

Debugging in MATLAB

Toby Driscoll, January 7, 1998

Debugging is available for any M-file function (not scripts). A complete list of commands can be found from `help debug`.

- To set a breakpoint when the function `foo` is called, type `dbstop in foo`. To stop at line 17, enter `dbstop at 17 in foo`.
- Use `dbtype foo` to get a listing of the function with line numbers.
- When MATLAB reaches a function breakpoint, it stops execution and gives you the prompt `K>>`. At this prompt, you can enter any commands, inspect or change variables, etc.
- To execute the next line only, enter `dbstep`. To continue execution to the next breakpoint or termination, use `dbcont`. To quit all function executions and return to the normal prompt, type `dbquit`.
- You can check the current function-call stack by issuing `dbstack`. You can move up and down the stack of function calls with `dbup` and `dbdown`. For instance, if `Euler` calls `foo` and you stop in `foo`, then `dbup` takes you into `Euler` at the point `foo` was called. Another `dbup` takes you into the base workspace.
- To see all breakpoints, type `dbstatus`. To clear all breakpoints in `foo`, use `dbclean all in foo`. Breakpoints automatically disappear whenever the function is modified and saved.
- Another useful tool is the `keyboard` command. This is a command you put in the function itself. When MATLAB reaches it, it behaves just as though a breakpoint were set there. The main advantage of `keyboard` is that it can be put inside an `if` statement for conditional stopping.