

# CSC 374: Computer Systems 2, 2010 Fall, Assignment #5

Last modified 2010 Nov 12

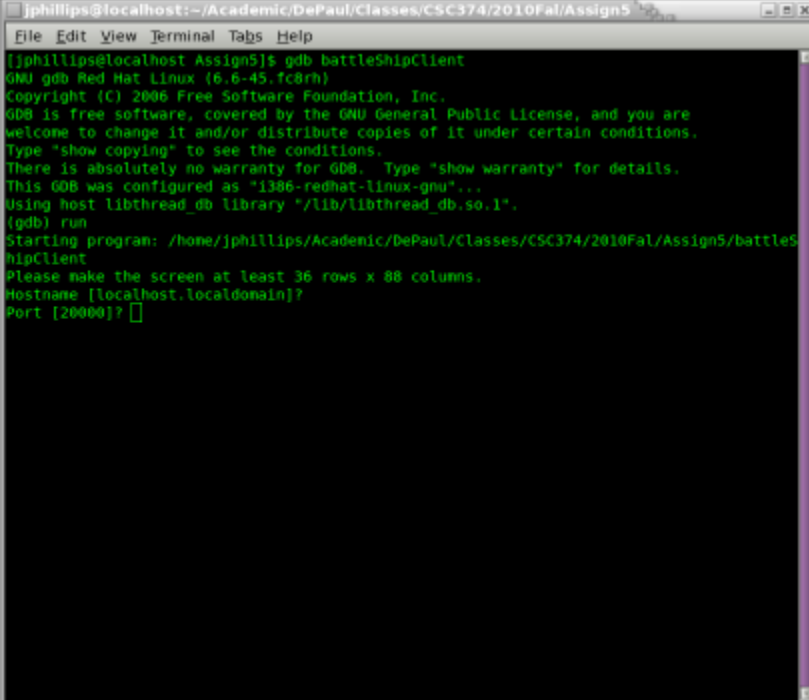
## Purpose:

To go over system calls related to sockets and cursor control with the ncurses package.

## Assignment:

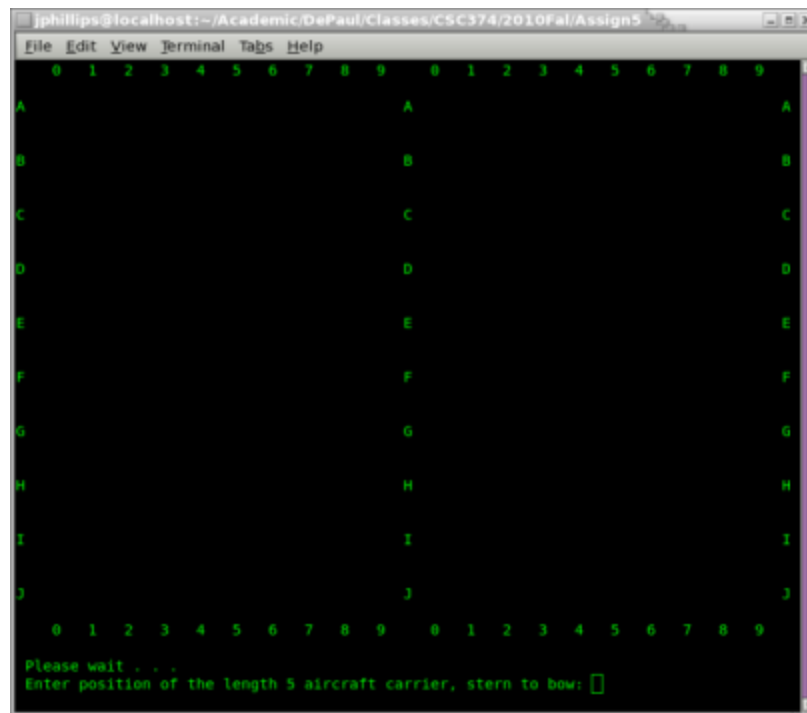
We will write the client program for "Battleship" which will:

