Rastin Rassoli Cellina Patricia

CS 246 - Project
"Demo"

Our chess program includes three main modes: Pregame, setup, and game. We will look through each mode step by step. Then we talk about the display and some other features. Invalid inputs and commands do not stop the program. An exception table is shown at the end of this doc. Please check it in order to understand the messages.

1) Pregame

Command	Functionality
results	Shows the current scoreboard
game username/computer[1-4] username/computer[1-4]	If usernames exist the game starts with those players, otherwise the players are created then the game starts
setup	Goes to the setup mode

The result command (extra feature) shows the scoreboard. The scoreboard includes the usernames and their score, and the total scores of Black and White colors. Note that only usernames with score higher than 0 are shown.

The game command starts the game with the given usernames (extra feature). Note that computers are considered as usernames. They will be shown in the scoreboards and real users with these usernames are not acceptable.

Figure 1. game command - napoleon.in

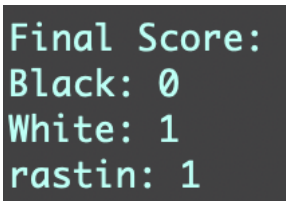


Figure 2. results command - napoleon.in

2) Setup

Command	Functionality
+ 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
= color	Makes it color's turn to go next
done	Goes to the pregame mode

For the setup mode the entire graphics board is updated for each command and the text observer is redisplayed each time. In case you want to remove the graphics board you should just comment line 15 in the chessview.cc (to make testing faster). The commands for this mode do not work anywhere else. Moreover, the setup is only applied to the next game. After the next game the game starts with the default setup. (Note: besides the next two tests, more tests are shown during other modes).

Figure 3. setup add command - computers.in

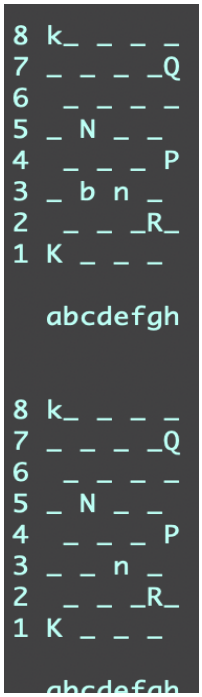


Figure 4. setup remove command - computers.in

3) Game

This mode is the main mode of the program and includes many commands, therefore we cannot show all of the commands in a table. However, the following extra commands (besides the normal commands) exist:

Command	Functionality
undo	Undos the last move
results	Shows the current result

As it was described earlier, if the client wants to use a default setup, they can directly start the game using the *game player1 player2* command. Player 1 and 2 should be either a username or computer[1-4]. If it is a human, the moves are customized by the client themselves. For computer players, there are 4 modes: level1, 2, 3, and 4. Level 1 does completely random legal moves. Level 2 allows capturing moves and checks over the other moves from the opponent. It first finds legal moves, then filters them by capturing moves. If there exist some legal capturing moves one of them is randomly chosen. If not, the available moves are filtered by checks. Again, if there are some moves one of them is randomly chosen. If no move exists a random legal move is chosen. Level 3 works similar but it also checks for the availability of avoiding capture moves. Level4 (extra feature) besides other moves, checks whether a checkmate move exists or not.

The two extra commands are undo (extra feature) and results (extra feature) as shown in the table. Undo is the only command which could be done after moving by the current player's turn. In other words when it is white's turn and it moves a piece, although after the move it is black's turn, white still can use undo to redo its last move. Undos do not have a limit (extra feature). If white did an undo, with another undo it will be black's turn and it can redo its last move. This could be done until you reach the beginning of the game. After that undo is illegal. The result command is similar to the result command in the pregame mode.

Promotion move is an exception move for pawn. When the pawn reaches the last row, it could be promoted to a knight, bishop, rook, or queen (this command is case sensitive, Q is white's queen and q is black's queen). The default promotion, when the user does not specify the piece type to which pawn is promoted, is queen promotion.

