**🎯 Assignment: Understanding the Tic-Tac-Toe Problem in AI**

**🧠 What is the Tic-Tac-Toe Problem?**

Tic-Tac-Toe is a simple two-player game played on a 3x3 grid. Players take turns placing their symbol (usually **X** and **O**) in empty squares. The first player to get **three of their symbols in a row** — horizontally, vertically, or diagonally — wins. If all 9 squares are filled without any player winning, the game is a **draw**.

In Artificial Intelligence, **Tic-Tac-Toe** is often used to teach:

* Game playing agents
* Search trees
* Minimax algorithm
* Utility-based decision making

**🧩 Components of a Tic-Tac-Toe Search Problem**

* **Initial State**: Empty 3x3 board.
* **Actions**: Place X or O in any empty cell.
* **Goal Test**: Check if the player has won (3 in a row) or if the board is full (draw).
* **Path Cost**: Usually not considered here, but each move could be counted as one step.
* **Type**: Goal-based problem, deterministic, fully observable.

**📘 Tasks**

**Task 1: Representing the Game**

1. Draw an initial empty 3x3 board.
2. List all possible first moves for Player X.
3. How many possible unique board states are there after the first move?

**Task 2: Game Tree Exploration**

1. Create a small **search tree** of Tic-Tac-Toe up to depth 2:
   * Root node: initial board
   * Depth 1: All possible moves by X
   * Depth 2: All possible responses by O
2. Label each node with the board state.

**Task 3: Classifying the Problem**

Answer the following questions:

1. Is this a **goal-based agent** problem?
2. Is Tic-Tac-Toe a **deterministic** game? Why?
3. Is it a **fully observable** environment?
4. Is it a **single-agent** or **multi-agent** problem?

**Task 4: Python Mini-Project**

Implement a simple **Tic-Tac-Toe game** in Python:

* Two-player mode (X and O input manually)
* Show the board after every move
* Detect win or draw conditions

(*Optional*: Use a 2D list and functions like check\_win(board))

**✅ Submission Guidelines**

* Submit your answers to conceptual questions as a Word doc.
* Python code (if attempted) with the Word doc must be pushed to your github repositories.