

LAB 5: Debugging gdb

John Dempsey

COMP-232 Programming Languages
California State University, Channel Islands
September 20, 2023

Debugging gdb shows some of the basic gdb command which you can use to step through your code line-by-line, print out variables, and locate where segmentation violation errors occur.

```
john@oho:~$ mkdir LAB5
```

```
john@oho:~$ cd LAB5
```

```
john@oho:~/LAB5$ sftp john@comp232.com
```

```
john@comp232.com's password:
```

```
Connected to comp232.com.
```

```
sftp> cd /home/LAB5
```

```
sftp> ls -l
```

```
-rw-r--r-- 1 john john 334 Feb 21 05:20 Makefile
-rw-r--r-- 1 john john 211 Feb 21 05:20 getComputerMove.c
-rw-r--r-- 1 john john 308 Feb 21 05:20 getHumanMove.c
-rw-r--r-- 1 john john 101 Feb 21 05:20 initBoard.c
-rw-r--r-- 1 john john 843 Feb 21 05:20 is3inARow.c
-rw-r--r-- 1 john john 291 Feb 21 05:20 isAWin.c
-rw-r--r-- 1 john john 248 Feb 21 05:20 isBoardFull.c
-rw-r--r-- 1 john john 86 Feb 21 05:20 main.c
-rw-r--r-- 1 john john 156 Feb 21 05:20 makeMove.c
-rw-r--r-- 1 john john 374 Feb 21 05:20 max.c
-rw-r--r-- 1 john john 374 Feb 21 05:20 min.c
-rw-r--r-- 1 john john 919 Feb 21 05:20 minMax.c
-rw-r--r-- 1 john john 280 Feb 21 05:20 printBoard.c
-rw-r--r-- 1 john john 1327 Feb 21 05:20 runGame.c
-rw-r--r-- 1 john john 607 Feb 21 05:20 tictactoe.h
```

```
sftp> mget *
```

```
Fetching /home/LAB5/Makefile to Makefile
/home/LAB5/Makefile 100% 334
5.0KB/s 00:00
Fetching /home/LAB5/getComputerMove.c to getComputerMove.c
/home/LAB5/getComputerMove.c 100% 211
3.6KB/s 00:00
Fetching /home/LAB5/getHumanMove.c to getHumanMove.c
```

```

/home/LAB5/getHumanMove.c                               100%  308
5.9KB/s   00:00
Fetching /home/LAB5/initBoard.c to initBoard.c
/home/LAB5/initBoard.c                                 100%  101
1.8KB/s   00:00
Fetching /home/LAB5/is3inARow.c to is3inARow.c
/home/LAB5/is3inARow.c                               100%  843
12.5KB/s  00:00
Fetching /home/LAB5/isAWin.c to isAWin.c
/home/LAB5/isAWin.c                                  100%  291
3.8KB/s   00:00
Fetching /home/LAB5/isBoardFull.c to isBoardFull.c
/home/LAB5/isBoardFull.c                             100%  248
4.2KB/s   00:00
Fetching /home/LAB5/main.c to main.c
/home/LAB5/main.c                                    100%   86
1.5KB/s   00:00
Fetching /home/LAB5/makeMove.c to makeMove.c
/home/LAB5/makeMove.c                                100%  156
2.4KB/s   00:00
Fetching /home/LAB5/max.c to max.c
/home/LAB5/max.c                                      100%  374
6.7KB/s   00:00
Fetching /home/LAB5/min.c to min.c
/home/LAB5/min.c                                      100%  374
5.8KB/s   00:00
Fetching /home/LAB5/minMax.c to minMax.c
/home/LAB5/minMax.c                                  100%  919
13.8KB/s  00:00
Fetching /home/LAB5/printBoard.c to printBoard.c
/home/LAB5/printBoard.c                             100%  280
4.7KB/s   00:00
Fetching /home/LAB5/runGame.c to runGame.c
/home/LAB5/runGame.c                                 100% 1327
23.3KB/s  00:00
Fetching /home/LAB5/tictactoe.h to tictactoe.h
/home/LAB5/tictactoe.h                               100%  607
10.6KB/s  00:00

