

ARTIFICIAL INTELLIGENCE: FOUNDATIONS AND APPLICATIONS (AI61005)

**Autumn 2025
IIT Kharagpur**

Release Date: 24/08/2025

Submission Date: 10/10/2025

Problem Statement - 3 (Tic-Tac-Toe AI)

Tic-Tac-Toe is a well-known two-player game played on a 3×3 grid, where players alternate placing their marks (X or O) to align three in a row horizontally, vertically, or diagonally. In this assignment, you will implement an AI opponent that plays optimally against a human player using two classical algorithms:

- **Minimax Algorithm**
- **Alpha-Beta Pruning**

Game Rules and Assumptions

- The board initializes empty.
- The human player always plays first and uses the symbol 'X'.
- The AI uses the symbol 'O'.
- Moves alternate between players.
- The game concludes when either player achieves three in a row (horizontally, vertically, or diagonally), or when all positions are filled resulting in a draw.

Input Format

- At the start of the game, allow the user to select the AI algorithm (Minimax or Alpha-Beta Pruning) that the AI will use for all its moves during the game.
- On the human player's turn, prompt for move coordinates (x, y) (0-based indices).
- Validate inputs to ensure moves are within range and that the selected cell is empty. If invalid, prompt again.
- The AI computes and executes its move using the selected algorithm until the game ends.

Output Format

- After every move by either player, display the current board state.
- At the end of the game, display:
 - a. The final board.
 - b. Game result: Player wins, AI wins, or Draw.

- c. Total time taken by the AI across all moves (in milliseconds, rounded to two decimal places).
- d. Total number of recursive calls made by each algorithm.

Testing Instructions

- You must simulate and document three complete games to illustrate AI performance across varied outcomes:
 1. **Game 1:** Human wins (if possible)
 2. **Game 2:** AI wins
 3. **Game 3:** Draw
- If it is not possible for the human to win when playing optimally against a perfect AI (Minimax/Alpha-Beta), document attempts and explain why the AI prevents human victory.

Submission Instructions

Each group must submit the following:

1. Source Code

- A single, well-commented code file implementing both algorithms.
- The code must be written in Python.

2. Report (PDF) that includes:

- Names and roll numbers of all group members.
- Brief explanation of each algorithm (outlining completeness, optimality, time, and space complexity).
- A results summary table presenting performance metrics for all test games:

Test Case	Algorithm	Winner	Recursive Calls	Time Taken (ms)
Game 1	Minimax	Draw	112	13.52

- Your observations:
 - Why is Alpha-Beta typically more efficient?
 - In what scenarios do both perform similarly?

3. Output (output.txt)

- Include a plain text file containing the complete console output of all three games (Game 1, Game 2, and Game 3).

***Note:** If any GenAI tools (e.g., ChatGPT, Copilot, Gemini) were used for code, writing, or visualization, you must explicitly declare which parts were assisted by AI.*