

• [HOME](#) • [UP](#) •

[Top/Top-down parsing/Conflicts](#)

## 8.5. Parsing Conflicts

Let's build the top-down parsing table for a slightly different grammar.

1.  $L \rightarrow \epsilon$
2.  $L \rightarrow N$
3.  $N \rightarrow E$
4.  $N \rightarrow E, N$
5.  $E \rightarrow n$

The FIRST and FOLLOW sets for that grammar are:

$X$	$L$	$N$	$E$
<b>FIRST(<math>X</math>)</b>	{ $n, \epsilon$ }	{ $n$ }	{ $n$ }
<b>FOLLOW(<math>X</math>)</b>	{ $\$$ }	{ $\$$ }	{ $,, \$$ }

The parsing table is as follows.

Table D			
	$n$	$,$	$\$$
$L$	2		1
$N$	3,4		
$E$	6		

The algorithm adds two productions to one of the cells in the table. When that happens, it is called a *parsing conflict*.

The problem is that the parser cannot know whether to use production  $N \rightarrow E$  or  $N \rightarrow E, N$ , based on a single-token lookahead. That decision depends on whether the  $E$  is followed by a comma, and that would require a longer lookahead.

• [HOME](#) • [UP](#) •

[Top/Top-down parsing/Conflicts](#)

