

Tic Tac Toe (Computer Vs Computer)



```
def main():
    print("TIC TAC TOE")
    print("AI vs AI")
```

```
while True:
    theBoard = [' ']*10
    drawBoard(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
    split()
```

```
isGamePlaying = True
```

```
first = True
```

```
while isGamePlaying:
```

```
    for agent in ['x', 'o']:
```

```
        if first:
```

```
            move = randomNum-  
ber()
```

```
            theBoard[move] = agent
```

```
            first = False
```

```
    else:
```

```
        move = getAgentMove  
(theBoard, agent)
```

#1 check if it is a Winner
if Winner (theBoard, agent):
drawBoard(theBoard)

P.SAI DEEKSHIT

IBM18CS148

print(agent + " won the game")
isGamePlaying = False

break
#1 check if it is a Draw
if Draw (theBoard):
drawBoard(theBoard)

print("It is a tie")
isGamePlaying = False

break
drawBoard(theBoard)

step(1)

print("Do you want them play again: (Y/N)")

choice = input()

if choice == 'n' or choice == 'N':
break.

(2)

def getAgentMove(theBoard, letter):
 if letter == 'O':
 for move in range(1, 10):
 if isSpaceFree(theBoard, move) == 0:
 Time and CanWin(theBoard, 'O', move):

P.SAI DEEKSHITH

IBN18CS168

of Reg.

return move
 for move in range(1, 10):
 if isSpaceFree(theBoard, move) == 0:
 Time and CanWin(theBoard, 'X',
 move):

else
 for move in range(1, 10):
 if isSpaceFree(theBoard, move) == 0:
 Time and CanWin(theBoard, 'X',
 move):

return move
 for move in range(1, 10):
 if isSpaceFree(theBoard, move) == 0:
 Time and CanWin(theBoard,
 'O', move):

return WinStrategy(theBoard, letter)
 (3)



```
def WinStrategy(board, letter):  
    if board[5] == ' ':  
        return 5  
    for i in [1, 3, 7, 9]:  
        if board[i] == ' ':  
            return i  
    for i in [2, 4, 6, 8]:  
        if board[i] == ' ':  
            return i.
```

```
def canWin(board, letter, move):  
    temp = copy.deepcopy(board)  
    temp[move] = letter  
    return isWinner(temp, letter)
```

```
def isWinner(board, letter):  
    return ((board[1] == letter and board[3] == letter  
    and board[5] == letter) or  
    (board[4] == letter and board[6] == letter  
    and board[8] == letter) or  
    (board[7] == letter and board[9] == letter  
    and board[2] == letter))
```

(4)

(board[1] == letter and board[4] == letter and
board[7] == letter) or

(board[2] == letter and board[5] == letter
and board[8] == letter) or

(board[3] == letter and board[6] == letter
and board[9] == letter)).

5