The following pictures are some basic tests for this mode:

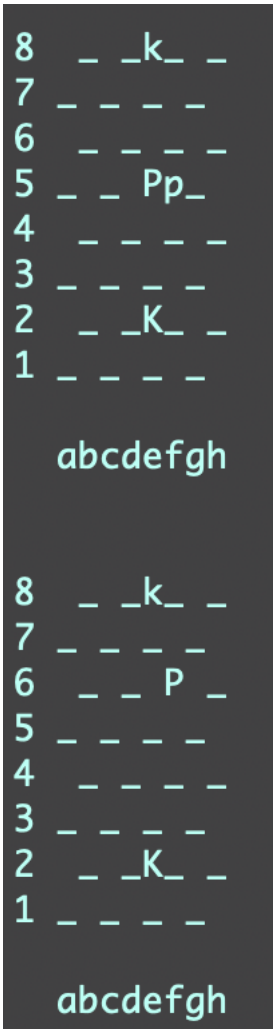


Figure 6. En Passant - enpassant.in

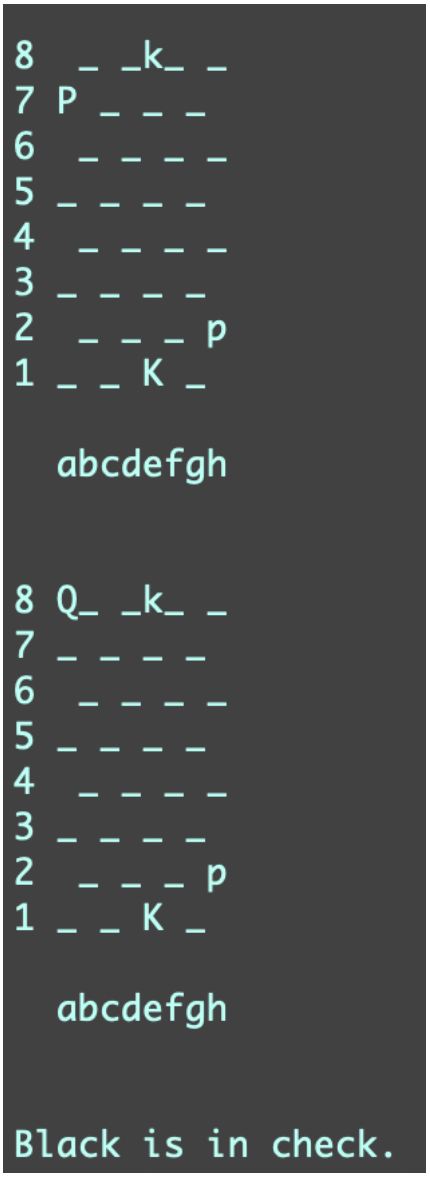


Figure 5. pawn promotion - promotion.in

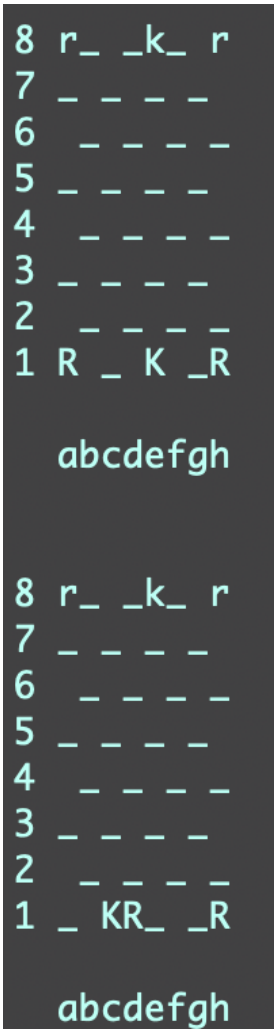


Figure 4. castle - castle.in

```
8 rnbqkbnr
7 pppp_ppp
6  _ _ _ _
5 _ _ p _
4  _ _P_ _
3 _ _ _ _
2 PPPP PPP
1 RNBQKBNR

  abcdefgh


8 rnbqkbnr
7 pppppppp
6  _ _ _ _
5 _ _ _ _
4  _ _P_ _
3 _ _ _ _
2 PPPP PPP
1 RNBQKBNR

  abcdefgh


8 rnbqkbnr
7 pppppppp
6  _ _ _ _
5 _ _ _ _
4  _ _ _ _
3 _ _ _ _
2 PPPPPPPP
1 RNBQKBNR

  abcdefgh
```

