

I created a fair number of extra unit tests, though all of them are negative ones. My reason in this was to increase code coverage for error cases. My 7 tests cover 7 branches that were not previously covered.

dereferenceNonStruct – tests the dereferencing of a non-struct variable in VarRValue

dereferenceNonExtantField – tests the dereferencing of a field that doesn't exist in VarRValue

arrayAccessOnNonArrayInStruct – tests array access of a non-array variable in VarRValue

arrayAccessOfNonIntPathExpr – tests array access using a non int in VarRValue

arrayAccessOfNonInt – tests array access using a non int for the first variable in the path in VarRValue

arrayMain – tests catching that [void] is not a valid main return type

unableToInfer – tests catching of usage of undeclared variables in assignStmt

Throughout this programming assignment, I wrestled with Javas object referencing appearing where I didn't expect it. I had to make sure to create a new DataType if I ever wanted to change anything in currType or anything that was a return from getFieldFromArray() or the symbolTable, lest their stored values be changed with my code.