Development: Revision Control

March 28th, 2004
Class Meeting 10

Software Development Process

- Creation of source files
  - RCS
- Compilation and linking
  - g++ and make
- Running and testing programs
  - gdb

Why use a debugger?

- No one writes perfect code the first time, every time
- Desk checking code can be tedious and error-prone
- Putting print statements in the code requires re-compilation and a guess as to the source of the problem
- Debuggers are powerful and flexible

Common Debugger Functions

- Run program
- Stop program at breakpoints
- Execute program one line of code at a time
- Display values of variables
- Show sequence of function calls
- Catch signals

The GNU Debugger (gdb)

- Free command-line debugger
- Common utilization:
  - gdb executable
  - gdb executable core
  - gdb executable process_id

Execution Commands

- list of l
  - lists source code
  - list
  - list function_name
  - list line_number
- run of r
  - run program from beginning
  - run
  - run argument1 argument2
- next of n
  - execute next line, stepping over function calls
- step of s
  - execute next line, stepping into function calls
Breakpoint Commands

- **break or b**
  - set a breakpoint
  - break function_name
  - break line_number
- **delete or d**
  - delete a breakpoint
  - delete breakpoint_number
- **continue or c**
  - continue execution when stopped

Program Information Commands

- **print or p**
  - prints value of a variable or expression
  - print x
  - print x*y
  - print function(x)
- **display**
  - continuously display value of a variable
- **undisplay**
  - Stop displaying value of a variable

Miscellaneous Commands

- **set**
  - change a variable’s value
  - set n=3
- **help or h**
  - display help text
  - help
  - help command_name
  - help keyword
- **quit or q**
  - quit gdb