

Lab 3 Frequently Asked Questions

Question 1:

My grammar looks correct but the parser does not parse the sample programs. For example, I get the following error.

Starting Lexer

Starting Parser

[15,28] expecting: ','

Answer:

This message means that an error occurred in column 28 of line 15. If you go to column 28 of line 15 in the input file you will find a + there. However, your grammar looks OK. The first thing you should do, when you are unsure whether the problem is in the lexer or the parser, is to run the LexerDriver program from Lab2 (instead of ParserDriver) using the same input. Then look at the token produced at [15, 28]. This often saves time when the problem is in the lexer (token definitions) rather than in the parser specification (grammar rules). The problem in your lexer specification is that you have defined two tokens that match '+'. Remember that the token definition that occurs first in the file is the only one that is ever recognized, but your production rules reference the second token definition which is never recognized.

In summary, if the grammar looks correct, then the lexer from Lab 2 is usually producing the wrong token because the lexer is incorrect. Usually the token has been defined twice in the lexer or the token is incorrectly defined. When the tokens generated by the lexer are printed using LexerDriver from Lab2, then you can quickly determine whether the problem is in the lexer or in the production rules.

Question 2:

I am getting a shift-reduce error other than the one caused by the dangling else. What should I do?

Answer:

If you are getting such shift-reduce errors, then you have not defined the grammar correctly. Check each production rule carefully to make sure your grammar specification matches exactly the grammar given in the lab3 requirements specification. The problem is usually caused by a missing comma, semicolon, or parenthesis, etc. token in a production rule.

Question 3:

I am having a problem defining the grammar for lab3. Whenever I define a list in my grammar specification, the SableCC tool always gives me a SHIFT REDUCE ERROR.

Please explain why I'm getting this error.

Example:

```
args = expr comma [exp1]:expr*;
```

Answer:

The reason you are getting a shift-reduce error is because the above production rule specifies that only one comma should occur in the construct, that is, your example solution specifies that the args construct should allow the following:

x, y z 5 a

The comma between each expression is necessary to avoid the shift-reduce error. The comma indicates the end of an expression and that another expression will follow.

The shift reduce error will be eliminated by either one of the following because they both require a comma between each expression.

Solution1:

```
args = expr
      | args comma expr
      ;
```

Solution2:

```
args = expr comma_expr*;

comma_expr = comma expr;
```