**NAME :** DUDHE ANIL AJINATH

**ROLL NO :** TEAD21269

**SUBJECT :** AI

**CLASS :** TE

**BRANCH :** AI&DS

**EXPERIMENT NO :**

**TITLE :**

**Implement Alpha-Beta Tree search for any game search problem**.

**CODE**

def print\_board(board):

print("Current State Of Board:\n")

for i in range(0, 9):

if (i > 0) and (i % 3) == 0:

print("\n")

if board[i] == 0:

print("- ", end=" ")

if board[i] == 1:

print("O ", end=" ")

if board[i] == -1:

print("X ", end=" ")

print("\n\n")

def user1\_turn(board):

pos = input("Enter X's position from [1...9]: ")

pos = int(pos)

if board[pos - 1] != 0:

print("Wrong Move!!!")

exit(0)

board[pos - 1] = -1

def user2\_turn(board):

pos = input("Enter O's position from [1...9]: ")

pos = int(pos)

if board[pos - 1] != 0:

print("Wrong Move!!!")

exit(0)

board[pos - 1] = 1

def minimax(board, player):

x = analyze\_board(board)

if x != 0:

return x \* player

pos = -1

value = -2

for i in range(0, 9):

if board[i] == 0:

board[i] = player

score = -minimax(board, player \* -1)

if score > value:

value = score

pos = i

board[i] = 0

if pos == -1:

return 0

return value

def comp\_turn(board):

pos = -1

value = -2

for i in range(0, 9):

if board[i] == 0:

board[i] = 1

score = -minimax(board, -1)

board[i] = 0

if score > value:

value = score

pos = i

board[pos] = 1

def analyze\_board(board):

win\_combinations = [

[0, 1, 2], [3, 4, 5], [6, 7, 8],

[0, 3, 6], [1, 4, 7], [2, 5, 8],

[0, 4, 8], [2, 4, 6]

]

for combo in win\_combinations:

if board[combo[0]] != 0 and board[combo[0]] == board[combo[1]] == board[combo[2]]:

return board[combo[0]]

return 0

def main():

choice = input("Enter 1 for single player, 2 for multiplayer: ")

choice = int(choice)

board = [0, 0, 0, 0, 0, 0, 0, 0, 0]

if choice == 1:

print("Computer : O Vs. You : X")

player = input("Enter to play 1(st) or 2(nd): ")

player = int(player)

for i in range(0, 9):

if analyze\_board(board) != 0:

break

if (i + player) % 2 == 0:

comp\_turn(board)

else:

print\_board(board)

user1\_turn(board)

else:

for i in range(0, 9):

if analyze\_board(board) != 0:

break

if i % 2 == 0:

print\_board(board)

user1\_turn(board)

else:

print\_board(board)

user2\_turn(board)

x = analyze\_board(board)

if x == 0:

print\_board(board)

print("Draw!!!")

elif x == -1:

print\_board(board)

print("X Wins!!! O Loses!!!")

elif x == 1:

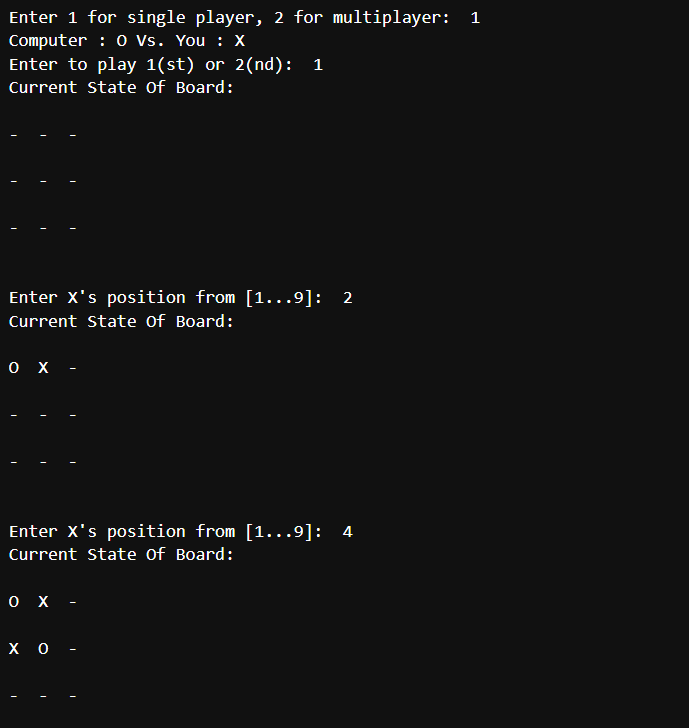
print\_board(board)

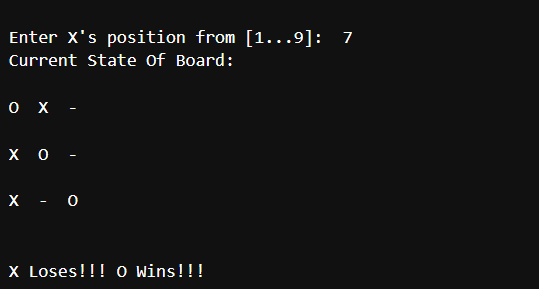
print("X Loses!!! O Wins!!!")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**OUTPUT:**

****

****