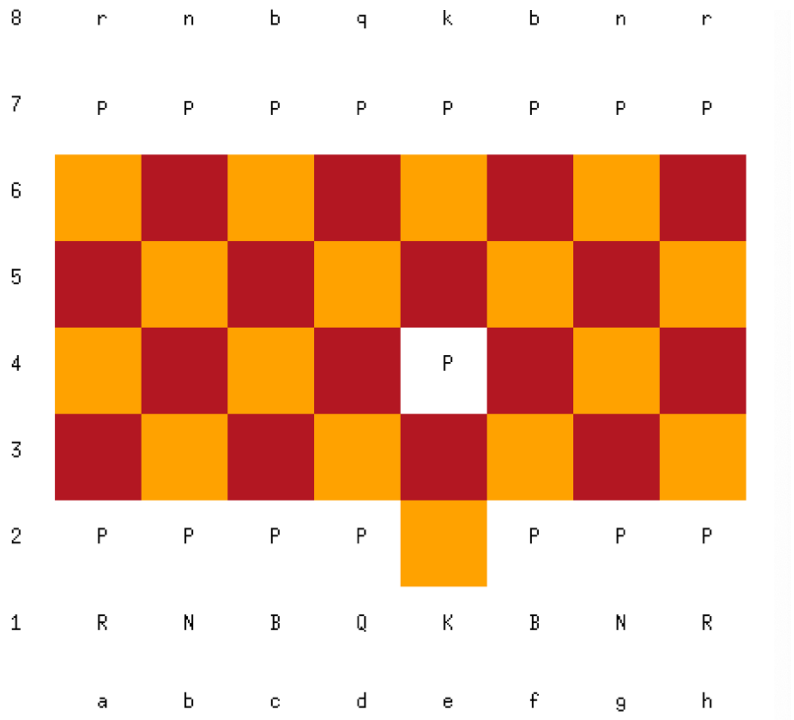
Figure 8. undo - undo.in

```
8 r_bqkbnr
7 _ppp_Qpp
6 p_n_ _ _
5 _ _ p _
4  _B_P_ _
3 _ _ _ _
2 PPPP PPP
1 RNB K NR

  abcdefgh


Checkmate! White wins!
```

Figure 8. checkmate - napoleon.in

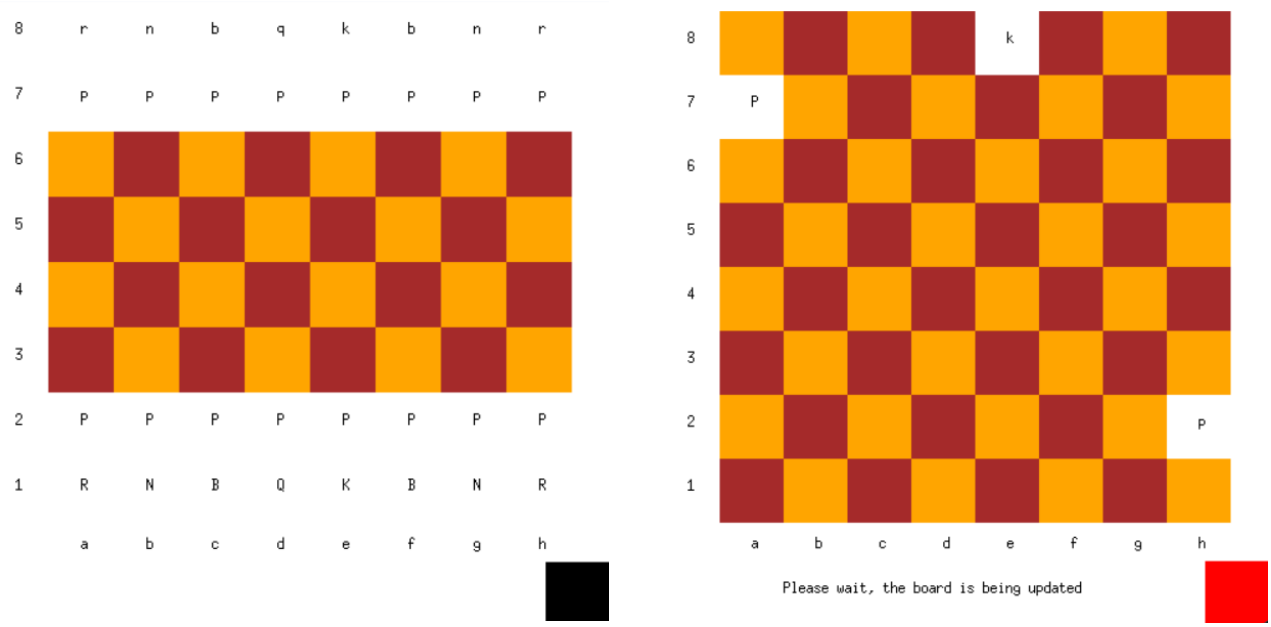
```
8 r_bqkbnr
7 pppp_Npp
6  _ _ _ _
5 _ _ p _
4  _ _P_ _
3 _ _ _ _
2 PPnP PPP
1 RNBQKB_R

  abcdefgh


White is in check.
8 r_bqkbnr
7 pppp_Npp
6  _ _ _ _
5 _ _ p _
4  _ _P_ _
3 _ _ _ _
2 PPQP PPP
1 RNB KB_R

  abcdefgh
```

