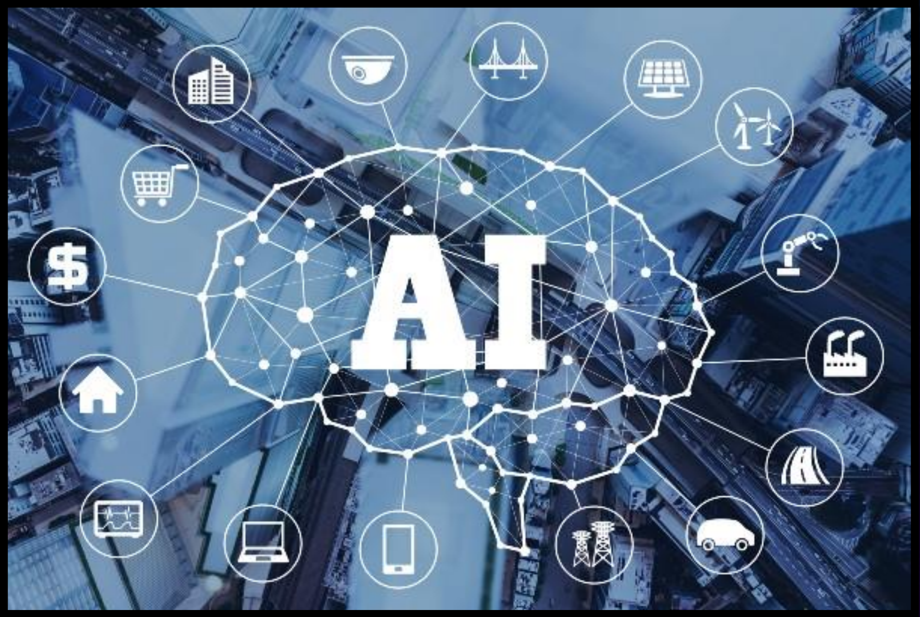
WINNING STRATEGY OF TIC TAC TOE

CONTINUOUS ASSESSMENT 1

Name : Saikat Saha
Univ Roll : 12000121053

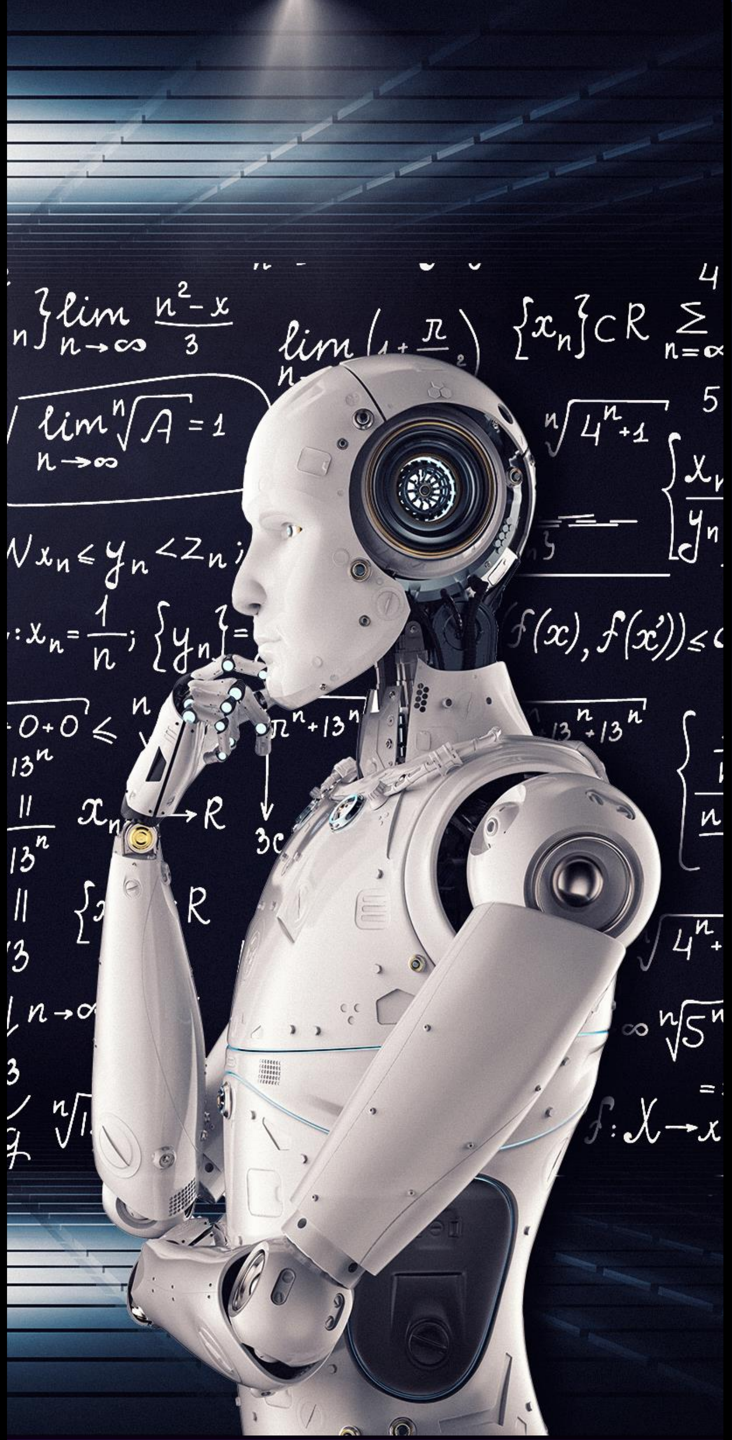
Introduction

- AI's remarkable growth in the gaming industry has revolutionized strategic gameplay.
- Tic Tac Toe, a classic paper-and-pencil game, serves as an ideal platform to explore advanced AI strategies.
