

2CEIT602: Artificial Intelligence

Practical-4

Aim: Write a python program to create tic-tac-toe game using the minimax algorithm.

- Create class StateNode with its proper members (getScoreValue(), isValidMove(), isGameOver(), drawboard(), getEmptyCells(), setMove(), isWin(), etc.)
- Create class ticTacToe with its proper members (minimax(), computerMove(), playerMove(), start_game (), etc.)
- Output should be according shown in Output image
- Use only math, random, matplotlib libraries in python.

Output:

Choose X or O
Chosen: O

First to start?[y/n]: y

Player Move [O]

Use numpad (1..9): 1

Computer Move [X]

O		

Player Move [O]

O		
	X	

Use numpad (1..9): 2

Computer Move [X]

O	O	X
	X	

Player Move [O]

O	O	X
	X	

Use numpad (1..9): 3
Bad move
Use numpad (1..9): 4

Computer Move [X]

O	O	X
O	X	

Player Move [O]

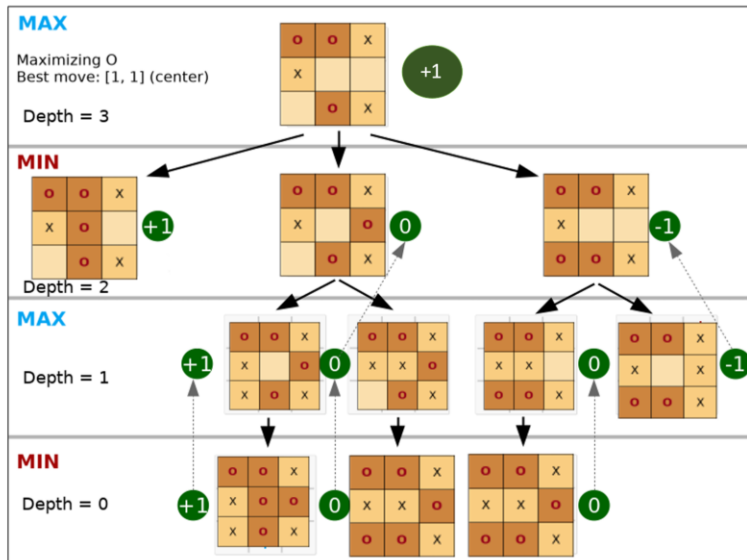
O	O	X
O	X	

Use numpad (1..9): 5
Bad move
Use numpad (1..9): 6

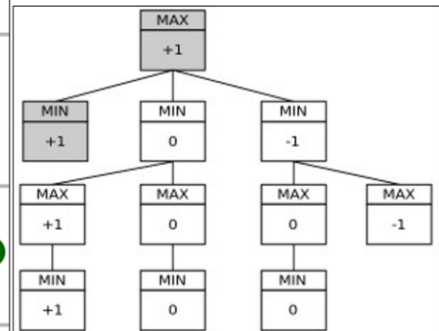
Computer Move [X]

O	O	X
O	X	
X		

YOU LOSE!



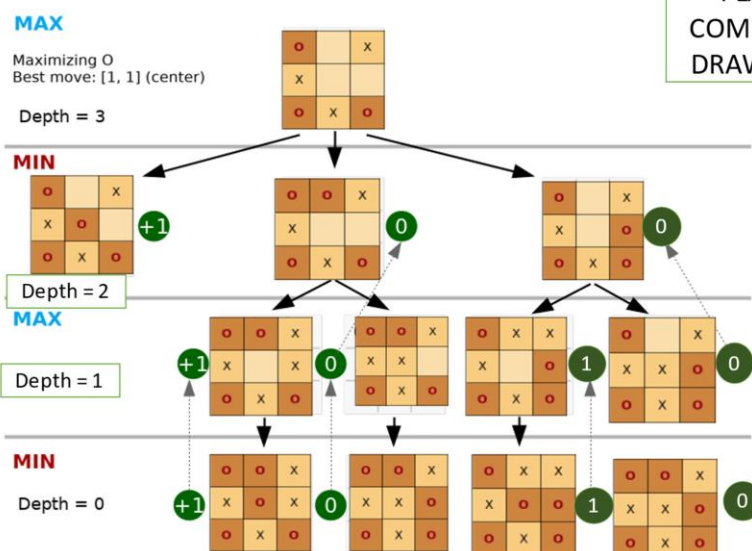
PLAYER = X = -1
COMPUTER = O = +1
DRAW (NO WIN) = 0



```

1 minimax(state, depth, player)
2   if (player = max) then
3     best = [null, -infinity]
4   else
5     best = [null, +infinity]
6   if (depth = 0 or gameover) then
7     score = evaluate this state for player
8     return [null, score]
9   for each valid move m for player in state s do
10    execute move m on s
11    [move, score] = minimax(s, depth - 1, -player)
12    undo move m on s
13    if (player = max) then
14      if score > best.score then best = [move, score]
15    else
16      if score < best.score then best = [move, score]
17  return best
18 end

```



PLAYER = X = -1
COMPUTER = O = +1
DRAW (NO WIN) = 0