yacc — Command

Parser generator
yacc [*option ...***]** *file*
cc y.tab.c [-ly]

Many programs process highly structured input according to given rules. Compilers are a familiar example. Two of the most complicated parts of such programs are *lexical analysis* and *parsing* (sometimes called *syntax analysis*). The COHERENT system includes two powerful tools called **lex** and **yacc** to assist you in performing these tasks. **lex** takes a set of lexical rules and writes a lexical analyzer, whereas **yacc** takes a set of parsing rules and writes a parser; both output C source code that can be compiled into a full program.

The term *yacc* is an acronym for "yet another compiler-compiler". In brief, the **yacc** input *file* describes a context free grammar using a BNF-like syntax. The output is a file **y.tab.c**; it contains the definition of a C function **yyparse**(), which parses the language described in *file*. The output is ready for processing by the C compiler **cc**. Ambiguities in the grammar are reported to the user, but resolved automatically by precedence rules. The user must provide a lexical scanner **yylex**(), which you may generate with **lex**. The **yacc** library includes default definitions of **main**, **yylex**, and **yyerror**, and may be included with the option **-ly** on the **cc** command line.

yacc recognizes the following options:

-d Enable debugging output; implies **-v.**

-hdr *headerfile*
Put the header output in *headerfile* instead of **y.tab.h**.

-items *N*
Allow *N* items per state. This option is designed to help **yacc** users deal with the ANSI C grammar.

-l *listfile*
Place a description of the state machine, tokens, parsing actions, and statistics in file *listfile*.

-sprod *N*
Allow *N* symbols per production; default, 20. This option is designed to help **yacc** users deal with the ANSI C grammar.

-st Print statistics on the standard output.

-v Verbose option. Like **-l**, but places the listing in file **y.output** by default.

The following options are useful if table overflow messages appear:

-nterms *N*
Allow for *N* nonterminals; default, 100.

-prods *N*
Allow for *N* productions (rules); default, 350.

-states *N*
Allow for *N* states; default, 300.

-terms *N*
Allow for *N* terminal symbols; default 100.

-types *N*
Allow for *N* types; default, ten.

LEXICON

Files

y.tab.c — C source output
y.tab.h — Default C header output
y.output — Default listing output
/lib/yyparse.c — Protoparser
/tmp/y[ao]* — Temporaries
/usr/lib/liby.a — Library

See Also

cc, commands, lex, Programming COHERENT
Introduction to yacc, Yet Another Compiler-Compiler

Diagnostics

yacc writes onto the standard error the number of R/R (reduce/reduce) and S/R (shift/reduce) conflicts (ambiguities).

Notes

The version of **yacc** shipped prior to release 4.2 of COHERENT included the header file **<action.h>** in its output. This file's data are now built into parser skeleton in **/lib/yyparse**, thus obviating **<action.h>**. This header has been dropped from COHERENT. You should re-run **yacc** to update the source files generated by previous versions of **yacc**.

yes — Command

Print infinitely many responses
yes [*string* **]**

With no argument, **yes** prints the string **y\n** forever. If a *string* is named on the command line, then **yes** prints it forever.

Example

The following example scribbles the string **foo\n** over a high-density, 5.25-inch floppy disk in drive 0 (drive A):

```
yes foo >/dev/fha0
```

See Also

commands