import numpy as np

# Define two pairs of vectors

x1 = np.array([1, 1, 1, -1])

y1 = np.array([1, -1])

x2 = np.array([-1, -1, 1, 1])

y2 = np.array([-1, 1])

# Compute weight matrix W

W = np.outer(y1, x1) + np.outer(y2, x2)

# Define BAM function

def bam(x, layer=1):

# Forward association: Input -> Output

if layer == 1:

y = np.dot(W, x)

# Reverse association: Output -> Input

else:

y = np.dot(W.T, x)

y = np.where(y >= 0, 1, -1)

return y

# Test BAM with inputs (forward association)

x\_test = np.array([1, -1, -1, -1])

y\_test = bam(x\_test, layer=1) # Forward: Input -> Output

# Print output

print("Input x:", x\_test)

print("Output y:", y\_test)

# Test reverse association (output -> input)

y\_reverse\_test = bam(y\_test, layer=2) # Reverse: Output -> Input

print("Predicted Input (from Output):", y\_reverse\_test)