Figure 7-9. game1.in



4) Display

The game is represented graphically with strings and squares as below:



The pieces are represented with only strings, while the rest are squares of orange and brown. The left-most column is the coordinate to allow more convenience for the client to identify the position of pieces, and so the bottom alphabetical row. During moves, only the last move will be updated not the entire board. When updating the board takes time, the square at the right-bottom-corner will blink black-red (extra feature) to signify the board is updating. Due to some implementations, there are times where the board is re-rendered (setup and undo) from the beginning even if there are no changes at certain positions. This might confuse the clients as if the board is stuck, that is the existence of the blinking square.

5) Extra Features

As you have already noticed the extra features are shown above in the parenthesis. The full list is:

- Usernames
- Scoreboards for Usernames
- Results in the game and pregame mode

- Computer level 4
- Updating square
- Undo
- Unlimited Undo
- Smart Pointers with no leak

```
==66192== Memcheck, a memory error detector
==66192== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==66192== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==66192== Command: ./chess
==66192==

Final Score:
Black: 0
White: 0
==66192==
==66192== HEAP SUMMARY:
==66192==     in use at exit: 0 bytes in 0 blocks
==66192==   total heap usage: 490 allocs, 490 frees, 160,353 bytes allocated
==66192==
==66192== All heap blocks were freed -- no leaks are possible
==66192==
==66192== For lists of detected and suppressed errors, rerun with: -s
==66192== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 1 from 1)
```

6) Exception Guide

Error Message	Explanation
not a valid move, you are/will be in check!	After the move your king will be in check or the move does not block the check
undo illegal	It is the beginning of the game
not a computer turn	Move command is not valid here
no piece detected	No piece detected in the given position
invalid color	The chosen piece does not match the turn
no move detected	From = To in move
invalid move	Illegal move
current game mode is setup (or game)	Invalid command, current mode is shown
not a valid position	Position is out of range
no pawn detected at the given position	
promotion to rook, bishop, knight, or queen is valid	Wrong promoting piece name
invalid promotion	
promoted and piece color do not match	Pawn cannot be promoted to another color (white turn: move e7 e8 q is not valid, but move e7 e8 Q is valid)
invalid king/castle move	
two player is required	Players are not registered
white and black kings must be in the game	Setup mode should include both the white and the black king
kings cannot be in check	Kings should not be in check in the Setup
pawns cannot be in the first or last rows	(In the setup mode)
there should be exactly one white and black king in the game	(In the setup mode)
computer turn	It is the computer's turn

All the tests mentioned above are included in the submitted zip file. We have tried to present some basic commands and functionalities; however, full coverage was not the purpose of these tests.