NAME

compiler488 − a compiler for the CSC488S Project Language

SYNOPSIS

compiler488 [−X] [**−D**[*abxy*]] [**−T**[*acilpsxy*]][**−O** *outputFilename*]
[**−E** *errorFilename*] [**−R** *traceFilename*] [**−U** *dumpFilename*]
[**−I** *runInputFilename*] [*sourceFile*]

DESCRIPTION

compiler488 is an incredibly well designed and implemented compiler for the CSC488S Course Project Language.

The compiler compiles and then attempts to execute one program written in the CSC488S Project Language. The compiler reads the source program from *sourceFile* if it was specified in the command that invoked the compiler. Otherwise it exepcts the source program on standard input.

OPTIONS

The options currently implemented by the compiler488 are:

−X Suppress execution of the compiled program. Saves time when testing an incomplete code generator.

−D Specify dump options. The letters *abxy* indicate which information should be dumped to the compilers *dumpFile*.
a − dump Abstract Syntax Tree after Pass 1
b − dump Abstract Syntax Tree after Pass 2
x − dump the compiled code just before execution
y − dump symbol table information

−T Specify trace options. The letters *acilpsxy* indicate which trace information should be written to the compilers *traceFile*.
a − trace Abstract Syntax Tree operations
c − trace code generation
i − trace source program input in the scanner
l − trace tokens in lexical analysis
p − trace parsing
s − trace semantic analysis
x − trace program execution
y − trace symbol table operations

−E *errorFile*
Specify an alternative file to receive error messages generated by the compiler. Default for error messages is stderr.

−O *outputFileName*
Specify an alternative file to receive compilation output Default is stdout

−R *traceFileName*
Specify an alternative file to receive compilation trace informantion. Default for trace information is stdout.

−S *traceFileName*
Specify an alternative file to receive execution trace informantion. Default for trace information is stdout.

−U *dumpFileName*
Specify an alternative file to receive compiler dump information. Default for dump information is stdout.

−V Print information about the compiler488 version on standard output and continue processing. This option uses the embedded string constant *version* in the top level classes. These constants will need to be manually maintained for this option to provide useful information.

−I *runInputFileName*

Specify an alternative file to serve as a souce of input during execution of the compiled program. Default for execution time input is stdin.

ENVIRONMENT

The compiler does not use any environment variables.

SEE ALSO

The CSC488S course bulletin board.

AUTHORS

The **compiler488** skeleton was conceived and implemented by Prof. Dave Wortman. Conversion from C to Java was partially done by Danny House. The present version of the skeleton was designed by Profs. Marsha Chechik and Dave Wortman.

Abstract Symbol Tree creation, semantic analysis, code generation and the symbol table mechanisms were skillfully designed and implemented by **PUT YOUR NAMES HERE**

BUGS

There are none; only a few unexpected features.

DUMPING and TRACING

The compiler driver main program implements the `-D` and `-T` to provide an easy interface to implementation defined dumping and tracing of the compiler phases. These flags are provided as a convenience for teams that choose to implement this functionality.

In the current release of the compiler
- The scanner and parser trace options (`i, l`) are unimplemented.
- AST dumping (`a, b`) is implemented via the AST `printOn` interface.
- Compiled program dumping (`x`) and program execution tracing (`x`) are implemented.

All of the other dump and trace operations implied by this man page are **optional**. The implementation of these operations to assist in the debugging of the compiler is a strategic decision for each team to make. None of these dumps or traces are required.