Task 2

TIC-TAC-TOE AI Implement an AI agent that plays the classic game of Tic-Tac-Toe against a human player. You can use algorithms like Minimax with or without Alpha-Beta Pruning to make the AI player unbeatable. This project will help you understand game theory and basic search algorithms.

Implementing a Tic Tac Toe game where a human player competes against an AI. The AI utilizes the Minimax algorithm to determine its best moves.

The code is structured into several functions:

* **Board Printing:** Displays the current state of the game board.
* **Win Checking:** Determines if a player has won.
* **Board Fullness:** Checks if the board is completely filled.
* **Empty Cell Identification:** Locates empty cells on the board.
* **Minimax Algorithm:** Implements the Minimax algorithm to calculate the best move for the AI.
* **Best Move Finding:** Uses the Minimax algorithm to determine the AI's optimal move.
* **Game Control:** Manages the overall game flow.

The Minimax algorithm simulates various possible moves and their outcomes, selecting the move that maximizes the AI's chances of winning. The game continues until a player wins or the board is full.

Here's a breakdown of the game's flow:

1. **Initialization:** The game board is initialized, and the current player is set to the human.
2. **Board Display:** The current state of the game board is displayed.
3. **Human Move:** The human player is prompted to enter their move (row and column).
4. **Move Validation:** The validity of the move is checked (i.e., if the cell is empty). If the move is invalid, the human player is asked to enter a new move.
5. **Board Update:** The game board is updated with the human player's move.
6. **Win Check:** The game is checked for a winner. If a player has won, the game ends, and a message is displayed.
7. **Board Fullness Check:** The game board is checked for fullness. If the board is full, the game ends, and a message is displayed.
8. **AI Move:** The AI uses the Minimax algorithm to determine its best move and updates the game board.
9. **Board Display:** The updated game board is displayed.
10. **Game Loop:** Steps 3-9 are repeated until the game ends.

The Minimax algorithm is a recursive algorithm that explores all possible game states and chooses the move that leads to the best outcome for the AI. It works by assuming that the opponent will make the best possible move and then selecting the move that maximizes the AI's chances of winning.