There are 49 nonterminals in my grammar. 99 production rules. the first plus sets of more than one production rules of any same non-terminal does not have any elements in common which makes the grammar LL(1).

Commands to run the code.(make sure each file is in the same folder)

For compilation:

javac Proscanner.java Scan.java TokenNames.java Token.java Pair.java RegularExpressions.java Parser.java

When the code compiles give the following command

java Parser (input file)

here input file is the c file that needs to be parsed.

Implementation:

I have used the scanner provided, which separates each token. I have stored each token in the array list named “tokens” . My parser.java file uses this arraylist of Proscanner.java file . For parsing procedure if I encounter a valid token , the token is deleted from the array list. So the first element of the arraylist will always be the next token to be parsed. The first nonterminal ‘program’ is called from the main method of parser.java file. The nonterminal further calls other nonterminals or terminals that appears in the production rule. Hence the RECURSIVE DESCENT PARSER approach is satisfied this way.

When a nonterminal rule gives empty the first+ set is checked it does not remove it from the array list instead the rule from which the nonterminal is called will consume and parse the token, so the empty conditions will be satisfied. Hence when every condition I satisfied the original call return to program prime, if it return successfully then pass method will be called.

The pass method in turn prints counts of variables, functions and statements.

Variable counts :

Whenever Identifier is parsed inside the datadecls rule it will increase the variable count.

Functioncount

When right brace is encountered inside func prime that is func definition is completed it will increase the function count.

Statement count

Whenever semicolon is parsed from the rule statements it will increase the statement count.

The code successfully parses trivial, recursion, sort, branch, loop, expression, square,fibonacci files and gives the counts of everything

It shows fail in other files which I believe in some is incorrect.