Kasey Davis

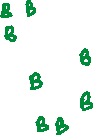
Kathrine Gibson

AI Explanation



1. When the AI is tasked with making a move, it is given the current gameboard.

In this example, the AI is black:



1. It then finds a list of all moves than it could perform. The possible moves are shown here in green:
2. For each potential move, it clones the board, tests the move, and then performs the minimax function. The call to minimax informs us of the best possible score under each move. The minimax and AB pruning algorithm is otherwise identical to our previous implementation. The heuristic used is simply black’s score minus white’s score.
3. Now that each potential move has a score-potential, we select the move with the highest potential or the least, depending on whether the AI is ‘mini’ or ‘max’. This move is returned to the game loop manager.

AB Pruning Execution Example:

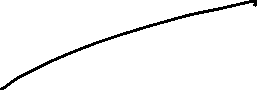
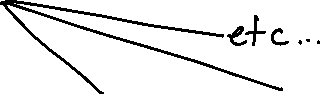
Assume a depth level of 2, for the sake of example. In our implementation, we used a depth limit of 4.











Heuristic: 0 H= 0 H= 2 H= 2

The AB Pruning is demonstrated here on C2, one of the possible moves that the AI can perform. C2 now holds the value for the most ideal score possible for that move. After all other moves, C1-Cn, are evaluated, the move with the highest heuristic value (or smallest if the AI is white/mini) is chosen and performed.