- Ask for the name of the host and the port to which to connect,

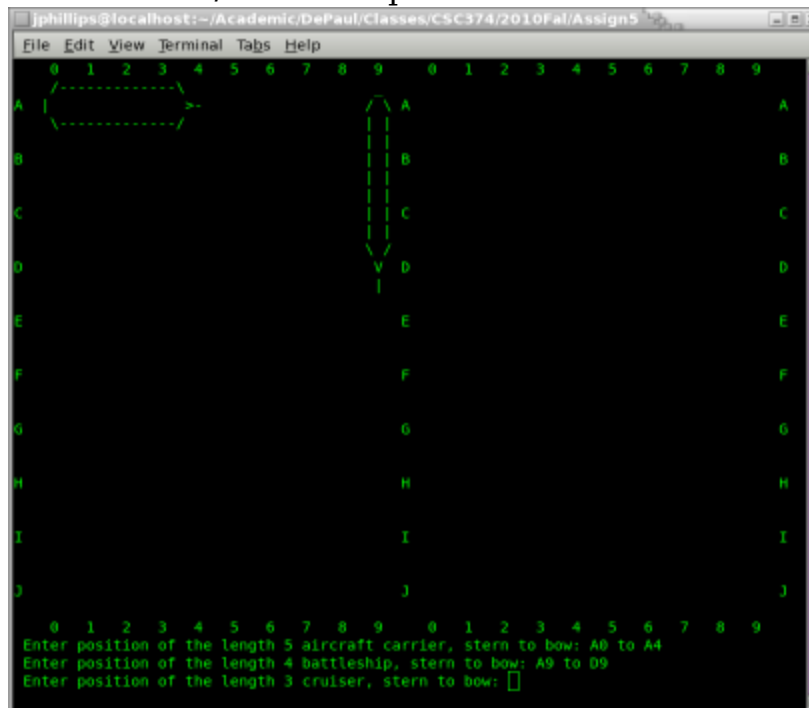
A screenshot of a terminal window titled "jphillips@localhost:~/Academic/DePaul/Classes/CSC374/2010Fall/Assign5". The terminal shows the execution of a GDB client program. The prompt is "[jphillips@localhost Assign5]\$ gdb battleShipClient". The GDB startup sequence is visible, including the GNU GDB version (6.6-45.fc8rh), copyright information, and the configuration path. The user enters the command "(gdb) run". The program starts and prompts the user to make the screen at least 36 rows x 88 columns. It then asks for the hostname, with "[localhost.localdomain]" shown, and the port, with "[20000]?" shown. The rest of the terminal is black.

```
jphillips@localhost:~/Academic/DePaul/Classes/CSC374/2010Fall/Assign5
File Edit View Terminal Tabs Help
[jphillips@localhost Assign5]$ gdb battleShipClient
GNU gdb Red Hat Linux (6.6-45.fc8rh)
Copyright (C) 2006 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for details.
This GDB was configured as "i386-redhat-linux-gnu"...
Using host libthread_db library "/lib/libthread_db.so.1".
(gdb) run
Starting program: /home/jphillips/Academic/DePaul/Classes/CSC374/2010Fall/Assign5/battleS
hipClient
Please make the screen at least 36 rows x 88 columns.
Hostname [localhost.localdomain]?
Port [20000]?
```

- Attempt to connect to the server using sockets
- Initialize the screen with ncurses to show both one's own fleet positions on the left board and what is known about the opposing fleet on the right
- Initialize and status text screen below both boards



- Let the user choose his/her fleet positions



- Play the game

```

jphillips@cdmlinux:~/classes/csc374/2010Fall/Assign5
File Edit View Terminal Tabs Help

 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
X X-X X-X X-X X\X X X X X X
A |X X X X >X /X\ A X X X X X X
X X-X X-X X-X X/X X X X X X X
X X X X X X X X X X X X X X X
B |X| B X X X X X X X X X X B
C |X| C X X X X X X X X X X C
D X D X X X X X X X X X X D
E X X X X X X X X X X X E
F F X X X X X X X X X X F
G G X X X X X X X X X X G
H H X X X X X X X X X X H
I |X| I X X X X X X X X X X I
J \X/ J X X X X X X X X X X J
X X X X X X X X X X X X X X X
You hit their battleship!
Ouch! They hit your submarine!
Where would you like to shoot (or 'Q') to quit?

