Artificial Intelligence Lab Test - 1

Name: Anusree Manoj K
USN: 1BM18CS017
Section: 5A
Date : 10-11-2020

Implement Jic-tac-toe using a 2-agent algorithm
(computer vs computer)

```
board = [' '] *9

def check_win (p agent_mark, board):
    win = [f '{agent_mark}'] *3
    return board [:3] == win or board [3:6] == win or board [6:9]
        == win or [board [0], board [4], board [8]] == win or
        [board [2], board [4], board [6]] == win or
        [board [0], board [3], board [6]] == win or
        [board [1], board [4], board [7]] == win or
        [board [2], board [5], board [8]] == win

def check-draw (board):
    return ' ' not in board

def board-copy (board):
    new_board = []
    for c in board:
        new_board += c
    return new_board
```

Anusree

Anusree Manoj K
IBHU8CS017
10-11-2020

```
def test_win_move (move, agent_mark, board):
    copy = board_copy (board)
    copy [move] = # agent_mark
    return check_win (agent_mark, copy)


def win_strategy (board):
    if board [4] == ' '
        return 4
    for i in [0, 2, 6, 8]:
        if board [i] == ' ':
            return i
    for i in [1, 3, 5, 7]:
        if board[i] == ' ':
            return i


def get_agent_move (board):
    for i in range (9):
        if board [i] == ' ' and test_win_move (i, 'x', board):
            return i
    for i in rang (9):
        if board [i] == ' ' and test_win_move (i, 'o', board):
            return i

    return win_strategy (board)
```

Anuree Manoj K
1 BM CSC S017
10-11-2020

```
def tic_tac_toe ( ):
    playing = True
    while playing :
        in_game = True
        board = [ ' ' ] * 9
        agent 1 = 'X'
        agent 2 = 'O'
        while in_game:
            for agent in ['X', 'O']
                move = get_agent_move (board)
                board [move] = agent &
                if check_win( agent &, board ):
                    in_game = False
                    display_board (board)
                    print (f "{agent} won")
                    break
                if check_draw (board):
                    in_game = False
                    display_board (board)
                    print ( " Its draw")
                    break
                display_board (board)
                sleep (2)
                move = get_agent_move (board)
                board [move] = agent 2
        print ( 'Continue? )
        ans = input ( )
        if ans not in 'yY':
            playing = False
```

③

Anuree