Debugging

CMSC 202

Warmup

What is the bug in the following code?

```
int* foo(int a)
{
  return &a;
}
```

What are Errors?

- Syntax Errors
  - Compiler/Linker catch these
  - Mistakes in your formatting of C++
- Semantic/Logic Errors
  - Nothing catches these
  - Misunderstanding of programmer about what system is supposed to do
    - These are BUGS
    - Mismatch between what system is supposed to do and what it actually does

Finding Bugs

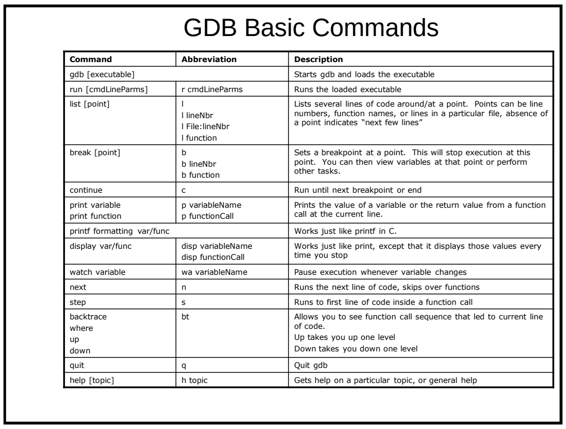
- Categories of Bugs
  - Seg-fault or core-dump (fatal!)
  - Program infinitely loops
  - Runs but output is incorrect
- Strategies
  - Look through code line by line
  - Print values every once in a while
  - Use a debugger (best choice!)
- Professional Programmers?
  - Use a mix of these strategies!

Debuggers on GL

- GDB
  - "GNU DeBugger"
  - Text-based
  - Fast to load
- DDD
  - "GNU Data Display Debugger"
  - Graphically based
  - Easier to use
  - Slower to load/interact with (remotely)
  - Must install an *NIX emulator
    - Check Resources page
      - "Remotely Accessing the GL Servers"

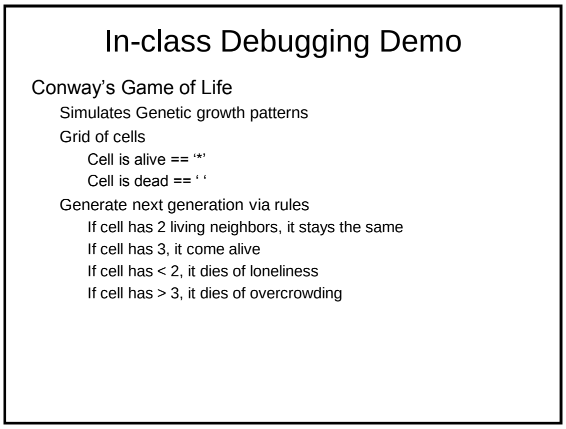
GDB/DDD – Linux/UNIX debugger

- Allows you to:
  - Run program from start
  - See which line seg-faulted
  - Run program line by line
  - Stop at any point
  - Print variables at any point
  - View parameters
  - Trace through function calls
  - Exit
  - Get Help on any feature



GDB Basic Commands

Command	Abbreviation	Description
gdb [executable]		Starts gdb and loads the executable
run [cmdLineParms]	r cmdLineParms	Runs the loaded executable
list [point]	l l lineNbr l File:lineNbr l function	Lists several lines of code around/at a point. Points can be line numbers, function names, or lines in a particular file, absence of a point indicates "next few lines"
break [point]	b b lineNbr b function	Sets a breakpoint at a point. This will stop execution at this point. You can then view variables at that point or perform other tasks.
continue	c	Run until next breakpoint or end
print variable print function	p variableName p functionCall	Prints the value of a variable or the return value from a function call at the current line.
printf formatting var/func		Works just like printf in C.
display var/func	disp variableName disp functionCall	Works just like print, except that it displays those values every time you stop
watch variable	wa variableName	Pause execution whenever variable changes
next	n	Runs the next line of code, skips over functions
step	s	Runs to first line of code inside a function call
backtrace where up down	bt	Allows you to see function call sequence that led to current line of code. Up takes you up one level Down takes you down one level
quit	q	Quit gdb
help [topic]	h topic	Gets help on a particular topic, or general help



In-class Debugging Demo

- Conway's Game of Life
  - Simulates Genetic growth patterns
  - Grid of cells
    - Cell is alive == '*'
    - Cell is dead == ' '
  - Generate next generation via rules
    - If cell has 2 living neighbors, it stays the same
    - If cell has 3, it come alive
    - If cell has < 2, it dies of loneliness
    - If cell has > 3, it dies of overcrowding