import math

# Create the board

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

# Print the board

def print\_board():

    for i in range(3):

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

# Check for winner

def is\_winner(brd, player):

    win\_combos = [

        [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(brd[i] == player for i in combo) for combo in win\_combos)

# Check if the board is full

def is\_full(brd):

    return ' ' not in brd

# Get list of empty spots

def get\_moves(brd):

    return [i for i, val in enumerate(brd) if val == ' ']

# Minimax algorithm

def minimax(brd, is\_max):

    if is\_winner(brd, 'O'):

        return 1

    if is\_winner(brd, 'X'):

        return -1

    if is\_full(brd):

        return 0

    if is\_max:

        best = -math.inf

        for move in get\_moves(brd):

            brd[move] = 'O'

            score = minimax(brd, False)

            brd[move] = ' '

            best = max(best, score)

        return best

    else:

        best = math.inf

        for move in get\_moves(brd):

            brd[move] = 'X'

            score = minimax(brd, True)

            brd[move] = ' '

            best = min(best, score)

        return best

# AI chooses the best move

def ai\_turn():

    best\_score = -math.inf

    best\_move = None

    for move in get\_moves(board):

        board[move] = 'O'

        score = minimax(board, False)

        board[move] = ' '

        if score > best\_score:

            best\_score = score

            best\_move = move

    board[best\_move] = 'O'

# Play the game

def play():

    print("Tic Tac Toe — You are X, AI is O")

    print\_board()

    while True:

        # Player move

        try:

            move = int(input("Your move (1-9): ")) - 1

            if board[move] != ' ':

                print("Cell already taken. Try again.")

                continue

        except:

            print("Invalid input. Enter 1-9.")

            continue

        board[move] = 'X'

        print\_board()

        if is\_winner(board, 'X'):

            print("You win!")

            break

        if is\_full(board):

            print("It's a tie!")

            break

        # AI move

        print("AI's move:")

        ai\_turn()

        print\_board()

        if is\_winner(board, 'O'):

            print("AI wins!")

            break

        if is\_full(board):

            print("It's a tie!")

            break

# Start the game

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

    play()

Tic Tac Toe — You are X, AI is O

| | | |

| | | |

| | | |

Your move (1-9): X

Invalid input. Enter 1-9.

Your move (1-9): 1

| X | | |

| | | |

| | | |

AI's move:

| X | | |

| | O | |

| | | |

Your move (1-9): 4

| X | | |

| X | O | |

| | | |

AI's move:

| X | | |

| X | O | |

| O | | |

Your move (1-9): 2

| X | X | |

| X | O | |

| O | | |

AI's move:

| X | X | O |

| X | O | |

| O | | |

AI wins!