C-Fast

version 0,2 -- preAlpha release

the Programmers tool
by Jeff Watkins

What C-Fast Does:

C-Fast is a C syntax checker for Microsoft Windows. C-Fast is one small step away from being a real compiler: it builds a symbol table, but does not output any object files. It does output a browser file that contains all the information necessary to browse on your project. But that is a separate package (I will be starting it soon).

How To Use C-Fast:

Well, excepting the fact that it is a pre-Alpha release, run C-Fast. It will come up (warning, this is for 800x600 vga, that is only because that is the way the code for setting up the windows is written. As C-Fast will not ultimately look like this at all, this shouldn't be a real problem.

There are two menus in C-Fast: *File* and *Syntax Check*. From the file menu you can select a c file to check or you can exit. (lot's of choices eh?) From the syntax check menu there is only one option: *go*. This starts the syntax check.

What are these windows?

The top left window is the source listing window. The bottom left window is the error message listing window. The three windows on the right are in order from top to bottom: global type names, global variables, and function names.

The error message listing is currently turned off. That is because as a standard list box, it often overflows and uae's. Remember, this is not what it will eventually look like.

There is also a abort button in the bottom right corner (only visible in 800x600 mode, sorry). This sets a flag that tells the parser to abort. The Parser will eventually test this flag and abort.

The future of C-Fast:

Well, eventually C-Fast will be merely an import tool for a c code browser. This browser outputs standard c code. It launches your compiler, and finds your errors. There should be very few errors as the code is interactively parsed and syntax corrected on the fly.

Also, there is no reason to limit this to C. The parser generator has a standard parser skeleton which I hacked to make it run politely under windows. It also comes with grammars for C, Pascal, ADA, and (ick) FORTRAN. Now, these

grammars are probably as sloppily written as the C grammar was (boy patching that was a lot of fun); but, this shouldn't be too bad.

Warning!

The only reason I released this version was to get ideas and suggestions from the programming community at large. I would appreciate all comments (so long as they are constructive :-)

You may reach the *author* via Internet at jwatkins@cadsun6.gatech.edu