Lab 8 – Minimax Algorithm with Alpha-Beta Pruning

Overview

This lab implements the Minimax Algorithm with Alpha-Beta Pruning. This is used in decision-making, especially in games like chess and tic-tac-toe.

Working

1. Minimax simulates all possible moves.
2. Maximizer tries to maximize value.
3. Minimizer tries to minimize value.
4. Alpha-Beta pruning eliminates branches that cannot influence the final decision.
5. Recursion continues until reaching leaf nodes.

Code Explanation

- minimax() accepts parameters depth, index, turn, values, maxDepth, alpha, beta.
- If depth == maxDepth → return leaf value.
- Maximizer updates alpha; Minimizer updates beta.
- Pruning occurs when alpha >= beta.
- Final return gives optimal decision value.

Importance

Alpha-Beta pruning drastically reduces the number of states explored, improving performance.

Conclusion

The minimax algorithm with alpha-beta pruning efficiently determines the optimal game decision.