**Game Trees and AI Agents**

**Understanding Game Trees**

A **game tree** is a graphical representation of all possible sequences of moves that can occur in a two-player game. Each node in the tree represents a game state, and the edges represent the possible moves that can be made from that state.

* **Root Node:** Represents the initial state of the game.
* **Leaf Nodes:** Represent terminal states, where the game ends.
* **Internal Nodes:** Represent intermediate states of the game.

**Minimax Algorithm**

The **minimax algorithm** is a decision-making algorithm used by AI agents to determine the optimal move in a two-player game. It works by assuming that the opponent will always make the best possible move.

1. **Create the game tree:** The algorithm starts by creating a game tree representing all possible game states.
2. **Evaluate terminal nodes:** The terminal nodes (game end states) are assigned a value based on the outcome of the game (e.g., win, lose, draw).
3. **Propagate values:** The values of terminal nodes are propagated up the tree.
	* **Maximizing Player:** At nodes representing the maximizing player's turn, the maximum value among its children is chosen.
	* **Minimizing Player:** At nodes representing the minimizing player's turn, the minimum value among its children is chosen.

**Alpha-Beta Pruning**

**Alpha-beta pruning** is an optimization technique that can significantly reduce the number of nodes that need to be explored in the game tree. It works by maintaining two values:

* **Alpha:** The best value found so far for the maximizing player.
* **Beta:** The best value found so far for the minimizing player.

If at any point during the search, the value of alpha becomes greater than or equal to beta, the search can be pruned because the current branch cannot possibly lead to a better outcome for the maximizing player.

**Example:** Consider a game where the maximizing player (A) wants to maximize its score, and the minimizing player (B) wants to minimize A's score. If A finds a move that guarantees a score of 5, and B later finds a move that can limit A's score to 3, there's no need for A to explore any other moves in that branch, as B's move is already better for A.

By using minimax and alpha-beta pruning, AI agents can make informed decisions in two-player games, often achieving a competitive level of play.