# Minimax Algorithm Code and Explanation

Minimax Algorithm is a strategy used in decision-making and game theory, especially for two-player turn‑based games.   
The goal of this algorithm is to select the best possible move while assuming that the opponent will also play optimally.  
  
The game involves two players:  
• Maximizer: Tries to get the highest possible score.  
• Minimizer: Tries to reduce the score of the opponent.  
  
**How It Works:**

1. The algorithm evaluates the game tree starting from the root node.  
2. Each level represents one player's turn. The levels alternate between Max and Min.  
3. The leaf nodes contain predefined values that represent the final scores of the game.  
4. The algorithm moves backward from the leaf nodes, selecting the best score for the player whose turn it is.  
5. Finally, it returns the most optimal value from the root node, representing the best possible move.  
  
**Base Case:**

When the maximum tree depth is reached, the algorithm directly returns the score of the current node.  
  
**Recursive Case:**

• If it is Max Player's turn, the function returns the maximum of the possible values.  
• If it is Min Player's turn, the function returns the minimum of the possible values.  
  
This ensures that both players choose their best actions, resulting in the optimal game strategy.

This implementation demonstrates how Minimax selects the most optimal score for the current player.