```

- Inform the user about the outcome

```

jphillips@cdmlinux:~/classes/csc374/2010Fall/Assign5
File Edit View Terminal Tabs Help

 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
X X-X X-X X-X X\X X X X X X X X X X X
A |X X X X X >X /X\ A X X X X X X X X
X X-X X-X X-X X/X X X X X X X X X X
X X X X X X X X X X X X X X X X X
B |X| B X X X X X X X X X X B
C |X| C X X X X X X X X X X C
D X D X X X X X X X X X X D
E X X X X X X X X X X X E
F F X X X X X X X X X X F
G G X X X X X X X X X X G
H H X X X X X X X X X X H
I |X| I X X X X X X X X X X I
J \X/ J X X X X X X X X X X J
X X X X X X X X X X X X X X X
You hit their destroyer, and sunk the sucker!
Your fleet is battered but victorious!
The High Seas are yours, you win!

```

## Design of the Battleship client program

The Battleship client program uses 7 files:

- headers.h: Used by all .cpp files because it includes most of the required library headers, as well as files constants.h, declare.h, StatusDisplayWindow.h, and

BattleShipBoard.h. (**No need to edit this**)

- constants.h: Defines many constants (and few basic types) used through-out the program. (**No need to edit this**)
- declare.h: *Declares* functions that battleShipClient.cpp (and the server too) also use that are defined in battleShipCommon.cpp. Also defines some very simple inline functions. (**No need to edit this**)
- StatusDisplayWindow.h: Manages the text window used for text communication to/from the user and the server. (**You must complete this!**)
- BattleShipBoard.h: Manages the positioning of the fleet, the status of the fleet (who's been hit, who's still afloat), and the display of the fleet. (**You must complete this!**)
- battleShipCommon.cpp: Has miscellaneous functions used both by client and server for both socket communication and display. (**You must complete this!**)
- battleShipClient.cpp: Has the code directly relating to the operation of the client. (**You must complete this!**)

## What you need to do

1. Download 2010Fall\_CSC374\_Assign5\_code.zip from [Course On-Line](#)
2. Download the executable battleShipServer from [Course On-Line](#)
3. Upload them both to a Linux environment (if you are not already on one). Please use an sftp program like [Filezilla \(http://filezilla-project.org/\)](http://filezilla-project.org/) to do so. Just use:
  - a. *Host*: ctlinux1.cstcis.cti.depaul.edu
  - b. *Username*: (Your username)
  - c. *Password*: (Your CDM password)
  - d. *Port*: 22
4. Login to ctlinux1.cstcis.cti.depaul.edu
5. Unzip 2010Fall\_CSC374\_Assign5\_code.zip with:

```
linux$ unzip 2010Fall_CSC374_Assign5_code.zip
```

6. Edit the files with nano. The main commands are **Ctrl-O** to save (write-Out) and **Ctrl-X** to quit (eXit).
7. Start coding! Compile and link with:

```
linux$ g++ -g -c battleShipClient.cpp
linux$ g++ -g -c battleShipCommon.cpp
linux$ g++ -o battleShipClient battleShipClient.o battleShipCommon.o -lnurses
```

Look for the comments that say *YOUR CODE HERE*

8. **WARNING:** We are doing socket programming, and we need both clients and the server to be *on the exact same machine!* Please do *not* use cdmlinux.cdm.depaul.edu because it is too non-specific. Please use either:

- 3 ctilinux1.cstcis.cti.depaul.edu windows (1 server, 2 clients), or
- 3 ctilinux2.cstcis.cti.depaul.edu windows (1 server, 2 clients)

Please start the server first, then start up one client, then the other.