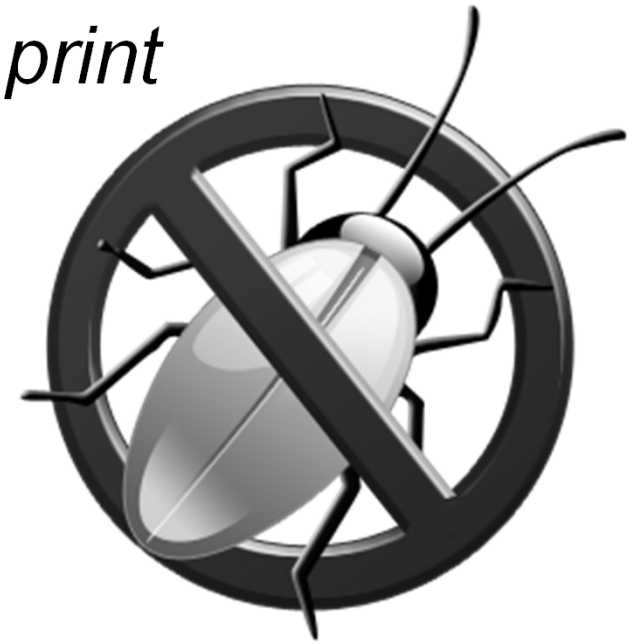
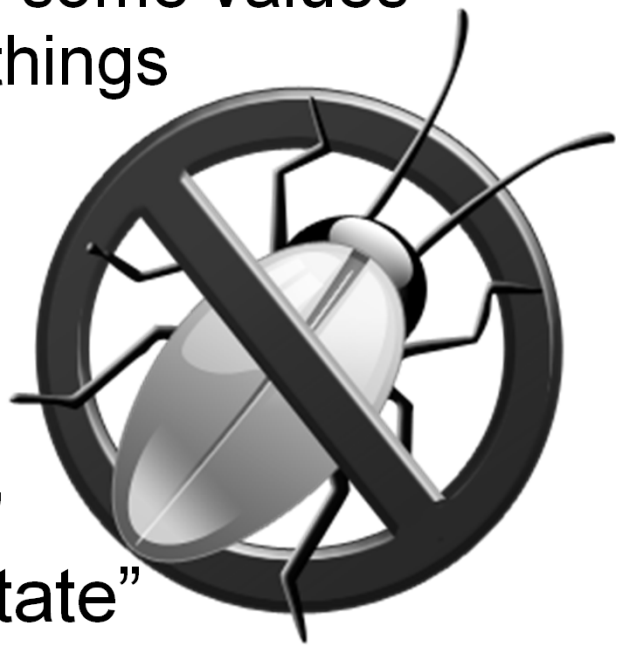
Popular Ways to Debug

- Using “printf”
  - Often mocked, viewed as unscientific
  - In fact, just an easy way to apply dynamic/log analysis and perform experiments
  - If you ask the right questions, *print* can be a great debugging tool
    - Supports scientific debugging
    - I have no lessons here, other than to print intelligently, not blindly grope around



