EXPERIMENT 13

Write a python program to implement minimax algorithm for gaming

CODE:

import math

def print\_board(b):

for r in b: print(" | ".join(r)); print("-"\*5)

def winner(b):

for i in range(3):

if b[i][0]==b[i][1]==b[i][2]!=" ": return b[i][0]

if b[0][i]==b[1][i]==b[2][i]!=" ": return b[0][i]

if b[0][0]==b[1][1]==b[2][2]!=" ": return b[0][0]

if b[0][2]==b[1][1]==b[2][0]!=" ": return b[0][2]

return None

def full(b): return all(c!=" " for r in b for c in r)

def minimax(b, turn):

w = winner(b)

if w=="O": return 1

if w=="X": return -1

if full(b): return 0

best = -math.inf if turn=="O" else math.inf

for i in range(3):

for j in range(3):

if b[i][j]==" ":

b[i][j]=turn

score=minimax(b,"X" if turn=="O" else "O")

b[i][j]=" "

best=max(best,score) if turn=="O" else min(best,score)

return best

def best\_move(b):

move=None; best=-math.inf

for i in range(3):

for j in range(3):

if b[i][j]==" ":

b[i][j]="O"

score=minimax(b,"X")

b[i][j]=" "

if score>best: best,move=score,(i,j)

return move

def play():

b=[[" "]\*3 for \_ in range(3)]

print("You=X, Computer=O")

while True:

print\_board(b)

r,c=map(int,input("Your move row col: ").split())

if b[r][c]!=" ": continue

b[r][c]="X"

if winner(b)=="X": print\_board(b); print("You win!"); break

if full(b): print\_board(b); print("Draw!"); break

i,j=best\_move(b); b[i][j]="O"

if winner(b)=="O": print\_board(b); print("Computer wins!"); break

if full(b): print\_board(b); print("Draw!"); break

play()

OUTPUT:

