

Final Project Proposal

Sentient Chess

Team National Swag Agency wants to code a sentient chess game engine.

Method of Attack:

1. We want to first create a fully functional chess game taking into account the specificities of chess.
 - a. Structurally, the various pieces will inherit a “piece” class, and we will also include a board class. A driver will run the game.
 - b. In the beginning, the board will be implemented in terminal, and we will then create a Gui. The board will be implemented using processing (as described below).
2. Then, we’d like to create and implement an algorithm so a computer can play chess with you.
3. After that, we would like to create a Plug-in Development Environment (.pde) that displays the chess board and shows a piece’s possible placements by clicking it
4. -----

Classes:

Piece wrapper

class for piece (pawn, knight, queen, king, rook, bishop)

-pawn needs a Moved boolean to detect if it can move up 2 places or not

- also pawn needs a separate attack vs. move

Board for terminal

Board for processing

Algorithm/engine class

Driver class

Piece

Things we need to account for:

Check vs checkmate

- We need a possible moves array per piece. If kings’ array empty, checkmate, else if in attack, must be moved

Pinning

- If piece in kings way, cannot move

En passant

- If pawn moves up two places and is adjacent to other pawn, append possible moves array to include the attacking of the pawn

Castling

- King also needs a move boolean, if it hasnt moved and the squares separating the rook and the king are clear (and the king will not be exposed after the fact), castling is a possible move

Piece movement

- If king gets exposed, or if other pieces in the way, it cannot move further

Processing the possible moves, showcasing that on the board, detecting which moves are under

Concepts used:

Stack, 2D array traversals, algorithm, subclasses + inheritance, instance variables, processing