```
clear;

x1=[ [ -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, 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, 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, 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, -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],[ -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, -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, 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],[ -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, 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, -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,
-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, 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, 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, -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, -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, -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],[ 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, 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, -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, -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, 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, -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,
-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, 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, -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],[ -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, 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, -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,
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, -1, -1, -1, 1, 1, -1] ];

% x = [[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, -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, 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, 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, -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], [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, -1, -1, 1, -1]];
x = [[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, 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, -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, -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, 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], [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, 1, 1, -1, -1]];
% x = [[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, -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, 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, 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, -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], [-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, 1, -1, -1, 1]];

xCombination = [x1; x2; x3; x4; x5];
networkState = x;
networkWeight = HebbsRule(xCombination);

runResult = RunHopfieldNetwork(networkWeight, networkState, x);

runResultReshapedTransposed = ReshapeTranspose(runResult);
PlotResult(runResultReshapedTransposed);
PrintResult(runResultReshapedTransposed);
```

```
function networkWeight = HebbsRule(xCombination)

networkWeight = zeros(160);
for i = 1:5
    networkWeight = networkWeight + (1/160) * transpose(xCombination(i,:)) *
xCombination(i,:);
end
for j = 1:160
    for k = 1:160
        if j == k
            networkWeight(j,k) = 0;
        end
    end
end


end

function runResult = RunHopfieldNetwork(networkWeight, networkState, x)
```

```
for i = 1:160
    networkState(i) = sign(networkWeight(i,:) * transpose(networkState));
    if networkState(i) == 0
        networkState(i) = 1;
    end
end
while isequal(networkState, x)
    for i = 1:160
        networkState(i) = sign(networkWeight(i,:) * transpose(networkState));
        if networkState(i) == 0
            networkState(i) = 1;
        end
    end
end

runResult = networkState;

end

function runResultReshapedTransposed = ReshapeTranspose(runResult)

runResultReshaped = reshape(runResult,10,16);
runResultReshapedTransposed = transpose(runResultReshaped);

end

function PlotResult(runResultReshapedTransposed)

imagesc(runResultReshapedTransposed);
colormap(flipud(gray));
axis image;

end

function PrintResult(runResultReshapedTransposed)

formatSpecFirst = '[[%d, %d, %d, %d, %d, %d, %d, %d, %d, %d], ';
formatSpecMiddle = '[%d, %d, %d, %d, %d, %d, %d, %d, %d, %d], ';
formatSpecLast = '[%d, %d, %d, %d, %d, %d, %d, %d, %d, %d]]';
for j = 1:16
    if j == 1
        fprintf(formatSpecFirst, runResultReshapedTransposed(j,:));
    elseif j == 16
            fprintf(formatSpecLast, runResultReshapedTransposed(j,:));
    else
        fprintf(formatSpecMiddle, runResultReshapedTransposed(j,:));
    end
end
```

```
end
```