NAME

jay – an LALR(1) parser generator for Java and C#

SYNOPSIS

jay [**-ctv**] [**-b** *file_prefix*] *filename* **<** *skeleton*

DESCRIPTION

Jay reads the grammar specification in the file *filename* and generates an LALR(1) parser for it. A parser consists of a set of parsing tables and a driver routine from the file *skeleton*. Suitable *skeleton* files exist for Java and C#. Tables and driver are written to standard output.

The following options are available:

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

-c arranges for C preprocessor **#line** directives to be incorporated in the compiled code.

-t arranges for debugging information to be incorporated in the compiled code. This option depends on additional runtime packages.

-v causes a human-readable description of the generated parser to be written to the file *y.output*.

If the environment variable TMPDIR is set, the string denoted by TMPDIR will be used as the name of the directory where the temporary files are created.

FILES

y.output
/tmp/yacc.aXXXXXX
/tmp/yacc.tXXXXXX
/tmp/yacc.uXXXXXX

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.

HISTORY

Jay is derived from Berkeley *yacc*. Input conventions closely follow those of *yacc*; for details, consult the desk calculator and error recovery examples and the available *skeleton* files. A tutorial example for Java can be found at

> http://www.cs.rit.edu/˜ats/lp-2002-2/