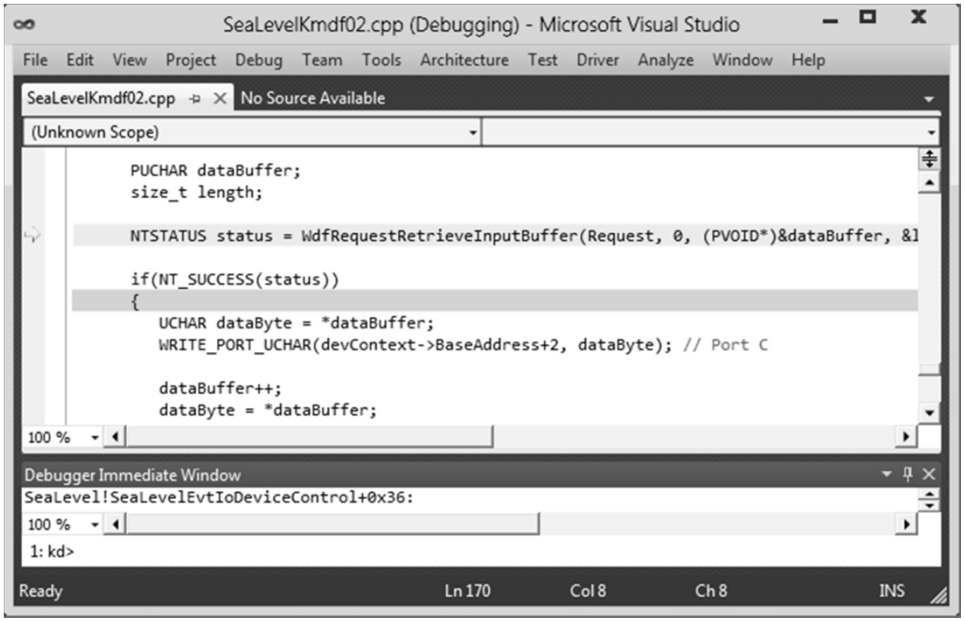
Popular Ways to Debug

- Using a debugger
  - Usually thought of as “more scientific”
  - It can be!
  - A debugger is good when you want to:
    - Inspect closely what happens to some values
    - Slow down and carefully watch things during one part of a run
    - Get information across a lot of state at once
  - Hard to make guidelines, but in general printf is for “across time” and debuggers are for “across state”



Debuggers

- Let you “take over” a program and run it under more control than usual
- Command line (gdb) and visual debuggers are both popular and widely used



GDB

- There are many other debuggers out there
- Won't spend a lot of time on GDB specifics
- Major features (common to many debuggers or becoming more common)
  - Single step through a program line at a time
    - GDB: **step** and **next**
      - **next** skips over functions called in a line
  - Inspect memory locations/values
  - Set breakpoints – places to stop execution
    - Can be conditional (break at line XX if y > z)
  - Set watchpoints – events to watch for in execution
  - Change values in memory
  - GDB can also "run a program backwards" a little bit now!

