Artificial Intelligence: Project Search

1. Search strategy  
   1. Minimax algorithm  
      The main idea of the search strategy is the minimax algorithm. However, if we apply the minimax algorithm directly without any modification, the search time is unacceptable due to the size of the chess board. To solve this problem, we can evaluate the possible game results in advance and discard the worse situations first. As proposed by Shannon in his paper *Programming a Computer for Playing Chess*, we predict the game results by using an evaluation function which simply calculates the summation of the numbers which have not appeared in the chess board yet and the “safer” numbers in the chess board. According to the game rule, the larger number that a user has, the more chance he or she will win. Hence, we reach a conclusion that a user has an advantage whenever he or she has a larger number than his or her competitor.
   2. Iterative deepening  
      As described in Section 5.4.2 in the textbook, a better search method to make a move in the allocated time is to apply the iterative deepening. We gradually increase the search depth and record the best move in the deepest completed search. We use two constraints: one is the time constraint, which asks for making a move within 30 seconds, and the other one is the constraint on search depth. When time runs out (i.e., the search has lasted 30 seconds), the program returns the best move found so far.