CSE306 Software Quality in Practice

Dr. Carl Alphonce
alphonce@buffalo.edu
343 Davis Hall

Debugging with gdb

Common compiler options

-std set language standard

-o set output file name

-g debugging

-c compile/assemble do not link

-Wall report "all" warnings

-L library path <- search for library files here

-I include path <- where to find .h files

-l library <- search this library during linking

compiling and running

without debugger

- compile using gcc, with '-o' flag if you want to specify a name for the resulting executable (other than "a.out")
  - gcc -o factorial factorial.c main.c
- launch program using by running executable:
  - ./factorial 5

with debugger

- compile using gcc, with '-g' flag to include debugging information in executable (name of executable is up to you, but adding .debug is a reminder that debugging information is included).
  - gcc -g -o factorial.debug factorial.c main.c
- launch program using gdb
  - gdb ./factorial.debug

NB: no program argument supplied in gdb invocation

basic commands

- quit - get out of gdb
- help - on-line help system
- run (with program arguments)

Run in Emacs

https://emacsdocs.org/docs/emacs/GDB-Graphical-Interface

- M-x gdb
- (M-x gdb-display-disassembly-buffer)

Toggle between UI modes

- C-x C-a for TUI/standard mode toggle
- C-x 1 for code only
- C-x 2 for code and assembly

TUI mode commands:

https://sourceware.org/gdb/current/onlinedocs/gdb/TUI-Commands.html#TUI-Commands

TUI mode not always available

- Not all environments support the TUI mode
- All environments support the standard command-line mode: learn these commands

short demo

- bt (backtrace)
- up / down / frame N
- info frame / info args / info locals
- break <function> / break <line> / break <bp> if <expr>
- enable / disable
- ignore <bp> N
- tbreak (a once-only breakpoint)
- run / step / continue / next

Inspecting/changing variables

- print <var> (= <expr>)
- set var <var> = <expr>
- print <expr> -> evaluate and print, carrying out function calls
- call <expr> -> evaluate, do not print

returning from a function call

- return -> discard frame (and subframes)
- return <expr> -> as above, <expr> is returned
- finish -> complete execution of this function normally
- kill -> terminate execution of the program being debugged

factorial code from lecture

GitHub classroom link on course website:
Day06 exercise - gdb intro

helpful resources

https://www.recurse.com/blog/7-understanding-c-by-learning-assembly

https://sourceware.org/gdb/current/onlinedocs/gdb/