GDB

- Si
  de
  - ault
  - ther

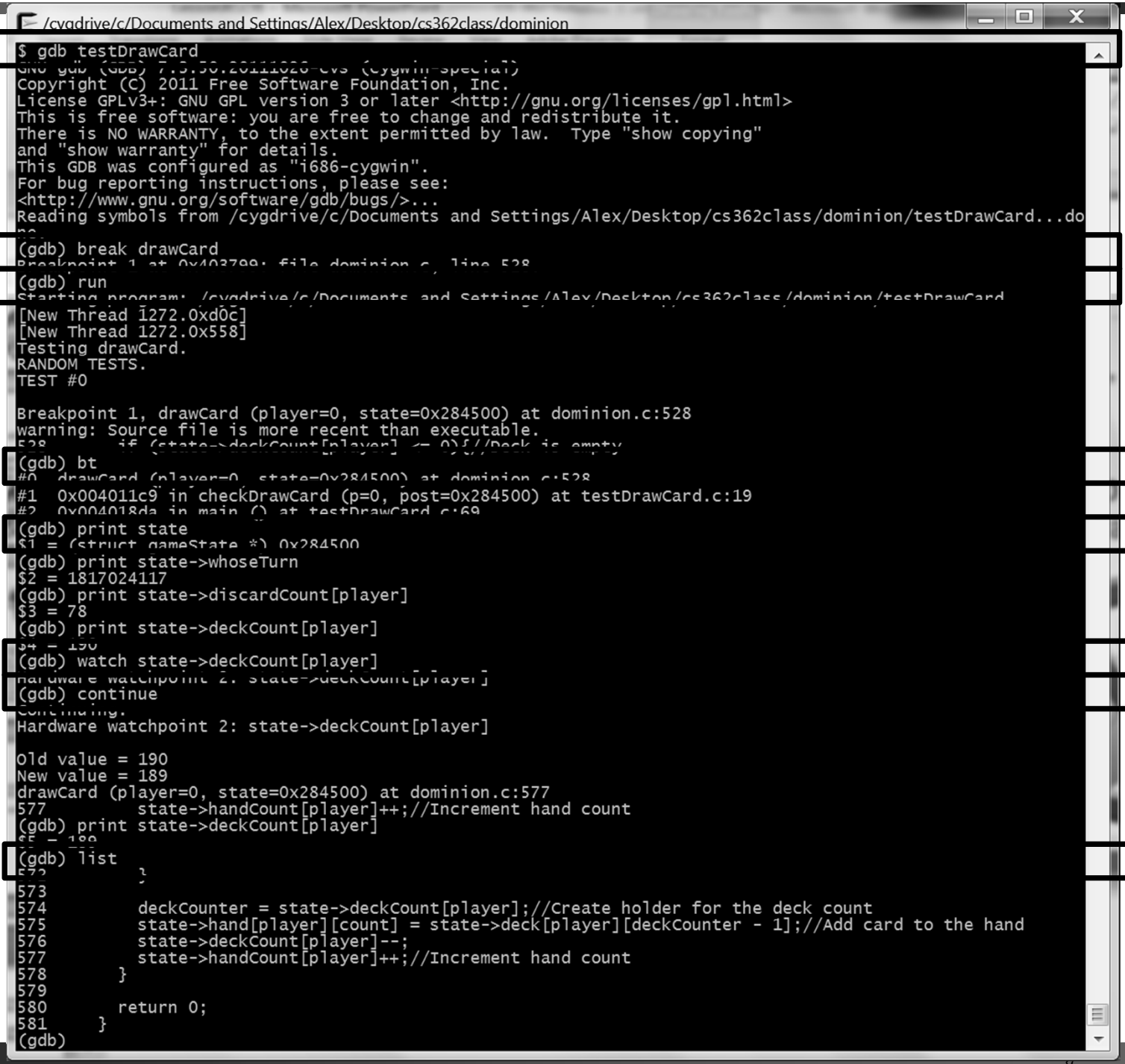
```
/cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/dominion

Alex@groce /cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/dominion
$ ./badTestDrawCard.exe
Testing drawCard.
RANDOM TESTS.
Segmentation fault (core dumped)

Alex@groce /cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/dominion
$ gdb ./badTestDrawCard.exe
GNU gdb (GDB) 7.3.50.20111026-cvs (cygwin-special)
Copyright (C) 2011 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-cygwin".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from /cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/
dominion/badTestDrawCard.exe...done.
(gdb) run
Starting program: /cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/dom
inion/badTestDrawCard.exe
[New Thread 1980.0x46c]
[New Thread 1980.0xf2c]
Testing drawCard.
RANDOM TESTS.

Program received signal SIGSEGV, Segmentation fault.
0x00403348 in drawCard (player=145, state=0x2844f0) at dominion.c:534
534             state->deck[player][i] = state->discard[player][i];
(gdb) bt
#0  0x00403348 in drawCard (player=145, state=0x2844f0) at dominion.c:534
#1  0x004011a4 in checkDrawCard (p=145, post=0x2844f0) at badTestDrawCard.c:14
#2  0x0040141d in main () at badTestDrawCard.c:38
(gdb) print i
$1 = 0
(gdb) print player
$2 = 145
(gdb)
```

GDB


/cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/dominion
$ gdb testDrawCard
Copyright (C) 2011 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-cygwin".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from /cygdrive/c/Documents and Settings/Alex/Desktop/cs362class/dominion/testDrawCard...do
(gdb) break drawCard
(gdb) run
[New Thread 1272.0xd0c]
[New Thread 1272.0x558]
Testing drawCard.
RANDOM TESTS.
TEST #0
Breakpoint 1, drawCard (player=0, state=0x284500) at dominion.c:528
warning: Source file is more recent than executable.
(gdb) bt
#1 0x004011c9 in checkDrawCard (p=0, post=0x284500) at testDrawCard.c:19
(gdb) print state
(gdb) print state->whoseTurn
$2 = 1817024117
(gdb) print state->discardCount[player]
$3 = 78
(gdb) print state->deckCount[player]
(gdb) watch state->deckCount[player]
(gdb) continue
Hardware watchpoint 2: state->deckCount[player]
Old value = 190
New value = 189
drawCard (player=0, state=0x284500) at dominion.c:577
577 state->handCount[player]++;//Increment hand count
(gdb) print state->deckCount[player]
(gdb) list
573
574 deckCounter = state->deckCount[player];//Create holder for the deck count
575 state->hand[player][count] = state->deck[player][deckCounter - 1];//Add card to the hand
576 state->deckCount[player]--;
577 state->handCount[player]++;//Increment hand count
578 }
579
580 return 0;
581 }
(gdb)