import math

def print\_board(board):

for i in range(3):

print(' | '.join(board[i\*3:(i+1)\*3]))

if i < 2:

print('---------')

def check\_winner(board, player):

wins = [

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

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

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

]

return any(all(board[pos] == player for pos in line) for line in wins)

def is\_board\_full(board):

return all(space != ' ' for space in board)

def utility(board, ai, human):

if check\_winner(board, ai):

return 10

elif check\_winner(board, human):

return -10

else:

return 0

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

score = utility(board, ai, human)

if score == 10 or score == -10 or is\_board\_full(board):

return score

if is\_maximizing:

best\_score = -math.inf

for i in range(9):

if board[i] == ' ':

board[i] = ai

val = minimax(board, depth + 1, False, ai, human)

board[i] = ' '

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

return best\_score

else:

best\_score = math.inf

for i in range(9):

if board[i] == ' ':

board[i] = human

val = minimax(board, depth + 1, True, ai, human)

board[i] = ' '

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

return best\_score

def best\_move(board, ai, human):

best\_val = -math.inf

move = None

for i in range(9):

if board[i] == ' ':

board[i] = ai

move\_val = minimax(board, 0, False, ai, human)

board[i] = ' '

if move\_val > best\_val:

best\_val = move\_val

move = i

return move

def main():

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

human = 'X'

ai = 'O'

current\_player = human

print("Welcome to Tic Tac Toe!")

print("Positions on the board are numbered 0-8 as follows:")

print\_board([str(i) for i in range(9)])

while True:

print("\nCurrent board:")

print\_board(board)

if current\_player == human:

try:

move = int(input("Your move (0-8): "))

if move < 0 or move > 8 or board[move] != ' ':

print("Invalid move. Please try again.")

continue

except ValueError:

print("Please enter a valid integer from 0 to 8.")

continue

board[move] = human

else:

move = best\_move(board, ai, human)

print(f"AI chooses position {move}")

board[move] = ai

# Check for win or tie

if check\_winner(board, current\_player):

print\_board(board)

if current\_player == human:

print(" Congratulations! You win!")

else:

print(" AI wins! Better luck next time.")

break

if is\_board\_full(board):

print\_board(board)

print(" It's a tie!")

break

current\_player = ai if current\_player == human else human

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

main()