Group members:

Djeghdjegh Sara

Djait ikram

Programming project report:
tic-tac-toe with Minimax algorithm

Table Contents:

Introduction:

This project implements a tic-tac-toe game, using the python programming language. This game allows a user to play against the computer (AI) which use Minimax algorithm.

Implemented Features:

1. **Board Display:**
   - The **print_board** function is used to display the game board at each stage of the game.
2. **Winner Detection:**
   - The **is_winner** function detects if a player has won by checking rows, columns, and diagonals on the board.

3. **Full grid detection:**
   - The **is_board_full** function checks if the grid is entirely filled without a winner.

4. **Empty Cells retrieval:**
   - The **get_empty_cells** function returns the coordinates of empty cells on the board.

5. **Minimax algorithm:**
   - The Minimax algorithm is used to enable the AI to make optimal decisions by evaluating all possible moves.

6. **AI's best moves retrieval:**
   - The **get_best_move** function uses the Minmax algorithm to determinate the best move for the AI.

7. **User interface:**
   - The main user interface in the main function allows the user to play against the AI.

Game Operations:

1. The user and the AI take turns making moves on the board.
2. The player can input the coordinates of their move (row and column) When it’s their turn.
3. The game checks if the move is valid; otherwise, it prompts thee user to try again.
4. The AI uses the Minmax algorithm to determine the best possible move.
5. The game continues until there is a winner or the grid is full(draw).

Conclusion:

The project has been successfully implemented, using the Minmax algorithm to create an Ai capable of playing tic-tac-toe optimally. It provides an interactive gaming experience for the user wile showcasing the use of programming concepts such as loops, conditions, functions, and array handling.