Using Yacc

File Format (1 of 4)

- Extension is .y
- <declarations>
  %%
  <translation rules>
  %%
  <supporting C functions>
- Anything in the <declarations> sections that is delimited by a line with "%{" to a line with "%}" is copied directly to the output C file
- All functions used in <translation rules> should be declared in the <declarations> section

File Format (2 of 4)

- Each line in the <declarations> section (other than those between "%{" and "%}") has the format:

  %start <nonTerminal>
  %token <listOfNames>
  %left <listOfTerminals>
  %right <listOfTerminals>
  %nonassoc <listOfTerminals>

- Precedence of tokens is in the order of declaration – lowest precedence first

File Format (3 of 4)

- Translation rules:
  <head> : <body$_1$> { <semantic action$_1$> }
  | <body$_2$> { <semantic action$_2$> }
  …
  | <body$_n$> { <semantic action$_n$> }
  ;
- A single quoted character is the terminal symbol
- $$ is the attribute associated with the head
- $*i* is the attribute associated with the *i*th grammar symbol of the body (either terminal or non-terminal)

File Format (4 of 3)

- Unquoted strings of letters and digits not declared to be tokens are taken to be non-terminals
- Copying the value is the default action for productions with a single grammar symbol in the body ($$ = $1;)

Dealing with Ambiguity in Yacc

- A reduce/reduce conflict is resolved by choosing the conflicting production listed first in the Yacc specification
- A shift/reduce conflict is resolved in favor or shift
- Precedence and associativity can be assigned to terminals by using %left, %right, and %nonassoc

Dealing with Ambiguity in Yacc

- Normally the precedence of a production is the same as that of its rightmost terminal
- This can be changed by appending

  %prec <terminal>

  to a production body

Including the Lexer

- Specify

  ```
  #include "lex.yy.c"
  ```

  in the third part of the Yacc input file to include the lexer built by Lex

Errors Detected by Yacc

- The function yyerror is called by Yacc whenever an error is detected
- A single parameter is passed to yyerror of the type:

  char *
- That string will contain a description of the error detected by Yacc

Compiling a Yacc file

- lex lexer.lex
- yacc parser.y
- gcc y.tab.c -ly -lfl -o parser
  - y.tab.c is the output of Yacc
  - -ly means to link with the Yacc libraries
  - -lfl means to link with the flex libraries (on some systems, -ll may be needed to link with lex libraries)
  - -o is used to specify the name of the executable file