NEED README File!

1.1 input interface: accepts user-provided file name as a parameter, as opposed to a hard-coded file name 1 Done via console args

* 1. output interface: clarity/usefulness of standard output, clarity/usefulness of alternate output to different files 2 Done via IOHelper, printing to console and specified file
  2. all errors are reported in a single stream in synchronized order, even if errors are found in different phases 2 Done via list of errors per section.

2.1.1 integers and floating point numbers (valid/invalid numbers according to assignment 1 handout) 1 Done in Lexical Analyzer, see LexicalAnalyzerTests

2.1.2 comments: inline comments, block comments, unending block comments, nested block comments 1 Needs tests for nested inline and block comments Done in Lexical Analyzer, see LexicalAnalyzerTests

2.2.1 lexical error detection: detecting all lexical errors in a program 1 Done see outlexerrors

2.2.2 lexical error reporting: accurate reporting of errors in a .outlexerrors file, including line number and useful description of the error 1 Done see outlexerrors

2.3.1 output of token stream in a .outlextokens output file 1 Done see outlextokens

3.1.1 variable declarations: int, float, class types, array, array of class types 1 Done see derivationtree/ast

3.1.2 main function 1 Done see derivationtree/ast

3.1.3 free functions 2 Done see derivationtree/ast

3.1.4 member function definitions 2 Not Done interpreted as free functions

3.1.5 class declarations: data member declarations, method declarations, inheritance list 2 Done see derivationtree/ast

3.1.6 complex expressions (all arithmetic, relational and logic operators in one expression) 2 Need to Test mostly done see derivationtree/ast

3.1.7 conditional statement, including nested if without brackets 2 Done see derivationtree/ast

3.1.8 loop statement, including nested for without brackets 2 Done see derivationtree/ast

3.1.9 read(var) / write(expression) / return(expression) statements 1 Done see derivationtree/ast

3.1.10 access to class members, including multiply nested and including array members 2 Done see derivationtree/ast

3.1.11 access to arrays: uni- and multi-dimensional, using expressions as index 2 Done see derivationtree/ast

3.2.1 syntax detection: detecting all syntax errors in a program 1 Done see outsyntaxerrors

3.2.2 syntax error reporting: accurate reporting of errors in a .outsyntaxerrors file including line number and useful description of the error 1 Done see outsyntaxerrors Double check

3.2.3 syntax error recovery: implementation of an effective syntax error recovery mechanism 2 Done see syntactic analysis freshen up on terminology and method

3.3.1 generation of an AST 3 Done but not to specifications due to not using visitor pattern

3.3.2 output a derivation of the compiled program in a .outderivation output file 2 Done

3.3.3 output the AST of the compiled program in a .outast output file 2 Done

4.1.1