I created my submission for project one which was a chess game. In that submission, you could play a full chess game using the console and verse stockfish, the best chess engine in the world currently.

In this submission, I’ve built upon the same structure of the chess game, and have added some more features, including a GUI, an opening book(using derby database) , minimax, and maximin, and a chess clock.

The GUI is simplistic and robust, drawing a chessboard like the chessboard from other chess programs such as <https://chess.com>, the chessboard GUI includes possible legal moves, signals possible captures.

Included in this chess program is a minimax algorithm which is used to calculate the best move for a position within a game. What this does is for a player, always selected the move that is the least destructive for the player, or the best move. In my project, I also utilise alpha-beta pruning, which deduces if a path to go down would not be a path either player would be willing to take, and if done right, reduces the search tree significantly. Alpha-beta pruning works best if the possible paths to take were in order of which is the best move and which isn’t, obviously we don’t know what the best move is because we haven’t searched yet; but included in the code are 2 ways to order moves to try to search for the better moves first. If we reach a position we haven’t searched before, then we can rank the moves based on characteristics of the moves, such as if the move is a capture, the value of the piece we are moving, if the move puts the opposing side in check etc. included in the program is a position tree with a doubly linked list. The linked list is sorted by the evaluation of each subsequent position. Which allows easy access to the best move using the head (whites best move) and tail (blacks best move) links. This way already searched positions can sort each subsequent move by the evaluation from the previous search.