Class: CSE4321-001 Software Testing and Maintenance

Professor: Jeff Lei

Team 04: Adalynn Cadence Everly, Mason Berry

Faults and Corrections Report

Fault: Unused Variables in get_token

In the get_token method, we originally had two variables, int i = 0; and int j;, that were declared but never actually used.

Fix: We deleted them to clean up the method and avoid confusion or bad practice warnings.

Fault: Incorrect Flag Assignments in get_token

On lines 119, string would set flag to 2.

Fix: Changed 2 to a 1.

Fault: Incorrect Flag Assignments in get_token

On lines 120, comment would set flag to 1.

Fix: changed 1 to a 2.

Fault: Faulty Character Comparison in is_char_constant

At line 293, the logic used > 2 instead of == 2

Fix: We corrected the syntax to use == properly and added the necessary logical operators for an accurate check.

Fault: Faulty Character Comparison in is_char_constant

In line 293, used || when it should use && in the conditional check

Fix: We corrected the syntax to use && properly.

Fault: Out-of-Bounds Access in is_num_constant

At line 313, the code was trying to access i + 1 in a way that could go out of bounds and crash.

Fix: We removed the +1 and just used i, which keeps it within the valid range.

Fault: String Termination Not Handled in is_str_constant

On line 340, the method wasn't correctly checking for the closing quote. It would return true even if a string wasn't actually closed.

Fix: We updated it so the function now returns false if a closing quote is missing, which correctly identifies an unterminated string.

Fault: Wrong return statements in is_identifer

This one was a bit rough. At line 364, the method returned false even when the input was a valid identifier.

At line 367, it returned true even when the input was invalid.

Fix: Changed 364 to return true.

Fault: Wrong return statements in is_identifer

At line 367, it returned true even when the input was invalid.

Fix: Changed 367 to return false.

Fault: Wrong Parenthesis Type in print_spec_symbol

At line 380, the logic used the wrong check for parentheses. The output would mislabel lparen or rparen.

Fix: We corrected the character checks so the method correctly identifies the specific symbol.

Fault: Wrong Character Check in is_spec_symbol

At line 445, the code was checking for a forward slash / instead of a single quote ', which didn't match how the print_spec_symbol logic works.

Fix: We changed the character to ' so both methods are in sync and properly detect the quote character as a special symbol.

Conclusion:

We ultimately found and corrected 11 distinct faults across the codebase. The issues we found included logic bugs, broken condition checks, character parsing problems, and a couple of unused variables that didn’t need to be there. After we made all the fixes, we ran the tests again and everything passed, which confirmed that the changes worked as expected.