

XXL Chess

Gameplay

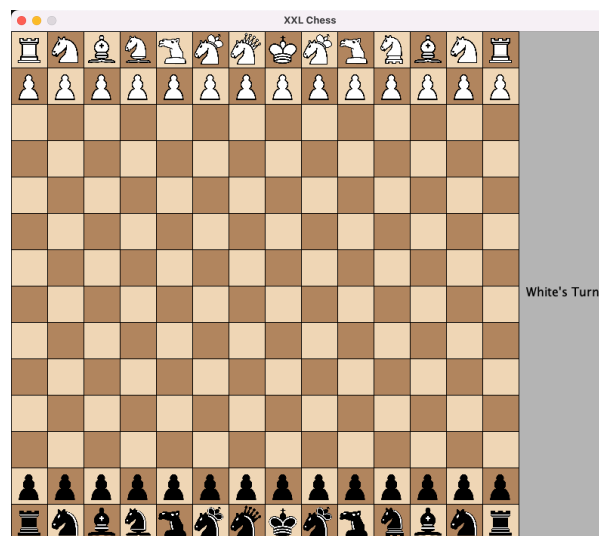
XXL Chess is a Java-based 2-player chess game that utilizes the Processing library for graphics and Gradle for dependency management. It aims to provide a classic chess experience with an expanded board and unique piece movements, challenging players to strategize and checkmate their opponent's king.

Requirements

- Java 8
- Gradle

Board

The game window is sized at 792x672 pixels, featuring a 14x14 grid of tiles. In addition to the board, a sidebar displays game information such as the current player's turn, stalemate conditions, or game over status.

























The board follows a traditional checkerboard pattern with alternating black and white tiles. Various tile highlights aid in gameplay clarity:

- Blue highlights indicate valid moves for a selected piece.
- Light red highlights show squares where a selected piece can move and capture another piece.
- Green highlights mark the player's currently selected piece.
- Yellow highlights denote the last moved piece.
- Dark red indicates squares where a king is under threat or where checkmate has occurred.

The initial arrangement of pieces is determined by a layout file specified in the "layout" attribute of the project's JSON configuration file.

Configuration

The game's configuration is stored in `config.json` in the project's root directory. This file specifies the name of another file containing the initial layout of chess pieces. Each character in the layout file represents a specific chess piece:

Sprites	Character	Chess Piece	Movement Description
 	P p	Pawn	Moves forward one square; captures diagonally forward.
 	R r	Rook	Moves horizontally or vertically any number of squares.
 	N n	Knight	2 squares in one direction, then 1 square perpendicular.
 	B b	Bishop	Moves diagonally any number of squares.
 	H h	Archbishop	Moves like a Knight or a Bishop.
 	C c	Camel	3 squares in one direction, then 1 square perpendicular.
 	G g	KnightKing	Moves like a Knight or a King.
 	A a	Amazon	Moves like a Knight, Bishop, or Rook.
 	K k	King	Moves one square in any direction; must avoid check.
 	E e	Chancellor	Moves like a Knight or a Rook.
 	Q q	Queen	Moves like a Bishop or a Rook.

Movement

The `piece_movement_speed` property in `config.json` determines the speed in pixels per frame at which pieces move. This speed is capped by `max_movement_time`, a limit in seconds that ensures moves are not too slow.

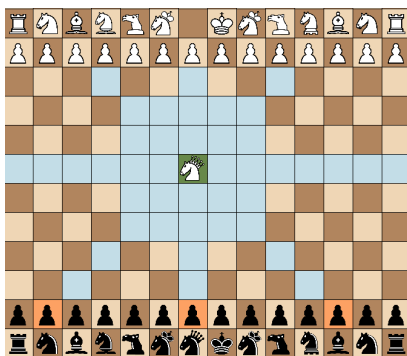
To execute a move, players first select a piece by clicking on its current position. Subsequently, they click on the destination tile where the piece should move. If a player clicks on one of their own pieces after selecting a piece, the new piece becomes selected instead. Clicking on an invalid move deselects the piece.

