



## 1. Grammar

- We will start with the grammar section, since this is the easiest to relate to what you are learning in Chapters 3 and 4.
- Productions

- Grammars are defined in near-BNF form. The differences can be summarized as follows:
- ◆ 1. Single characters used as terminals are put into single quote, but nonterminals are written out by name.

Appendix B.2 -- Yacc



- ◆ 3. Instead of —> in the production, Yacc uses a colon, but alternatives are separated by a | as usual.
- ◆ 4. Yacc uses a blank to represent an epsilon production.

Appendix B.2 -- Yacc

4













• Yacc assigns a numerical code to each token, and expects these codes to be returned to it by the lexical analyzer.

- This assignment is placed in yytab.h
- you can get Lex to use these by placing #include "yytab.h"
   inside the % { % } at the beginning of your Lex specification.
- Notice you do not have to declare nonterminals. Their appearances on the left-hand side of productions in the grammar section declares them automatically.
  Appendix B.2 - Yacc









