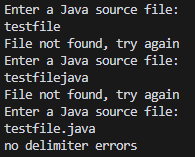
Daniel Smolsky

**Lessons learned:**

Implementing this project, I learned many new skills as well as developed and improved existing ones. The topic I learned most about was parsing text, although not part of the textbook reading this was the majority of my focus and in my opinion the hardest part. I encountered many bugs that were hard to trace and the recursive nature of some of my code was often hard to follow. I also learned how to implement stacks and hash maps, and how to manipulate data within the two structures efficiently. Finally, I further developed some of my knowledge on how to effectively utilize the Scanner class to read files and how to implement error catching code. Overall, this experience enhanced my problem-solving abilities and taught me the importance of writing clean, well-structured code.

**Test Plan:**These test cases use the TestFile.java file in the .zip

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test # | Purpose | Pos/Neg Test | Input Values | Expected Result | Pass/Fail |
| 1 | Check program requests file name input repeatedly until a valid one is entered | Positive | testfile, testfilejava, testfile.java | File not found, try again File not found, try again  No delimiter errors | pass |
| 2 | Check that a valid java program input with no delimiter errors does not return issues (testfile.java included in .zip) | Positive | Testfile.java | No delimiter errors | pass |
| 3 | Check that an invalid java program with delimiter errors has a matching output describing where the errors are | Positive | Testfile.java with the ( in line 8 removed | Incorrect closing delimiter ) detected at line number: 8 character number: 40 | pass |
| 4 | Check that delimiters within string / character literals or comments aren’t detected | Positive | Testfile.java | No delimiter errors | pass |
| 5 | Check that all delimiters are closed when the file ends | Positive | TestFile.java added a { at the very end of the file | remaining delimiter { not closed by end of file | pass |

Test plan 1:  


Test plan 2:



Test plan 3:



Test plan 4:



Test plan 5:

  
 **UML Diagram:**  
