Ryan Long
CS 4100 – Compilers Pt 3B
11/23/2020

Description of Part 3B

CS410SyntaxAssign-B-FA20.pdf has instructions to turn in a word-processed, 1-paragraph overall description of the program:

This program is a syntax analyzer that extends the functionality of part 3A. If 3A found a syntax error, it was reported to the user and the program would stop parsing the file. If 3B finds a syntax error, it is reported to the user and a Resync() method is used to iterate through remaining tokens until a token is found that could be the start of a <statement>, and syntax parsing continues. If a file has multiple syntax errors, the user is now able to view all the errors without having to fix one error, check the syntax, fix another error, check the syntax, etc.

The <Program> method that is called in main begins the syntax checking that the tokens in the text file match $UNIT <identifier> $SEMICOLON <block> $PERIOD, where <block> leads to other non-terminal options including label and variable declaration, then a <block-body> non-terminal which is described as $BEGIN <statement> {$SEMICOLON <statement>} $END. This is interesting because <statement> can be called recursively, and <statement> can also call <block-body>. These recursive calls in 3B allow a file to have all sorts of variance and length while parsing for syntax errors.

Layout of this PDF:

- Description (this page)
- Source code (note comments at start of program that list line numbers of important methods)
- Verbose output from Part3BGOOD-1.txt
- Non-verbose output from Part 3B- Bad Testfile 1.txt