- In this presentation, we'll delve into powerful AI techniques that guarantee optimal outcomes in Tic Tac Toe.



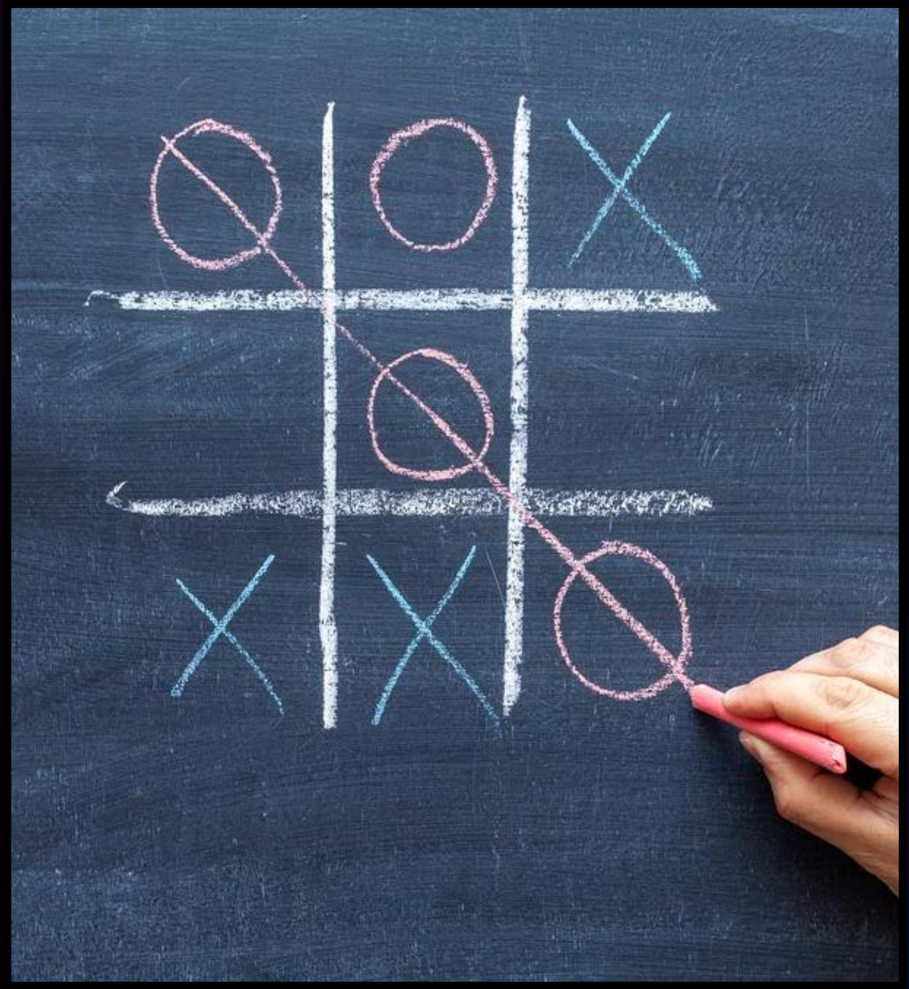
Objective

- Explore advanced AI strategies for playing Tic Tac Toe.
- Showcase how combining strategies leads to unbeatable AI performance.
- Discuss the ethical considerations in AI development.
- Inspire further exploration of AI in gaming and real-world applications.



