

My CUP runneth over and spilleth all over the floor

A rough guide to using CUP
and debugging grammars

Example from manual

/* Precedences */

precedence left PLUS, MINUS;

precedence left TIMES;

precedence left UMINUS;

/* The grammar rules */

expr_list ::= expr_list expr:e SEMI
 | expr:e SEMI

;

expr ::= expr:e1 PLUS expr:e2
 | expr:e1 MINUS expr:e2
 | expr:e1 TIMES expr:e2
 | MINUS expr:e
 %prec UMINUS
 | LPAREN expr:e RPAREN
 | NUMBER:n
 ;

{: System.out.println(e);:}

{: System.out.println(e);:}

{: RESULT = e1+e2; :}

{: RESULT = e1-e2; :}

{: RESULT = e1*e2; :}

{: RESULT = -e; :}

{: RESULT = e; :}

{: RESULT = n; :}

Syntax error @ ...

Error: Syntax error @ Symbol: COLON_COLON_EQUALS (unknown:105/6 - unknown:105/9)
Error : Illegal use of reserved word

Actual error: Production before line 105 did not end in semicolon.

(Common missing punctuation:

commas between symbols is “terminal” and “nonterminal” declarations,
semicolon to end each production.

Line number is often off-by-one, sometimes off by a bunch.)

... has not been declared

Warning : LHS non terminal "program" has not been declared

Error : java_cup.runtime.Symbol "classes" has not been declared

Suggestion:

- First write (parts of) the grammar without declarations.
(CUP will complain)
- Then add declarations for TERMINALS and nonTerminals by visually scanning grammar (CUP will still complain)
- Then add the missing symbols that CUP found. (CUP will still complain)
- Repeat until CUP stops complaining

Reduce/reduce conflict ...

Warning : *** Reduce/Reduce conflict found in state #95
between thisDispatch ::= exp1 DOT ID LPAREN actuals RPAREN (*)
and objDispatch ::= exp1 DOT ID LPAREN actuals RPAREN (*)
under symbols: {RBRACE, SEMI, ASSIGN, COMMA, DOT, AT, RPAREN, TIMES, DIV,
PLUS, MINUS, LEQ, THEN, ELSE, FI, DO, OD, IN, OF}
Resolved in favor of the second production.

This is a serious grammar error --- an ambiguity. Inspect the grammar and understand why both productions can match the same text.

Shift/reduce conflict

Warning : *** Shift/Reduce conflict found in state #95
between objDispatch ::= exp1 DOT ID LPAREN actuals RPAREN (*)
under symbol DIV
Resolved in favor of shifting.

Priority problem:

On exp1.ID(actuals) /
do I reduce exp1.ID(actuals) to objDispatch
or shift the “/” to build a bigger expression?

... never reduced

Warning : *** Production "thisDispatch ::= exp1 DOT ID LPAREN actuals RPAREN "
never reduced

Probably a typo in the production that should refer to thisDispatch;
might be a mistake in thisDispatch:

```
exp1 ::= objDispatch | classDispatch | thisDispatch | exp0 ;  
objDispatch ::= exp1 DOT ID LPAREN actuals RPAREN ;  
classDispatch ::= exp1 AT TYPEID DOT ID LPAREN actuals RPAREN ;  
thisDispatch ::= exp1 DOT ID LPAREN actuals RPAREN ;
```

Errors cascade: Fixing this typo repaired my shift/reduce and reduce/reduce conflicts