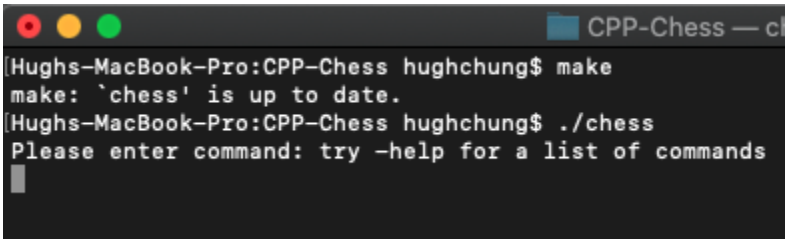
CS 246 Final Project Demo Document
Team Members: Diana Chung, Hugh Chung, Daniel Ephrat
d42chung@uwaterloo.ca, hy5chung@uwaterloo.ca , dephrat@uwaterloo.ca

Intro:

- Project type: Chess
- Language: C++

Compiling and Run:

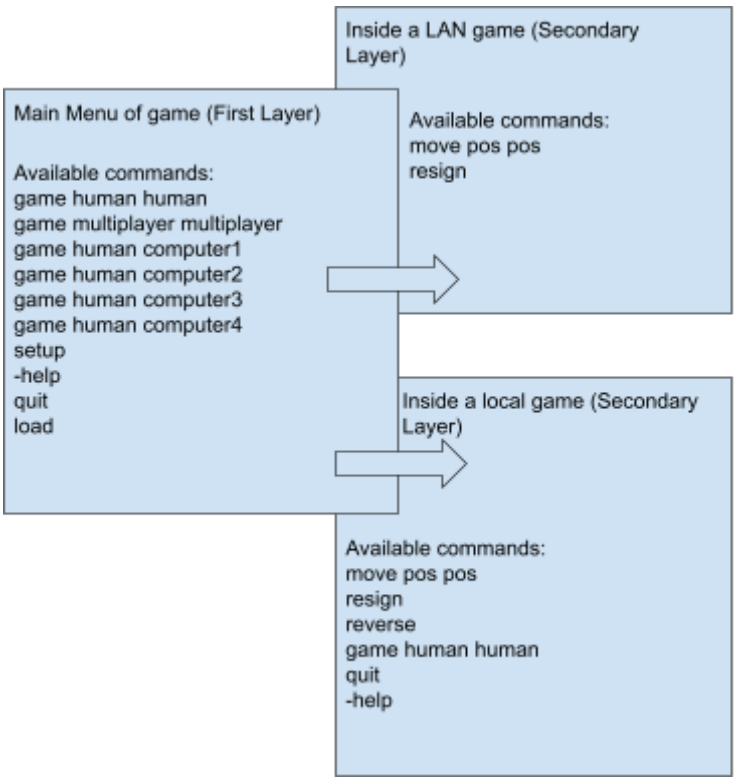
- We've included a Makefile to help you compile the program. To compile the program, simply type *make* in your terminal.
- The name of the program is chess. To run the program, type *./chess*



- The program will then prompt for input.
- Alternatively: ./chess -text (Disable Graphic Display)
- Alternatively: ./chess -seed xxx (xxx must be integer, sets the random number generator's seed to xxx.
- Alternatively: ./chess -text -seed xxx

Commands:

- The game is divided into two layers, each with its own set of commands.

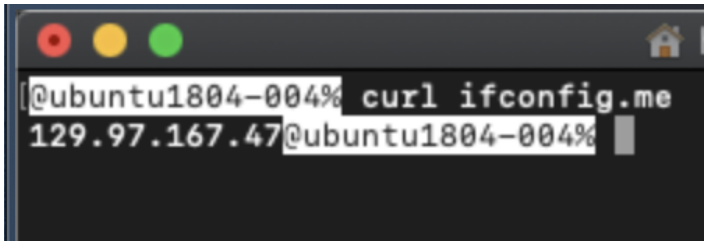


Commands in Main Menu:

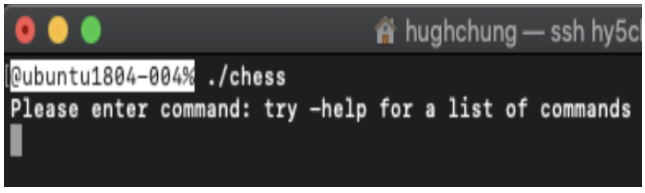
- **game human human:**
  - A new game begins. The first *human* indicates the white player, the second *human* indicates the black player.

```
Please enter command: try -help for a list of commands
game human human
8 rnbqkbnr
7 pppppppp
6  - - - -
5 - - - -
4  - - - -
3 - - - -
2 PPPPPPPP
1 RNBQKBNR

  abcdefgh

Please enter command: try -help for a list of commands
It is White's turn.
```

