Report

Minimax with Alpha-Beta Pruning

The engine uses the minimax algorithm to search with the assistance of the alpha-beta pruning heuristic. The alpha-beta pruning was used to greatly reduce the computation time. The minimax algorithm was used on a game tree that represented every state the board could have after any future move. The maximizer represented the player while the minimizer represented the opponent. Every board state would have a value associated with it, a positive value would favor the maximizer, while a negative value would favor the minimizer. This is used so that the best would be made assuming that the opponent was making optimal moves.

Evaluation Function

The evaluation function would feature two variables, one for the player and the other for the opponent. If the player won the game, the score for the player would equal 100. If the opponent won the game, the score for the opponent would equal 100. The function would then go through the state of the board and count the amount of times the player and the opponent had three markers in a sequence. The amount would be times by ten and added to the player's or opponent's score. The function would then count the amount of times two markers would be in a sequence by either the player or the opponent. The amount would be added to the player's and/or opponent's score. The function would then return the score of the player minus the score of the opponent. If the result returned with a value of zero, the game was a draw. If the value returned negative, the state of the board would favor the opponent. If the value returned positive, the state of the board would favor the player.

Performance Testing

The perft function was used to count the number of nodes that are in a tree when expanded to a given depth from a current position. The function was tested on multiple states such as a full board and a board with only a few empty spaces to place a marker. The perft function could only count the number of nodes when expanded to a depth of seven with the root node being an empty board. When attempting to expand to a depth of eight, it takes far too long to compute.