Department of Computer Engineering

Academic Term II : 22-23

Class: T.E (Comp A), SemVI **Subject Name: Artificial Intelligence**

Student Name: Vailantan Fernandes **Roll No: 9197**

Practical No:	**1**
Title:	**Tic-Tac-Toe**
Date of Performance:	
Date of Submission:	

Rubrics for Evaluation:

Sr. No	Performance Indicator	Excellent	Good	Below Average	Marks
1	On time Completion & Submission (01)	01 (On Time)	NA	00 (Not on Time)	
2	Logic/Algorithm Complexity analysis(03)	03(Correct)	02(Partial)	01 (Tried)	
3	Coding Standards (03): Comments/indention/Naming conventions Test Cases /Output	03(All used)	02 (Partial)	01 (rarely followed)	
4	Post Lab Assignment (03)	03(done well)	2 (Partially Correct)	1(submitted)	
Total					

Signature of the Teacher :

Experiment No: 1

Title: Tic Tac Toe game implementation using Magic Square (Heuristic Method)

Objective: To write a computer program in such a way that computer wins most of the time

Theory: It is two players, X and O, game who take turns marking the spaces in a 3×3 grid. The player who succeeds in placing three respective marks in a horizontal, vertical, or diagonal row wins the game.

1	2	3
4	5	6
7	8	9

Algorithm:

Approach 1: Brute Force

a) Consider a Board having nine elements vector.

b) Each element will contain

i) 0 for blank ii) 1 indicating X

player move iii) 2 indicating O

player move

c) Computer may play as X or O player.

d) First player who so ever is always plays X.

2) MT is a vector of 3^9 elements, each element of which is a nine element vector representing board position.

3) MT is a vector of 3^9 elements, each element of which is a nine element vector representing board position.
   a) Move Table (MT) is a vector of 39 elements, each element of which is a nine element vector representing board position.

Index	Current Board position	New Board position
0	000000000	000010000
1	000000001	020000001
2	000000002	000100002
3	000000010	002000010
⋮		

   b) To make a move, do the following:

      a. View the vector (board) as a ternary number and convert it to its corresponding decimal number.

      b. Use the computed number as an index into the MT and access the vector stored there.

         i. The selected vector represents the way the board will look after the move.

      c. Set board equal to that vector.

Approach 2: MINMAX

1) Board: A nine-element vector representing the board: B[1..9]

2) Following conventions are used

   (a) 2 - indicates blank
   (b) 3 - X
   (c) 5 - 0

Turn: An integer

(d) 1 - First move
(e) 9 - Last move

Make_2 →tries to make valid 2

Make_2 first tries to play in the center if free and returns 5 (square number). If not possible, then it tries the various suitable non corners square and returns square number.

Go(n) ← makes a move in square 'n' which is blank represented by 2.

PossWin (P) → Returns 0, if player P cannot win in its next move,otherwise the number of square that constitutes a winning move for P.

Rule: If PossWin (P) = 0 {P cannot win} then find whether opponent can win. If so, then block it.

2) Move 1: go (5)
3) Move 2: If B[5] is blank, then Go(5) else Go(1)
4) Move 3: If B[9] is blank, then Go(9) else Go(3) *{make 2}*
5) Move 4: *{By now human (playing X) has played 2 chances}*If PossWin(X) then *{block H}* Go (PossWin(X)) else Go (Make_2)
6) Move 5: *{By now computer has played 2 chances}* If PossWin(X) then *{won}* Go(PossWin(X)) else *{block H}* if PossWin(O) then Go(PossWin(O)) else if B[7] is blank then Go(7) else Go(3)
7) Move 6: *{By now both have played 2 chances}* If PossWin(O) then *{won}* Go(PossWin(O)) else *{block H}* if PossWin(X) then Go(PossWin(X)) else Go(Make_2)
8) Moves 7 & 9 : *{By now human (playing O) has played 3 chances}* If PossWin(X) then *{won}* Go(PossWin(X)) else *{block H}* if PossWin(O) then Go(PossWin(O)) else Go(Anywhere)
9) Move 8: *{By now computer has played 3 chances}* If PossWin(O) then *{won}* Go(PossWin(O)) else *{block H}* if PossWin(X) then Go(PossWin(X)) else Go(Anywhere)

Approach 3: Heuristic Function

1) Same as approach 2 except for one change in the representation of the board.

   a) Board is considered to be a magic square of size 3 X 3 with 9 blocks numbered by numbers indicated by magic square.

2) This representation makes process of checking for a possible win simpler.

3) Board Layout as magic square. Each row, column and diagonals add to 15.

8	3	4	**15**
1	5	9	**15**
6	7	2	**15**

4) Maintain the list of each player's blocks in which he has played.

   a) Consider each pair of blocks that player owns.

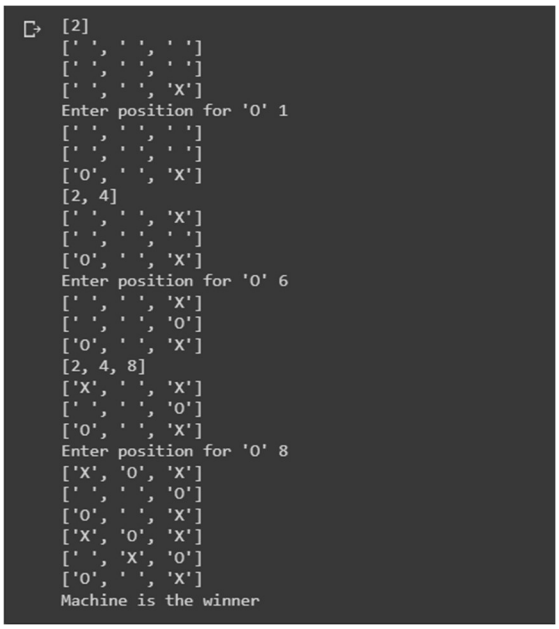
   b) Compute difference D between 15 and the sum of the two blocks.

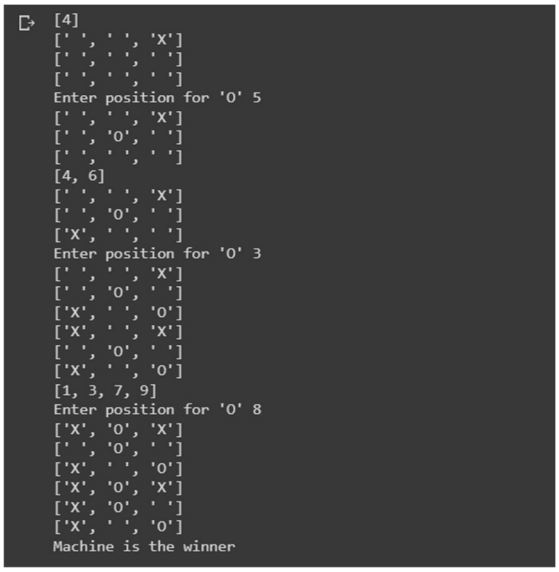
   c) If D < 0 or D > 9 then

      i) these two blocks are not collinear and so can be ignored

      ii) Otherwise if the block representing difference is blank (i.e., not in either list) then a move in that block will produce a win.

Output:





Post Lab Assignment:

1. What is the easiest trick to win Tic Tac Toe?
2. What is the algorithm to follow to win a 5*5 Tic Tac Toe?
3. Is there a way to never lose at Tic-Tac-Toe?
4. What can tic-tac-toe help you with?