Code for Recognising digits task

```
clear all
close all
clc

% Storing all the patterns
x1=[ [ -1, -1, -1, -1, -1, -1, -1, -1, -1, -1],[ -1, -1, -1, 1, 1, 1, 1, -1, -1, -1],[ -1, -1, 1
x2=[ [ -1, -1, -1, 1, 1, 1, 1, -1, -1, -1],[ -1, -1, -1, 1, 1, 1, 1, -1, -1, -1],[ -1, -1, -1, 1
x3=[ [ 1, 1, 1, 1, 1, 1, 1, 1, -1, -1],[ 1, 1, 1, 1, 1, 1, 1, 1, -1, -1],[ -1, -1, -1, -1, -1, 1
x4=[ [ -1, -1, 1, 1, 1, 1, 1, 1, -1, -1],[ -1, -1, 1, 1, 1, 1, 1, 1, 1, -1],[ -1, -1, -1, -1, -1
x5=[ [ -1, 1, 1, -1, -1, -1, -1, 1, 1, -1],[ -1, 1, 1, -1, -1, -1, -1, 1, 1, -1],[ -1, 1, 1, -1,
% Storing the pattern that is going to be feeded
feedPattern = [[1, -1, -1, 1, 1, 1, 1, -1, -1, 1], [1, -1, -1, 1, 1, 1, 1, -1, -1, 1], [1, -1, -
% Getting all the stored patterns into one matrix
storedPatterns = [x1; x2; x3; x4; x5]';
N = length(storedPatterns);

% Creating the weight matrix by dotproduct of the stored patterns
weightMatrix = storedPatterns * storedPatterns';
weightMatrix = weightMatrix./N;
n = size(weightMatrix,1);
for i = 1:size(weightMatrix,1)
    weightMatrix(i,i) = 0;      % making the diagonal weights to zero
end

updatedPattern = feedPattern;   % setting the updated pattern to feed pattern because
                                % The first iteration will be with the feed
                                % pattern
while true
    % Updating the pattern asynchronous determistic
    for i = 1:N
        weight = weightMatrix(i,:);
        localField = weight * updatedPattern;
        if localField == 0
            localField = 1;
        end
        newBit = sign(localField);
        updatedPattern(i,1) = newBit;

    end
    if isequal(updatedPattern,feedPattern)     % If updated pattern gets
                                               % to steady state, break
        break
    else
        feedPattern = updatedPattern;
    end
end
```

```
% Looping through all stored patterns to see if the updated pattern
% has converged to one of those (and also their inverted form)
for i = 1:5
    if isequal(updatedPattern,storedPatterns(:,i))
        fprintf(’The updated pattern corresponds to the following pattern index %.0f\n’, i)
    end
    if isequal(updatedPattern,-storedPatterns(:,i)) % Checks for inverted
        fprintf(’The updated pattern corresponds to the following pattern index %.0f\n’, -i)
    end
end

% To print the result. In other words the updated pattern.
fprintf([’[’ strjoin(repmat({’%.17g’},1,size(updatedPattern’,2)), ’, ’) ’],\n’], updatedPattern.
```