**🧠 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

**1. Game Playing Agents**

* How to build agents that can play games against a human or another agent.
* Agents can be designed to **never lose** in Tic-Tac-Toe by making optimal moves.

**2. Search Trees**

* The game is represented as a **tree of possible game states**.
* Each node is a board configuration; edges represent moves.
* The tree explores **all possible outcomes** of the game.

**3. Minimax Algorithm**

* A classic decision-making algorithm used in **turn-based games**.
* The algorithm assumes both players play optimally:
  + One player tries to **maximize** their chances of winning.
  + The other tries to **minimize** the opponent's chances.
* It chooses the move that leads to the best worst-case outcome.

**4. Utility-Based Decision Making**

* Each game state is assigned a **utility value** (e.g., win = +1, draw = 0, loss = -1).
* The AI evaluates moves based on these utilities to make the best decision.
* Encourages rational decision-making under **uncertainty** and **adversarial conditions**.

**🧩 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

**✅ 1. Initial State**

* A **completely empty 3×3 grid**.
* No moves have been made.
* Player X usually goes first.

**🎯 2. Actions**

* A player places their mark (X or O) in **any empty cell**.
* Each action leads to a new board state.

**🧪 3. Goal Test**

Checks whether the current state meets **any of these end conditions**:

* ✅ Player X or O has **3 in a row** (horizontal, vertical, or diagonal) → **Win**
* ⚠️ Board is **full with no winner** → **Draw**

**📊 4. Path Cost**

* Typically not critical in Tic-Tac-Toe.
* But for formality:
  + Each move can have a **uniform cost** (e.g., 1 per move).
  + Total path cost = **number of moves taken**.
* **🧠 5. Problem Type**

| **Property** | **Description** |
| --- | --- |
| 🎯 Goal-based | Agent aims to reach a **winning state** |
| 🔄 Deterministic | Outcome of each action is **predictable** |
| 👀 Fully observable | The **entire game state is visible** |
| 👤 Adversarial | Involves **two players with opposing goals** |
|  |  |

**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?

**1. Is this a goal-based agent problem?**

**✔️ Yes.**  
Tic-Tac-Toe is a **goal-based agent problem** because the agent (player) selects actions to achieve a specific goal — **winning the game** by forming a line of three marks. The agent's decisions are guided by the outcome it wants to reach.

**2. Is Tic-Tac-Toe a deterministic game? Why?**

**✔️ Yes.**  
Tic-Tac-Toe is a **deterministic game** because:

* The **outcome of every move is predictable**.
* There is **no randomness** or uncertainty in how the board changes after a move.
* Given the current state and an action, the **next state is fully determined**.

**3. Is it a fully observable environment?**

**✔️ Yes.**  
Tic-Tac-Toe is a **fully observable environment** because:

* Both players can **see the entire board** at all times.
* There is **no hidden information**.
* Every move and game state is **visible to both agents**.

**4. Is it a single-agent or multi-agent problem?**

**✔️ Multi-agent.**  
Tic-Tac-Toe is a **multi-agent problem** because:

* It involves **two players (agents)** competing against each other.
* Each agent must consider the actions of the **opponent**, making it an **adversarial setting**.