University of Maryland University College

CMSC 430 - Theory of Language Translation
Programming Project 1

The first project involves writing the lexical analyzer with lexical error checking, and the compilation listing generator for the compiler. The specification for the lexical structure of the language is the following:

1. Comments begin with `--` and end with the end of the line.
2. White space between tokens is permitted but not required.
3. Identifiers must begin with a letter, followed by letters or digits.
4. Integer literals consist of a sequence of digits.
5. Real literals consist of a sequence of digits containing a decimal point. At least one digit must be before the decimal point.
6. Boolean literals are `true` and `false`
7. The logical operators are `not`, `and` and `or`. Each logical operator should be a separate token.
8. The relational operators are `=`, `/=`, `>`, `>=`, `<`, and `<=`. All six lexemes should be represented by a single token.
9. The adding operators are the binary `+` and `-`. Both lexemes should be represented by a single token.
10. The multiplying operators are `*` and `/`. Both lexemes should be represented by a single token.
11. The following punctuation symbols should be accepted: commas, colons, semicolons, and parentheses.
12. The following are reserved words:

    `begin, boolean, else, end, endif, function, if, is, integer, real, returns, then`

The lexical analyzer should be created using flex. The compiler should produce a listing of the program with lexical error messages included after the line in which they occur. Any character than cannot start any token should be considered a lexical error.

It should also generate a file containing the lexeme-token pairs as a means to verify that the lexical analyzer is working correctly. Only token numbers are required, not token names. The token numbers for the punctuation symbols should be the ASCII value of the character. The remaining tokens should be numbered sequentially beginning at 256.

The 70 points for functionality will be allocated as follows:

Compilation listing generated with line numbers	20 points
Detects and reports lexical errors correctly	25 points
Generates file with correct lexeme-token pairs	25 points

See the general project requirements for the details on submitting your project.