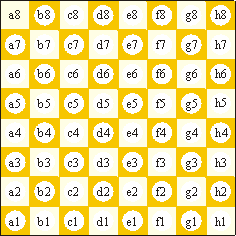
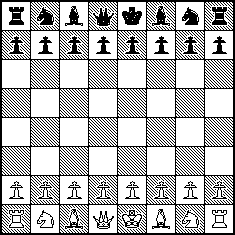
USER INTERFACE

The game is played on a chessboard, consisting of 64 squares: eight rows and eight columns. The squares are alternately light (white) and dark colored. The board must be laid down such that there is a black square in the lower-left corner. To facilitate notation of moves, all squares are given a name. From the view of the white player, the rows are numbered 1, 2, 3, 4, 5, 6, 7, 8; the lowest row has number 1, and the upper row has number 8. The columns are named, from left to right, a, b, c, d, e, f, g, h. A square gets a name, consisting of the combination of its column-letter and row-number, e.g., the square in the lower left corner (for white) is a1.





Alternately, the players make a move, starting with the white player (the player that plays with the white pieces.) A move consists of moving one of the pieces of the player to a different square, following the rules of movement for that piece - there is one special exception, named **castling**, where players move two pieces simultaneously.

A player can **take** a piece of the opponent by moving one of his own pieces to the square that contains a piece of the opponent. The opponents piece then is removed from the board, and out of play for the rest of the game.

ASSETS

IMAGES

* Images of different pieces
* Chess board (black and white)
* Background
* Players image

AUDIO

* Background sound
* Start, win, draw, lose game sound
* Pieces sound while moving or taking each turn.

LIST ALL FILES DESCRIPTION

|  |  |
| --- | --- |
| **File** | **Description** |
| [chessgui.h](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/chessgui.h) [chessgui.cpp](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/chessgui.cpp) | The chessgui files provide the Graphical User Interface; you probably will not have to edit these files. |
| [chess.h](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/chess.h) [chess.cpp](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/chess.cpp) | This is the GUI frontend that you will modify. It contains functions to access the GUI such as placing pieces, highlighting squares, dealing with user mouse clicks, etc. |
| [chessinclude.h](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/chessinclude.h) | The chessinclude file gives you constants that will be useful to allow you to access the image files and colors used by the ChessGui. |
| [messagebox.h](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/messagebox.h) [messagebox.cpp](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/messagebox.cpp) [inputbox.h](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/inputbox.h) [inputbox.cpp](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/inputbox.cpp) [buttonbox.h](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/buttonbox.h) [buttonbox.cpp](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/buttonbox.cpp) | Messagebox, inputbox, and buttonbox classes provide support for various simple dialog boxes. These functions are invoked using the Chess::messagebox, Chess::inputbox, and Chess::buttonbox functions. |
| [makefile](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/makefile) | This example makefile has been included which will compile all the files provided. Please look through the makefile. |
| [chess.tgz](https://faculty.cs.byu.edu/~rodham/cs240/chess/source/gui/chess.tgz) | A compressed tar file containing all the above files. You will need to download this file into your chess project directory and extract it. |

FUNCTIONS USED

* **void top\_label\_set\_text(const char \*text);**   
  This function sets the label at the top of the board.
* **void bottom\_label\_set\_text(const char \*text);**   
  This function sets the label at the bottom of the board.
* **void info\_area\_clear();**   
  This function clears the information area of the UI.
* **void info\_area\_put\_text(const char \*text);**   
  This function adds the text into the information area of the UI.
* **void input\_box\_clear();**   
  This function clears the input box of the UI.
* **const char \*input\_box\_get\_text();**   
  This function returns the information that is in the input box.
* **void place\_piece(int c, int r, int piece);**  
  This function places a piece on the board at row r and column c. piece is an int cooresponding to which piece to place on the board. A list of these values can be found in the chess.h include file.
* **void clear\_piece(int c, int r);**  
  This function clears a piece on the board at row r and column c.
* **void highlight\_square(int c, int r, int color);**  
  This function places a highlight on a given square at row r and column c. color is the color to set the tile to. It is a hexidecimal number with the format 0xRRGGBB. (Note: BLUE\_SQUARE defined in chessinclude.h is a good color to use).
* **void unhighlight\_square(int c, int r);**  
  This function removes a highlight from a given square.

BUGS

* the game doesn’t get recorded at times
* the game doesn’t resume where it is paused.

EXTENSIONS

* Add storage options to record more number of games.
* Link the storage options with drives.
* Add more background themes and add various colours.