- **game multiplayer multiplayer: (Special Features)**
  - Enters LAN (Local Area Network) mode. This mode allows two machines (One host, one client) to play the chess game over the LAN.
  - Before starting a game, the host should first obtain his IP address. To allow the client to enter the game, the host must exchange his IP address with the client. **(Important)**

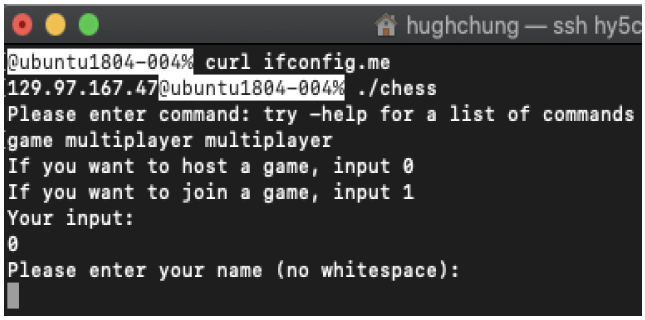


Demo 0: The host must provide the client with its IP address so that the client can connect to the game. The above command line is suggested for obtaining the IP address.
(*curl ifconfig.me)*



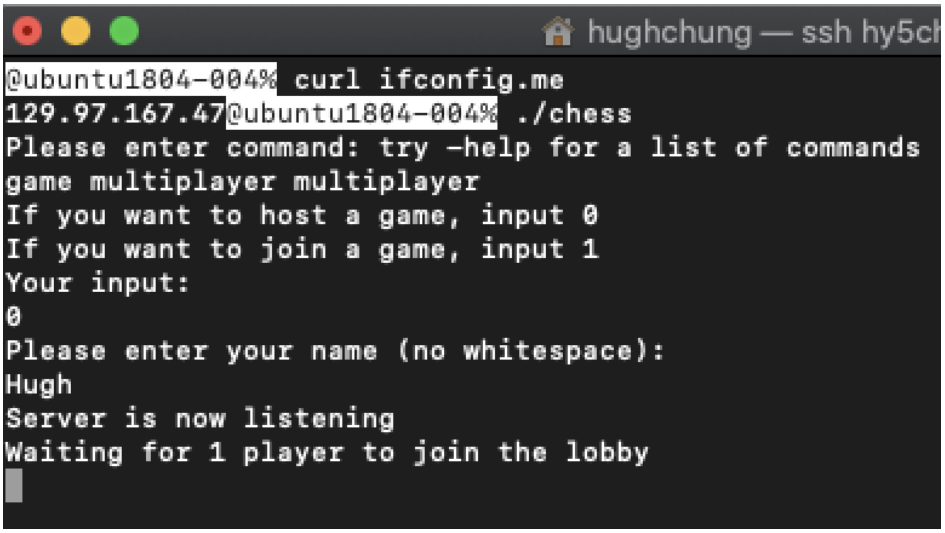

```
dephrat@ubuntu1804-010:~/cs246/randomFilesDir/chessFolder$ ./chess
Please enter command: try -help for a list of commands
```

Demo 1: Two players **(Black Terminal: Host, White Terminal: Client)** with different machines trying to play a chess game




```
dephrat@ubuntu1804-010:~/cs246/randomFilesDir/chessFolder$ ./chess
Please enter command: try -help for a list of commands
game multiplayer multiplayer
If you want to host a game, input 0
If you want to join a game, input 1
Your input:
1
Please enter your name (no whitespace):
```

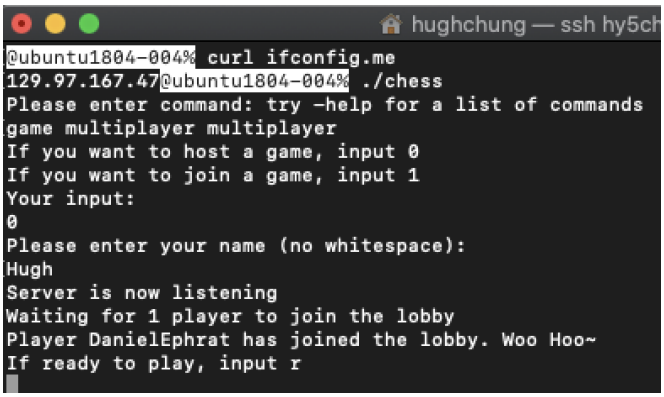
Demo 2: The game will then ask the player to either host or join a game. As well as an online player name. Host (left) inputs 0; Client (right) inputs 1. Right after is an option to enter your name.

