**Lab Assignment No. 9**

Q4. Write a python program to design a Hopfield Network which stores 4 vectors

**Code:**

import numpy as np  
  
class HopfieldNetwork:  
 def \_\_init\_\_(self, num\_neurons):  
 self.num\_neurons = num\_neurons  
 self.weights = np.zeros((num\_neurons, num\_neurons))  
  
 def train(self, vectors):  
 num\_vectors = len(vectors)  
 for vector in vectors:  
 vector = np.reshape(vector, (self.num\_neurons, 1))  
 self.weights += np.dot(vector, vector.T) / num\_vectors  
 np.fill\_diagonal(self.weights, 0)  
  
 def recall(self, input\_vector, max\_iter=100):  
 output\_vector = np.copy(input\_vector)  
 for \_ in range(max\_iter):  
 prev\_output = np.copy(output\_vector)  
 output\_vector = np.sign(np.dot(self.weights, output\_vector))  
 if np.array\_equal(output\_vector, prev\_output):  
 break  
 return output\_vector

network = HopfieldNetwork(4)  
  
vectors = np.array([[1, 1, 1, 1],  
 [1, -1, 1, -1],  
 [-1, 1, -1, 1],  
 [-1, -1, -1, -1]])  
  
  
network.train(vectors)  
  
for vector in vectors:  
 output = network.recall(vector)  
 print("Input:", vector)  
 print("Output:", output)  
 print()

**Output:**

Input: [1 1 1 1]  
Output: [1. 1. 1. 1.]  
  
Input: [ 1 -1 1 -1]  
Output: [ 1. -1. 1. -1.]  
  
Input: [-1 1 -1 1]  
Output: [-1. 1. -1. 1.]  
  
Input: [-1 -1 -1 -1]  
Output: [-1. -1. -1. -1.]