Special Moves:

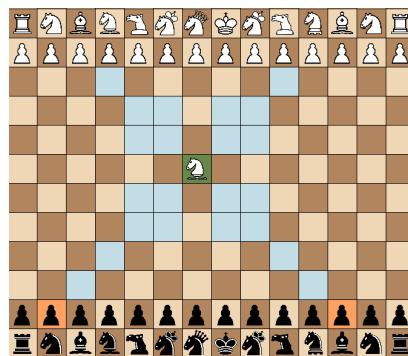
- **Pawn Initial Move:** A pawn can move two squares forward from its starting position if it is located on the 2nd row from the top or bottom of the board and has not moved before.
- **Pawn Promotion:** When a pawn reaches the 8th rank (crossing the halfway point on the 14x14 board), it is promoted to a queen. This promotion happens immediately upon reaching the 8th rank, allowing the pawn to function as a queen for all subsequent moves.

Only a camel or knight moves may jump over pieces, and a player may not move a piece onto a cell already containing one of their own pieces.

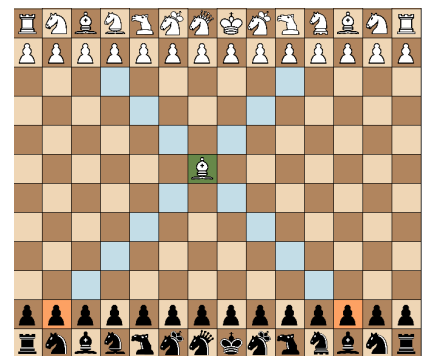
If a move causes the piece to enter a tile containing one of the opponent's pieces, the opponent's piece is "captured" and removed from the board. All pieces capture on the same tiles as their regular movement, with the only exception being pawns which capture diagonally forwards instead, if there is a piece there. This is the only time they are allowed to move diagonally. If there is a piece directly in front of a pawn, it is blocked and cannot move to the cell occupied by that piece.



