

NAME

Yacc – an LALR(1) parser generator

SYNOPSIS

yacc [**-dgilrtv**] [**-b** *file_prefix*] [**-p** *symbol_prefix*] *filename*

DESCRIPTION

Yacc reads the grammar specification in the file *filename* and generates an LALR(1) parser for it. The parsers consist of a set of LALR(1) parsing tables and a driver routine written in the C programming language. **Yacc** normally writes the parse tables and the driver routine to the file *y.tab.c*.

The following options are available:

-b *file_prefix*

The **-b** option changes the prefix prepended to the output file names to the string denoted by *file_prefix*. The default prefix is the character *y*.

-d The **-d** option causes the header file **y.tab.h** to be written. It contains `#define`'s for the token identifiers.

-g The **-g** option causes a graphical description of the generated LALR(1) parser to be written to the file **y.dot** in graphviz format, ready to be processed by `dot(1)`.

-i The **-i** option causes a supplementary header file **y.tab.i** to be written. It contains extern declarations and supplementary `#define`'s as needed to map the conventional *yacc* **yy**-prefixed names to whatever the **-p** option may specify. The code file, e.g., **y.tab.c** is modified to `#include` this file as well as the **y.tab.h** file, enforcing consistent usage of the symbols defined in those files.

The supplementary header file makes it simpler to separate compilation of *lex*- and *yacc*-files.

-l If the **-l** option is not specified, **yacc** will insert `#line` directives in the generated code. The `#line` directives let the C compiler relate errors in the generated code to the user's original code. If the **-l** option is specified, **yacc** will not insert the `#line` directives. `#line` directives specified by the user will be retained.

-o *output_file*

specify the filename for the parser file. If this option is not given, the output filename is the file prefix concatenated with the file suffix, e.g., **y.tab.c**. This overrides the **-p** option.

-p *symbol_prefix*

The **-p** option changes the prefix prepended to yacc-generated symbols to the string denoted by *symbol_prefix*. The default prefix is the string **yy**.

-P create a reentrant parser, e.g., "%pure-parser".

-r The **-r** option causes **yacc** to produce separate files for code and tables. The code file is named *y.code.c*, and the tables file is named *y.tab.c*. The prefix "y." can be overridden using the **-b** option.

-s suppress `"#define"` statements generated for string literals in a `"%token"` statement, to more closely match original **yacc** behavior.

Normally when **yacc** sees a line such as

```
%token OP_ADD "ADD"
```

it notices that the quoted "ADD" is a valid C identifier, and generates a `#define` not only for

OP_ADD, but for ADD as well, e.g.,

```
#define OP_ADD 257
#define ADD 258
```

The original **yacc** does not generate the second **#define**. The **-s** option suppresses this **#define**.

POSIX (IEEE 1003.1 2004) documents only names and numbers for **%token**, though original **yacc** and bison also accept string literals.

- t** The **-t** option changes the preprocessor directives generated by **yacc** so that debugging statements will be incorporated in the compiled code.
- v** The **-v** option causes a human-readable description of the generated parser to be written to the file *y.output*.
- V** print the version number to the standard output.
- y** **yacc** ignores this option, which bison supports for ostensible POSIX compatibility.

EXTENSIONS

yacc provides some extensions for compatibility with bison and other implementations of yacc:

%expect *number*

tell **yacc** the expected number of shift/reduce conflicts. That makes it only report the number if it differs.

%expect-rr *number*

tell **yacc** the expected number of reduce/reduce conflicts. That makes it only report the number if it differs. This is (unlike bison) allowable in LALR parsers.

%lex-param { *argument-declaration* }

By default, the lexer accepts no parameters, e.g., **yylex()**. Use this directive to add parameter declarations for your customized lexer.

%parse-param { *argument-declaration* }

By default, the parser accepts no parameters, e.g., **yyparse()**. Use this directive to add parameter declarations for your customized parser.

%pure-parser

Most variables (other than **yydebug** and **yynerrs**) are allocated on the stack within **yyparse**, making the parser reasonably reentrant.

PORTABILITY

According to Robert Corbett,

Berkeley Yacc is an LALR(1) parser generator. Berkeley Yacc has been made as compatible as possible with AT&T Yacc. Berkeley Yacc can accept any input specification that conforms to the AT&T Yacc documentation. Specifications that take advantage of undocumented features of AT&T Yacc will probably be rejected.

The rationale in

<http://pubs.opengroup.org/onlinepubs/9699919799/utilities/yacc.html>

documents some features of AT&T yacc which are no longer required for POSIX compliance.

That said, you may be interested in reusing grammar files with some other implementation which is not strictly compatible with AT&T yacc. For instance, there is bison. Here are a few differences:

- **Yacc** accepts an equals mark preceding the left curly brace of an action (as in the original grammar file

ftp.y:

```
|          STAT CRLF
= {
          statcmd();
}
```

- **Yacc** and bison emit code in different order, and in particular bison makes forward reference to common functions such as `yylex`, `yyparse` and `yyerror` without providing prototypes.
- Bison's support for "%expect" is broken in more than one release. For best results using bison, delete that directive.
- Bison has no equivalent for some of **yacc**'s command-line options, relying on directives embedded in the grammar file.
- Bison's "-y" option does not affect bison's lack of support for features of AT&T yacc which were deemed obsolescent.

DIAGNOSTICS

If there are rules that are never reduced, the number of such rules is reported on standard error. If there are any LALR(1) conflicts, the number of conflicts is reported on standard error.