import numpy as np

class McCullochPittsNeuron:

def \_\_init\_\_(self, weights, threshold):

self.weights = np.array(weights)

self.threshold = threshold

def activate(self, inputs):

net\_input = np.dot(inputs, self.weights)

return 1 if net\_input >= self.threshold else 0

# Define weights and threshold for the ANDNOT neuron

weights = [1, -1] # Weight for A is 1, weight for B is -1

threshold = 1 # Threshold to fire the neuron

# Initialize the McCulloch-Pitts neuron

andnot\_neuron = McCullochPittsNeuron(weights, threshold)

# Test the neuron with all input combinations

inputs = np.array([

[0, 0],

[0, 1],

[1, 0],

[1, 1]

])

print("ANDNOT Function Results:")

print("A B | Output")

for input\_pair in inputs:

output = andnot\_neuron.activate(input\_pair)

print(f"{input\_pair[0]} {input\_pair[1]} | {output}")