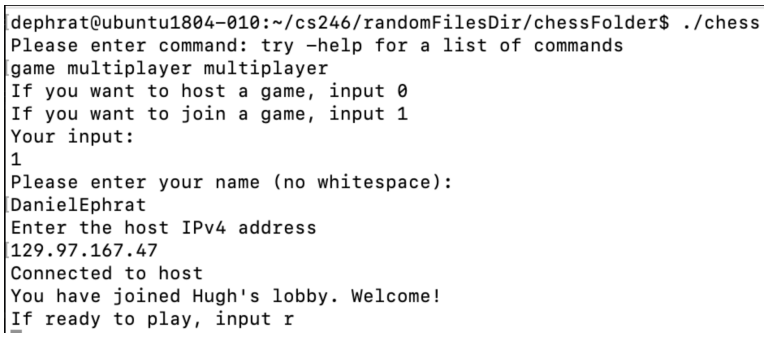


Demo 3: Before allowing others to join, a player must first host a game. **(Important)** As indicated, the host has now hosted a server and is waiting for players to enter.

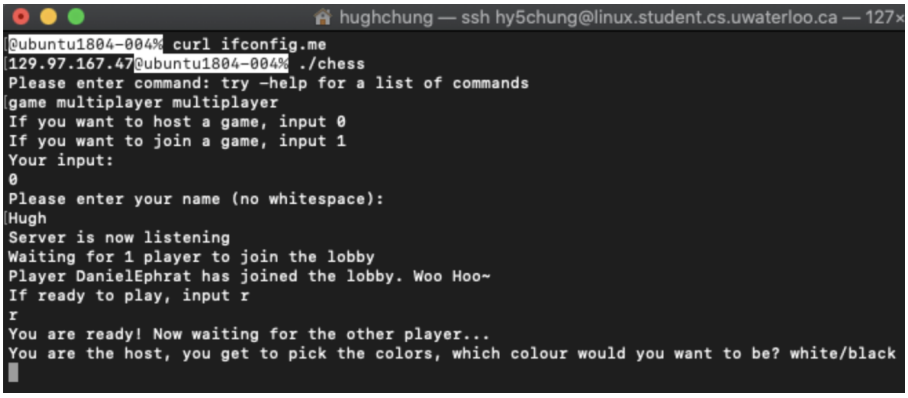
```
dephrat@ubuntu1804-010:~/cs246/randomFilesDir/chessFolder$ ./chess
Please enter command: try -help for a list of commands
game multiplayer multiplayer
If you want to host a game, input 0
If you want to join a game, input 1
Your input:
1
Please enter your name (no whitespace):
DanielEphrat
Enter the host IPv4 address
129.97.167.47
```

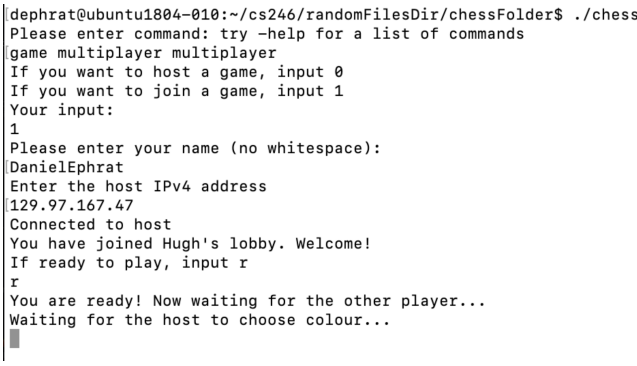
Demo 4: Once the host has started a game, the client shall join it. Entering his player name (DanielEphrat) as well as the host IP address.



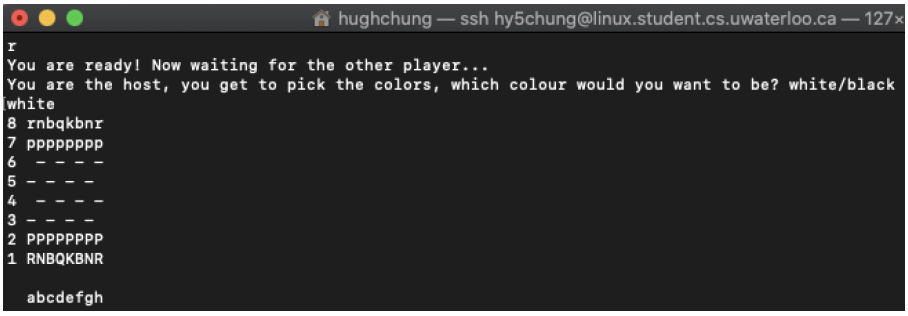


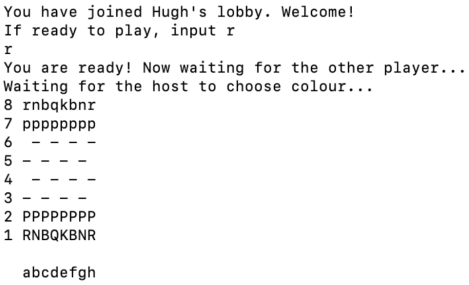
Demo 5: When the client enters the game, a welcome text will appear on both the host and client sides. Indicating that the client has established a connection to the host server and showing the player names on both terminals. The game will then ask for both players to input a single character *r*, if they are ready to play.



