

Assignment 4 – Grading Criteria

EXECUTION POINTS (40 POINTS)

- *The submitted parser source code should compile, run and produce the correct output.*

- **Test Case 1 (*simple.txt*) – 5 points**
 - This is a simple test of variable declarations and trivial assignments.
 - Each correct assignment is worth 1 point. No partial credit.
 - 1 point for correct symbol table.
 - 1 point for correct number of read/write references.

- **Test Case 2 (*expressions.txt*) – 10 points**
 - Tests the correct handling of expressions.
 - Each expression is worth 2 points.

- **Test Case 3 (*constructs.txt*) – 10 points**
 - This tests different language constructs. Partial credit is possible if the basic mechanisms like tests and jumps are in place.
 - While loop – 3 points
 - Do loop – 3 points
 - If-Else statement – 4 points

- **Test Case 4 (*errors.txt*) – 5 points**
 - The exact error handling and hence the number of errors will vary among students, but their program should not crash or go in an infinite loop.
 - 5 errors (lines 1, 3, 6, 10, and 12) and 1 warning (line 1) are generated. Each correctly identified error (exact error message is irrelevant) or warning is worth 1 point for a maximum of 5 points.

- **Test Case 5 (*mean_variance.txt*) – 10 points**
 - Test case 5 brings everything together in a program pushing the number of generated code instructions to almost 100.
 - We will not check each individual generated code instruction, but will look for the overall impression of the output.
 - Partial Credit based on how much of the input file is parsed.

IMPLEMENTATION DETAILS (50 POINTS)

- *We are looking for correct functionality (at least conceptually). The actual structure or naming of the functions is irrelevant.*
- Lexical Analyzer. Tokens are identified in source file and transmitted to parser – **10 points**
- Symbol Table exists and keeps track of array sizes and read/write references – **10 points**
- Correct expression handling (+, -, *, /, and parentheses). The student should show some functional structure that tries to handle associativity and precedence of the operators. One example is the structure with “expression”, “term” and “factor” functions explained in class slides. – **10 points**
- Error Detection – Syntax and parsing errors are identified. – **5 points**
- Error Recovery. When syntax / semantic errors are found, a minimal amount of error recovery should occur, e.g. looking for synchronizing tokens or jumping to next line etc. – **5 points**
- Tries to handle do / while loops – **5 points**
- Handles if/then/else constructs – **5 points**

MISCELLANEOUS POINTS (10 POINTS)

- Code formatting (comments, correct indentation, variable names, etc.) – **5 points**
- Output format roughly follows reference – **5 points**

NOTES:

- If you submitted your assignment late, each day past the original submission time results in a 5% deduction from the final grade, up to a 20% deduction when you submitted between 72 – 96 hours late.
- Should your program fail to compile or crash, we will spend up to 10 minutes trying to fix the problem in your code. If only a small fix is required 2 – 3 points are taken off, while a more extensive fix leads to a 15 point deduction. If we cannot find the problem, you will lose all execution points.