TECHNOHACKS TASK 3

def print\_board(board):

for row in board:

print(" | ".join(row))

print("-" \* 5)

def check\_winner(board, player):

# Check rows, columns and diagonals for a win

for row in board:

if all([spot == player for spot in row]):

return True

for col in range(3):

if all([board[row][col] == player for row in range(3)]):

return True

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

return True

return False

def is\_board\_full(board):

return all([spot != ' ' for row in board for spot in row])

def tic\_tac\_toe():

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

current\_player = 'X'

while True:

print\_board(board)

print(f"Player {current\_player}, it's your turn.")

# Get user input

while True:

try:

row = int(input("Enter the row (0, 1, 2): "))

col = int(input("Enter the column (0, 1, 2): "))

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

board[row][col] = current\_player

break

else:

print("That spot is already taken. Try again.")

except (ValueError, IndexError):

print("Invalid input. Enter numbers between 0 and 2.")

# Check for a winner

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

print\_board(board)

print(f"Player {current\_player} wins!")

break

# Check for a draw

if is\_board\_full(board):

print\_board(board)

print("It's a draw!")

break

# Switch players

current\_player = 'O' if current\_player == 'X' else 'X'

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

tic\_tac\_toe()