**EXPERIMENT 15**

board = [' ' for \_ in range(9)]

def print\_board():

for i in range(3):

print(board[3\*i] + '|' + board[3\*i+1] + '|' + board[3\*i+2])

if i < 2:

print('-+-+-')

def check\_winner(b, player):

win\_conditions = [

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

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

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

]

return any(all(b[i] == player for i in cond) for cond in win\_conditions)

def is\_draw(b):

return all(cell != ' ' for cell in b)

def alpha\_beta(b, depth, alpha, beta, is\_maximizing):

if check\_winner(b, 'O'):

return 1

elif check\_winner(b, 'X'):

return -1

elif is\_draw(b):

return 0

if is\_maximizing:

max\_eval = -float('inf')

for i in range(9):

if b[i] == ' ':

b[i] = 'O'

eval = alpha\_beta(b, depth + 1, alpha, beta, False)

b[i] = ' '

max\_eval = max(max\_eval, eval)

alpha = max(alpha, eval)

if beta <= alpha:

break

return max\_eval

else:

min\_eval = float('inf')

for i in range(9):

if b[i] == ' ':

b[i] = 'X'

eval = alpha\_beta(b, depth + 1, alpha, beta, True)

b[i] = ' '

min\_eval = min(min\_eval, eval)

beta = min(beta, eval)

if beta <= alpha:

break

return min\_eval

def best\_move():

best\_score = -float('inf')

move = -1

for i in range(9):

if board[i] == ' ':

board[i] = 'O'

score = alpha\_beta(board, 0, -float('inf'), float('inf'), False)

board[i] = ' '

if score > best\_score:

best\_score = score

move = i

return move

def play\_game():

while True:

print\_board()

move = int(input("Enter your move (0-8): "))

if board[move] != ' ':

print("Invalid move. Try again.")

continue

board[move] = 'X'

if check\_winner(board, 'X'):

print\_board()

print("You win!")

break

elif is\_draw(board):

print\_board()

print("It's a draw!")

break

ai\_move = best\_move()

board[ai\_move] = 'O'

print(f"AI plays at position {ai\_move}")

if check\_winner(board, 'O'):

print\_board()

print("AI wins!")

break

elif is\_draw(board):

print\_board()

print("It's a draw!")

break

play\_game()