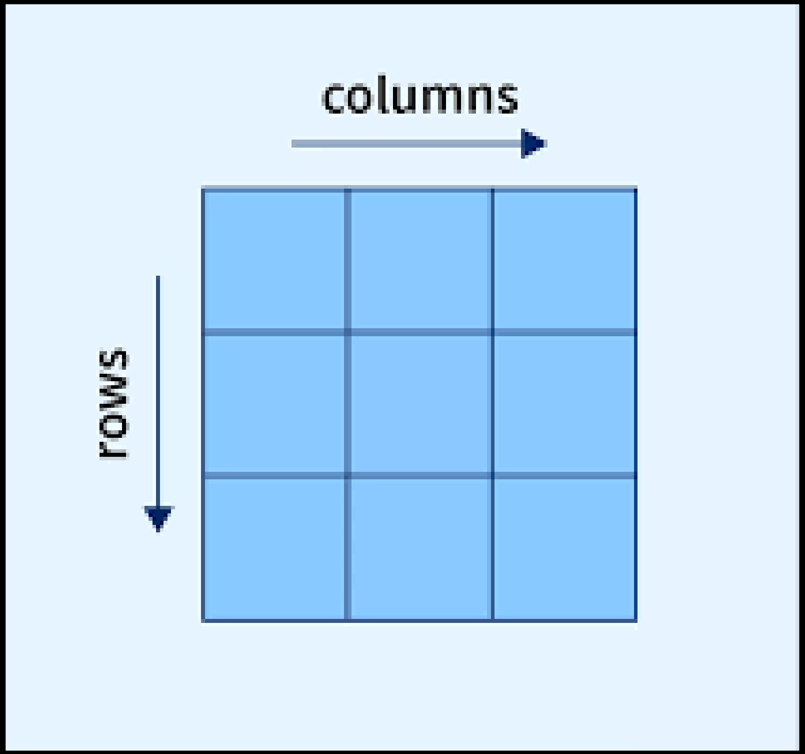
Basis rules of Tic tac toe

- Tic Tac Toe is a two-player game played on a 3x3 grid.
- Players take turns placing their markers (X or O) in empty cells.
- The objective is to create a line of three markers horizontally, vertically, or diagonally.
- If the board fills up without a winner, the game ends in a draw.
- The first player to achieve a line of three markers wins the game.
- The game is simple, yet strategic, making it an excellent platform for AI development.



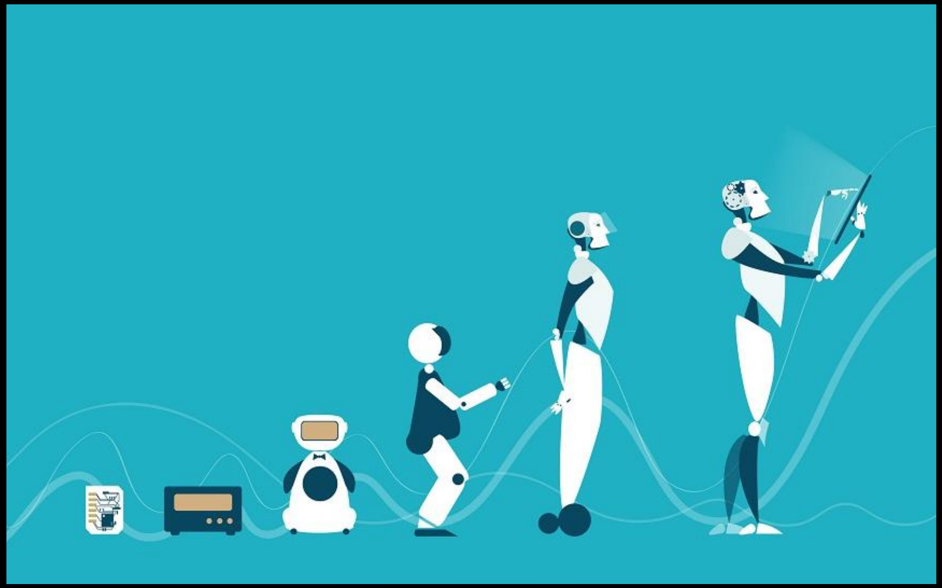
Game Representation

- Representing the Tic Tac Toe board is crucial for AI decision-making.
- Various data structures can be used, such as a 2D array or a list of lists.
- Each cell in the structure corresponds to a position on the board.
- Empty cells are denoted by 2, X by 3, and O by 5.
- Keeping track of the current game state enables the AI to make informed moves.
- Choosing the appropriate data structure is essential for efficient AI implementation.



The AI Evolution - From Beginner to Advanced

- Beginner AI: Random Moves
  - Simple strategy, making random moves on the board.
  - Lacks intelligence and predictability.
- Intermediate AI: Minimax Algorithm
  - An optimal strategy ensuring unbeatable play.
  - Explores all possible moves in the game tree.
  - Guarantees a draw or win if the opponent makes mistakes.
- Advanced AI: Additional Techniques
  - Alpha-Beta Pruning: Optimizes Minimax by cutting off irrelevant branches in the game tree.
  - Heuristic Evaluation Functions: Approximates game states' desirability for faster evaluation.



Methods used

- **Make2:** returns 5 if centre square of the board is blank i,e if Board[5] = 2

else returns any blank non-corner square i.e. 2/4/6/8.

- **Posswin(P):** Returns 0 if player can not win on the next move.

else returns the number of the square that constitute the winning move.

- **Rules in Posswin( )**

if Posswin(P) = 0 {P cannot win } then find whether opponent can win.
If so block it.

- **Strategy used by Posswin ()**

Posswin() checks one at a time for each row, column and diagonal as follows:
if 3*3*2 =18 then player X can win.
if 5*5*2 = 50 then player O can win.

- **Go(n) :** Move to Board(n) setting Board[n] to 3 if Turn is odd.

else 5 if Turn is even.

Assumptions

- The first player always uses symbol X.
- There are in all 8 moves in the worst case.
- Computer is represented by C and Human in represented by H.
- Convention used in algorithm on next slide –
  - If C plays first (Computer plays X, Human plays O) – Odd moves
  - If H plays first (Human plays X, Computer plays O) – Even moves
  - For the sake of clarity, we use C and H

Algorithm : Human plays first

- Move 1: H plays
- Move 2: If B[5] is blank, then Go(5) else Go(1)
- Move 3: H plays
- Move 4: (By now H has played 2 chances)
  - If PossWin(H) then (block H] Go (PossWin(H))
  - else Go (Make_2)
- Move 5: H plays
- Move 6: (By now both have played 2 chances)
  - If PossWin(C) then (won) Go(PossWin(C))
  - else (block H) if PossWin(H) then Go(PossWin(H)) elseGo(Make 2)
- Moves 7 & 9 : H plays
- Move 8: {By now computer has played 3 chances}
  - If PossWin(C) then (won) Go(PossWin(C))
  - else (block H) if PossWin(H) then Go(PossWin(H)) elseGo(Anywhere)

Algorithm : AI plays first

- Move 1: Go (5)
- Move 2: H plays
- Move 3: If B[9] is blank, then Go(9) else Go(3) (make 2)
- Move 4: H plays
- Move 5: (By now computer has played 2 chances)
  - If PossWin(C) then (won) Go(PossWin(C))
  - else (block H) if PossWin (H) then Go(PossWin(H)) else if B/71 isblank then Go(7) else Go(3)
- Move 6: H plays
- Moves 7 & 9 :
  - If PossWin(C) then (won/ Go(PossWin(C))
  - else (block H) if PossWin(H) then Go(PossWin(H)) elseGo(Anywhere)
  - Move 8: H plays

Conlusion

- Combining strategies in hybrid AI enhances adaptability and performance, creating powerful opponents.
- Ethical considerations in AI development are crucial to ensure fair play and transparency.
- The world of AI in gaming offers endless possibilities for innovation and real-world applications.