import math

class TicTacToe:

def \_\_init\_\_(self):

self.board = [[' ' for \_ in range(3)] for \_ in range(3)]

def display\_board(self):

for row in self.board:

print("|".join(row))

print("-----")

def is\_winner(self, player):

for i in range(3):

if all(self.board[i][j] == player for j in range(3)) or all(self.board[j][i] == player for j in range(3)):

return True

if all(self.board[i][i] == player for i in range(3)) or all(self.board[i][2 - i] == player for i in range(3)):

return True

return False

def is\_full(self):

return all(self.board[i][j] != ' ' for i in range(3) for j in range(3))

def get\_empty\_cells(self):

return [(i, j) for i in range(3) for j in range(3) if self.board[i][j] == ' ']

def make\_move(self, player, row, col):

if self.board[row][col] == ' ':

self.board[row][col] = player

return True

return False

def minimax(game, depth, maximizing\_player):

if game.is\_winner('X'):

return 1

elif game.is\_winner('O'):

return -1

elif game.is\_full():

return 0

if maximizing\_player:

max\_eval = -math.inf

for i, j in game.get\_empty\_cells():

game.make\_move('X', i, j)

eval = minimax(game, depth + 1, False)

game.board[i][j] = ' ' # undo move

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

return max\_eval

else:

min\_eval = math.inf

for i, j in game.get\_empty\_cells():

game.make\_move('O', i, j)

eval = minimax(game, depth + 1, True)

game.board[i][j] = ' ' # undo move

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

return min\_eval

def best\_move(game):

best\_score = -math.inf

best\_move = None

for i, j in game.get\_empty\_cells():

game.make\_move('X', i, j)

score = minimax(game, 0, False)

game.board[i][j] = ' ' # undo move

if score > best\_score:

best\_score = score

best\_move = (i, j)

return best\_move

def print\_instructions():

print("Tic-Tac-Toe Game")

print("Enter your move in the format 'row,col'. For example, '0,0' for the top-left corner.")

print("You are 'O', and the computer is 'X'. Let's begin!\n")

def main():

print\_instructions()

game = TicTacToe()

while True:

game.display\_board()

player\_move = input("Your move: ")

row, col = map(int, player\_move.split(","))

if game.make\_move('O', row, col):

if game.is\_winner('O'):

game.display\_board()

print("Congratulations! You win!")

break

elif game.is\_full():

game.display\_board()

print("It's a draw!")

break

computer\_move = best\_move(game)

game.make\_move('X', \*computer\_move)

if game.is\_winner('X'):

game.display\_board()

print("Sorry, you lose. Better luck next time!")

break

elif game.is\_full():

game.display\_board()

print("It's a draw!")

break

else:

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

print("Game over.")

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

main()