Monte Carlo Chess Engine

Joshua Shim

November 2024

Board Representation

I have chosen to use bitboards to represent the state of the game in my chess engine. Bitboards in chess are 64 bit words, where each bit represents one square of the chess board. This is done such that the least significant bit represents the square $h1$ and the most significant bit represents the square $a8$. Hence we can imagine that the bits of the word go left to right then top to bottom in order of significance.

More specifically, I use an array of 12 bitboards, as there are 6 different types of pieces and 2 colours that each piece can be.(insert sc of code)

Furthermore, I require a couple flags regarding the special moves in chess, namely *en passant* and *castling*. *En passant* requires me to store the last move to check that one of my pawns have been passed by the opponent's pawns. *castling* requires me to store a flag that checks whether *castling* is possible as if the respective rook or king has moved or if the king is in check then *castling* is illegal.

Hence on top of the 12 bitboards, I have decided to keep two 8 bit words to keep track of what the last move was, where the first 4 bits of the first word indicates the rank and the last 4 bits the file of the piece that had moved originally was and where the second word similarly depicts the new position of the piece's new position. (insert sc of code)

Furthermore, I have decided to keep another 8 bit word to keep track of whether left and right castling is possible and whether to king is in check to both colours. (insert sc of code)

Move Generation

Move Selection