CSCE 434 Spring 2021

Programming Project #3: Lexical Analysis and Symbol Table

This part of the programming project is to implement the lexical analyzer and symbol table for the source language ZiNC. For the lexical specification, we will start with the ZiNC language given on the eCampus. I might (but probably won't) make modifications to this specification as the semester progresses. Note that the lexical specification includes integers, but not any floating point, doubles, etc. You will also implement a symbol table.

For the lexical analyzer, you are to use flex (or some version of lex).

Design of your lexical analyzer:

1. Define the character set.
2. Design the token types, values and actions.

Implement the lexical analyzer:

3. Remove comments and white space
4. Keep track of line numbers
5. Set up a framework for error reporting and report lexical errors, deciding whether the lexer will report all errors or will give error reporting information to other parts of the compiler.
6. Define a regular expression for each token type
7. Implement the actions

Test the lexical analyzer:

8. Define test cases and expected results output
9. Give testing results

Implementing the symbol table:

10. Define a method to insert identifiers to the symbol table,
11. Define a method to lookup identifiers in the symbol

Due Dates:
By **Saturday, April 3,** 11:59p.m., submit files to Gradescope. Submit as individual files, not zipped.

What to hand in:
Design and test document, can be named README, or some other obvious name, describing design decisions and test results. Program files, including code documentation.