import math

# Board setup

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

# Display board

def print\_board():

print()

for row in [board[i\*3:(i+1)\*3] for i in range(3)]:

print('| ' + ' | '.join(row) + ' |')

print()

# Check for winner

def check\_winner(board, player):

win\_conditions = [

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

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

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

]

for condition in win\_conditions:

if all(board[i] == player for i in condition):

return True

return False

def is\_draw(board):

return ' ' not in board

# Get available moves

def available\_moves(board):

return [i for i, spot in enumerate(board) if spot == ' ']

# Minimax Algorithm

def minimax(board, depth, is\_maximizing):

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

return 1

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

return -1

elif is\_draw(board):

return 0

if is\_maximizing:

best\_score = -math.inf

for move in available\_moves(board):

board[move] = 'O'

score = minimax(board, depth + 1, False)

board[move] = ' '

best\_score = max(score, best\_score)

return best\_score

else:

best\_score = math.inf

for move in available\_moves(board):

board[move] = 'X'

score = minimax(board, depth + 1, True)

board[move] = ' '

best\_score = min(score, best\_score)

return best\_score

# AI makes move

def ai\_move():

best\_score = -math.inf

move = None

for i in available\_moves(board):

board[i] = 'O'

score = minimax(board, 0, False)

board[i] = ' '

if score > best\_score:

best\_score = score

move = i

board[move] = 'O'

# Play game

def play\_game():

print("Welcome to Tic-Tac-Toe! You are 'X' and AI is 'O'")

print\_board()

while True:

# Human turn

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

if board[move] != ' ':

print("Invalid move. Try again.")

continue

board[move] = 'X'

print\_board()

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

print("You win!")

break

elif is\_draw(board):

print("It's a draw!")

break

# AI turn

ai\_move()

print("AI's move:")

print\_board()

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

print("AI wins!")

break

elif is\_draw(board):

print("It's a draw!")

break

# Run the game

play\_game()