```

sftp> quit

john@oho:~/LAB5\$ ls -l

total 16

```

-rw-r--r-- 1 john john 334 Feb 20 21:27 Makefile
-rw-r--r-- 1 john john 211 Feb 20 21:27 getComputerMove.c
-rw-r--r-- 1 john john 308 Feb 20 21:27 getHumanMove.c
-rw-r--r-- 1 john john 101 Feb 20 21:27 initBoard.c
-rw-r--r-- 1 john john 843 Feb 20 21:27 is3inARow.c
-rw-r--r-- 1 john john 291 Feb 20 21:27 isAWin.c
-rw-r--r-- 1 john john 248 Feb 20 21:27 isBoardFull.c
-rw-r--r-- 1 john john 86 Feb 20 21:27 main.c
-rw-r--r-- 1 john john 156 Feb 20 21:27 makeMove.c
-rw-r--r-- 1 john john 374 Feb 20 21:27 max.c

```

```
-rw-r--r-- 1 john john 374 Feb 20 21:27 min.c
-rw-r--r-- 1 john john 919 Feb 20 21:27 minMax.c
-rw-r--r-- 1 john john 280 Feb 20 21:27 printBoard.c
-rw-r--r-- 1 john john 1327 Feb 20 21:27 runGame.c
-rw-r--r-- 1 john john 607 Feb 20 21:27 tictactoe.h
```

```
john@oho:~/LAB5$ cat Makefile
# Makefile
```

```
SOURCE=\
    getComputerMove.c\  
    getHumanMove.c\  
    initBoard.c\  
    is3inARow.c\  
    isAWin.c\  
    isBoardFull.c\  
    makeMove.c\  
    max.c\  
    min.c\  
    minMax.c\  
    printBoard.c\  
    runGame.c
```

```
CFLAGS = -m64 -g -w
```

```
OBJS = $(SOURCE:.c=.o)
```

```
tictactoe: $(OBJS) main.o
    gcc main.c $(CFLAGS) $(OBJS) -lm -o $@
    ls -l tictactoe
```

```
clean:
    rm *.o tictactoe
```

```
john@oho:~/LAB5$ make
cc -m64 -g -w -c -o getComputerMove.o getComputerMove.c
cc -m64 -g -w -c -o getHumanMove.o getHumanMove.c
cc -m64 -g -w -c -o initBoard.o initBoard.c
cc -m64 -g -w -c -o is3inARow.o is3inARow.c
cc -m64 -g -w -c -o isAWin.o isAWin.c
cc -m64 -g -w -c -o isBoardFull.o isBoardFull.c
cc -m64 -g -w -c -o makeMove.o makeMove.c
cc -m64 -g -w -c -o max.o max.c
cc -m64 -g -w -c -o min.o min.c
```

```
cc -m64 -g -w -c -o minMax.o minMax.c
cc -m64 -g -w -c -o printBoard.o printBoard.c
cc -m64 -g -w -c -o runGame.o runGame.c
cc -m64 -g -w -c -o main.o main.c
gcc main.c -m64 -g -w getComputerMove.o getHumanMove.o initBoard.o is3inARow.o isAWin.o
isBoardFull.o makeMove.o max.o min.o minMax.o printBoard.o runGame.o -lm -o tictactoe
ls -l tictactoe
-rwxr-xr-x 1 john john 39416 Feb 20 21:27 tictactoe
```

```
john@oho:~/LAB5$ tictactoe
```

```
^C
```

```
john@oho:~/LAB5$ gdb tictactoe
```

```
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04) 9.2
```

```
Copyright (C) 2020 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 "x86_64-linux-gnu".
```

```
Type "show configuration" for configuration details.
```

```
For bug reporting instructions, please see:
```

```
<http://www.gnu.org/software/gdb/bugs/>.
```

```
Find the GDB manual and other documentation resources online at:
```

```
<http://www.gnu.org/software/gdb/documentation/>.
```

```
For help, type "help".
```

```
Type "apropos word" to search for commands related to "word"...
```

```
Reading symbols from tictactoe...
```

```
(gdb) run
```

```
Starting program: /home/john/LAB5/tictactoe
```

```
^C
```

```
Program received signal SIGINT, Interrupt.
```

```
runGame () at runGame.c:18
```

```
18          j++;
```

```
(gdb) layout next
```

```
john@baho ~$ LAB5
runGame.c
6      void runGame() {
7          int    i = 0;
8          int    j = 0;
9          int    k = 0;
10
11          // ERROR: Infinite loop. Comment out the entire while loop.
12          // Use "watch i" and "watch j" to see values change.
>13      while(i<100) {
14          if (i < 50) {
15              i++;
16          }
17          else {
18              j++;
19          }
20      }
21
22      printf("\nChoose X or O. O moves first !!\n\n");
native process 1941 In: runGame                                L13  PC: 0x80019cd
(gdb) run
Starting program: /home/john/LAB5/tictactoe
^C
Program received signal SIGINT, Interrupt.
runGame () at runGame.c:13
(gdb) bt
#0  runGame () at runGame.c:13
#1  0x00000000080011e6 in main (argc=1, argv=0x7ffffffdfb8) at main.c:5
(gdb)
```