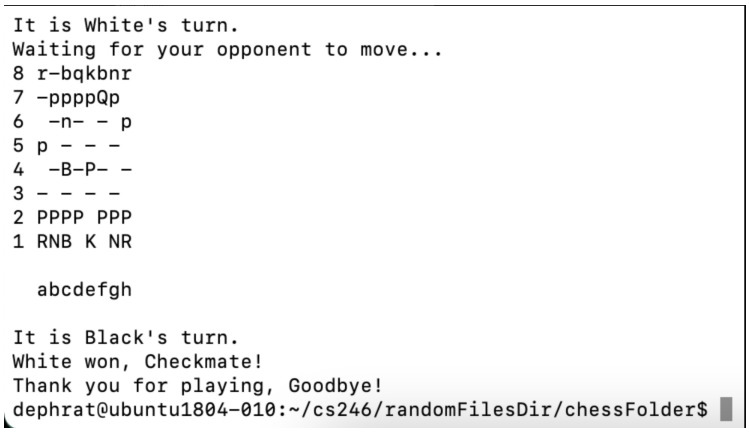


Demo 6: Once both players have entered a single character r, the host will select a color/side, and the client will wait for the host to select a colour.





Demo 7: The game starts after the host has finished selecting its colour.



Demo 8: When a game is completed, the program will immediately stop. This can be either through checkmate, stalemate, or either player's resignation. Players must restart the program in order to play again.

- **Important:** It should be noted that when an online game is running, neither the client nor the host should use ^C to exit the game. This will result in unstable port behaviour and can cause the port to become congested. As a result, starting a new network session again would cost undefined behaviour. More information can be found in the design document.
- Both players should only use (*resign*) to quit an online game.

- **game human computer1~4:**
  - This will launch a game of human vs. computer. Where 1/2/3/4 represents difficulty. 1 is the easiest, and 4 is the most difficult.
  - Level 1 plays random legal moves. Level 2 prefers captures and checks. Level 3 is like level 2, but is biased against moves that allow it to be captured. Level 4 is capable of checkmate, is slightly biased towards the center, and prefers checks and captures, but avoids captures that result in a negative trade (e.g. trading a queen for a bishop = giving up 9 points for 3 points, 3 - 9 = -6, so the computer avoids the trade)

- **-help:**
    - This will print out a help page with a list of commands.

- **setup:**
    - Commands: (Examples)
        - **+ K e1:** places the piece K (i.e., white king in this case) on the square e1. If a piece is already on that square, it is replaced. The board should be redisplayed.
        - **- e1:** removes the piece from the square e1 and then redisplays the board. If there is no piece at that square, take no action.
        - **= colour:** makes it colour's turn to go next.
        - **done:** leaves setup mode and starts the game.
        - **quit:** leaves setup mode but clears the board and back to the main menu.

- **load:**
    - When playing a local human vs. human game, a savelog ("savelog.txt") will be created for you when the game is exited in any way. The savelog acts as a history of all moves made in the game up until that point.
    - "load" brings the program back to the state it was in when the savelog was generated. So, if a user wants to replay the game starting from a certain position, they can use load to bring the board back to the state in savelog, then play on from there.
    - Loading can only be done when a game is not currently underway. Otherwise, the program gives you error messages, e.g. `"You are currently in a game. Resign or finish the game to load a savelog."`

- **quit:** Safely quits the program.

Commands during a chess game:

- **-help (Local Mode Only):**
  - This will print out a help page with a list of commands.
- **Reverse (Local Mode Only):**
  - Undo a move.
- **quit (Local Mode Only):**
  - Safely quits the program.
- **game human human (Local Mode Only):**
  - Restart a new game.
- **move pos pos:** A move consists of the command move, followed by the starting and ending coordinates of the piece to be moved. For example: move e2 e4. Castling would be specified by the two-square horizontal move for the king: move e1 g1 or move e1 c1 for white, move e8 g8 or move e8 c8 for black. If pawn promotion is detected, the game would additionally ask for the piece type to which the pawn is promoted: e.g. move e7 e8, where the piece originally at e7 is a white pawn. Note: If the player is a computer, the program automatically promotes to a queen.
- **Resign:**
  - Local game: concedes the game to your opponent, and back to the main menu.
  - LAN game: concedes the game to your opponent, and quit the program.