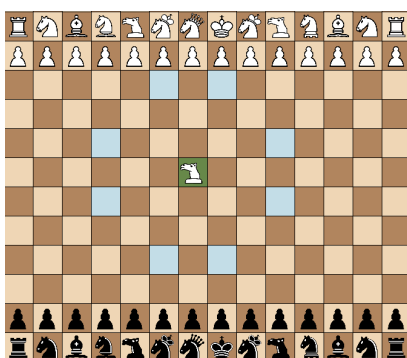
Amazon



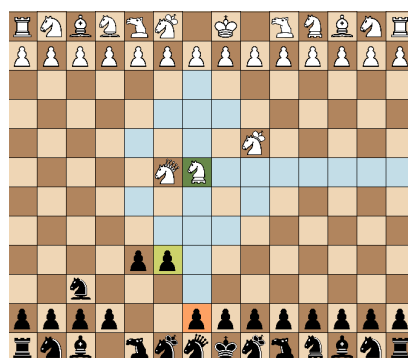
Archbishop



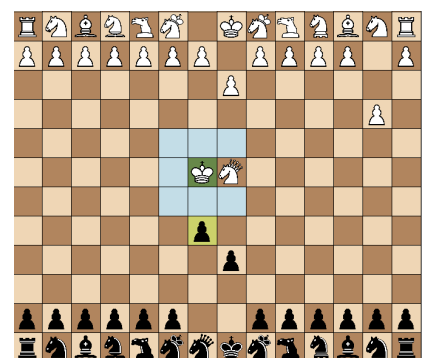
Bishop



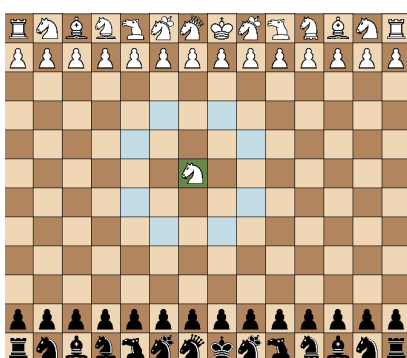
Camel



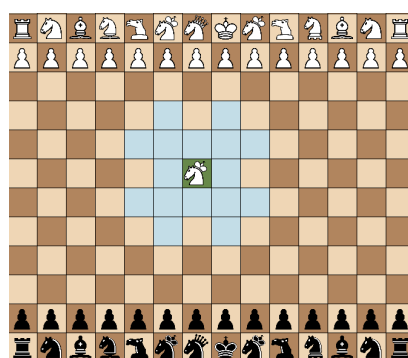
Chancellor



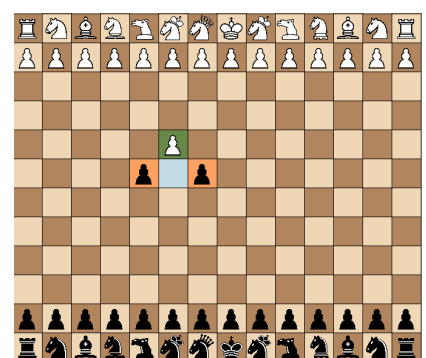
King



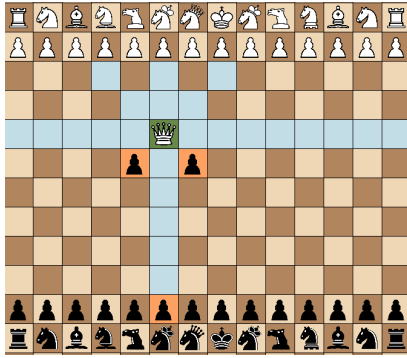
Knight



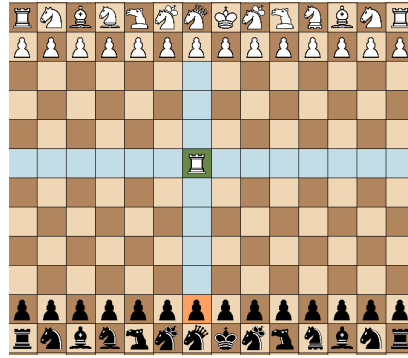
Knight King



Pawn



Queen

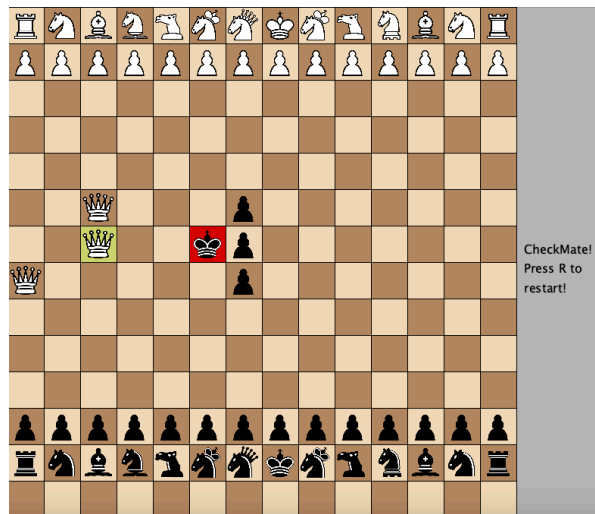


Rook

Check and Checkmate:

If after a move, a king is under attack, the king is said to be in 'check'. The player whose king is in check must do one of the following (all must already be legal moves):

- Move their king to a safe square.
- Move a piece to block the attack.
- Capture the attacking piece.



If none of these possibilities are available, then the player has been checkmated - there is no move available to them that would save their king, and they have lost. When check occurs, the king's square is highlighted in dark red.

A player cannot make a move that would result in their king coming under attack. This could be any of either:

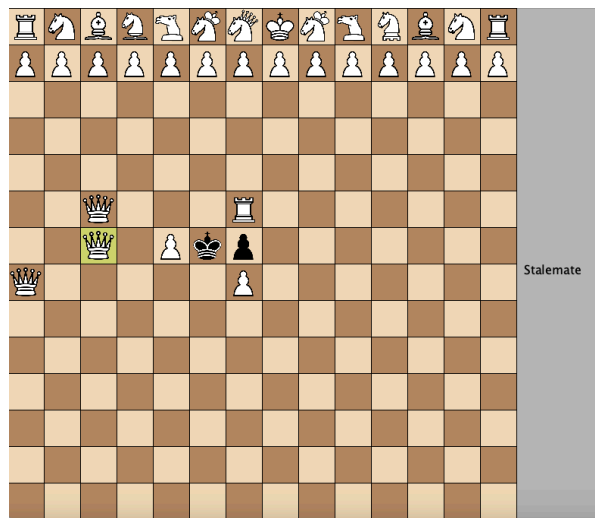
- Moving the king to a square which is under attack by the opponent.
- Moving a piece that is blocking an attack on their king by the opponent.

Win and Lose Conditions:

The game ends when either one player runs out of time, or their king is checkmated. The other player wins.

When the game ends, the board remains intact and frozen so that the player cannot make any moves (but may restart the game with the key press 'r').

If there are no legal moves for a player, then the game is considered a draw and enters the end state. Display the message “Stalemate - draw”.



Credits

- This project was created as part of an assignment for The University of Sydney.
- All resources and assets used in this project were provided by The University of Sydney.

Author

This chess game project was created by Fahim Faisal Osman.