(gdb) bt

#0 runGame () at runGame.c:18

#1 0x00000000080011e6 in main (argc=1, argv=0x7ffffffdfb8) at main.c:5

(gdb) l runGame

```
1  #include "tictactoe.h"
2
3  int COMP = 1;
4  int HUMAN = 0;
5
6  void runGame() {
7      int i = 0;
8      int j = 0;
9      int k = 0;
10
```

(gdb) l

```
11     // ERROR: Infinite loop. Comment out the entire while loop.
12     // Use "watch i" and "watch j" to see values change.
13     while(i<100) {
14         if (i < 50) {
```

```
15     i++;
16     }
17     else {
18     j++;
19     }
20     }
```

(gdb) **b 18**

Breakpoint 1 at 0x80019c9: file runGame.c, line 18.

(gdb) **frame**

#0 runGame () at runGame.c:18

```
18     j++;
```

(gdb) **p j**

\$1 = 1622339873

(gdb) **n**

```
13     while(i<100) {
```

(gdb) **n**

```
14         if (i < 50) {
```

(gdb) **p i**

\$2 = 50

(gdb) **n**

Breakpoint 1, runGame () at runGame.c:18

```
18     j++;
```

(gdb) **p j**

\$3 = 1622339874

(gdb) **n**

```
13     while(i<100) {
```

(gdb) **p j**

\$4 = 1622339875

(gdb) **watch i**

Watchpoint 2: i

(gdb) **watch j**

Watchpoint 3: j

(gdb) info break

Num	Type	Disp	Enb	Address	What
1	breakpoint	keep	y	0x0000000080019c9	in runGame at runGame.c:18
breakpoint already hit 1 time					
2	watchpoint	keep	y		i
3	watchpoint	keep	y		j

(gdb) n

```
14         if (i < 50) {
```

(gdb) ← Carriage Return repeats the previous command. In this case, the next command.

Breakpoint 1, runGame () at runGame.c:18

```
18         j++;
```

(gdb) n

Watchpoint 3: j

Old value = 1622339875

New value = 1622339876

runGame () at runGame.c:13

```
13     while(i<100) {
```

(gdb) n

```
14         if (i < 50) {
```

(gdb) !ls -l runGame.c

```
-rw-r--r-- 1 john john 1327 Feb 20 21:27 runGame.c
```

(gdb)!vi runGame.c

After editing the runGame.c function to avoid the infinite loop, you will need to recompile the programs by running “make” and rerunning the tictactoe program.

Upload gdb.txt File to Receive Credit to comp232.com

To receive credit for **Debugging GDB**, create a text file called gdb.txt and copy/paste the tic-tac-toe output that results in a cat game or you win game.

SFTP the gdb.txt file to your LAB5 directory on comp232.com.