```
clc; clear;
%%
% Name: Devosmita Chatterjee
% Assignment3 a
experimentnumber=100;
m_experimentnumber=zeros(1,experimentnumber);
for i =1:1:experimentnumber
    p = 7;

    N = 200;% number of pixels of each pattern

    % Input data
    mat = randi([0 1], N,p);% Generate random pattern

    x = zeros(N,p);

    for a = 1:1:N
        for j = 1:1:p
            if mat(a,j) == 0
                x(a,j) = 1;
            else
                x(a,j) = -1;
            end
        end
    end

    % Calculate weight matrix
    W = zeros(N,N);

    for j = 1:p
        W = x(:,j)*x(:,j)'+W;
    end
    W = (1/N)*W;
    W = W - diag(diag(W));%1
 %%
    T = 2*10^5;
    m = zeros(1,T);
    m(1) = 1;
    c = x(:,1);
   for t =2:T
        a1 = randi([1 N],1,1);% Generate randomly chosen neuron for the
asynchronous update%2


        s = W(a1, :) * c;


        beta = 2;
        prob = 1/(1+exp(-2*beta*s));

        r = rand%0.1*(randi(11)-1);

        if (le(r,prob))
```

```
            out = 1;%3
        else
            out = -1;
        end

        S=c;
        S(a1)=out;


        m(t) = (1/N)*S'*x(:,1);%4
        c = S;

    end%5

    m_sum = (1/T)*sum(m);%6
    m_experimentnumber(i) = m_sum;
end%7
%%
avg=(1/experimentnumber)*sum(m_experimentnumber);%8
fprintf('%.3f\n',avg)


%0.845
```

```
clc; clear;
%%
% Name: Devosmita Chatterjee
% Assignment3 b
experimentnumber=100;
m_experimentnumber=zeros(1,experimentnumber);
for i =1:1:experimentnumber
    p = 45;

    N = 200;% number of pixels of each pattern

    % Input data
    mat = randi([0 1], N,p);% Generate random pattern

    x = zeros(N,p);

    for a = 1:1:N
```

```
        for j = 1:1:p
            if mat(a,j) == 0
                x(a,j) = 1;
            else
                x(a,j) = -1;
            end
        end
    end

    % Calculate weight matrix
    W = zeros(N,N);

    for j = 1:p
        W = x(:,j)*x(:,j)'+W;
    end
    W = (1/N)*W;
    W = W - diag(diag(W));%1
 %%
    T = 2*10^5;
    m = zeros(1,T);
    m(1) = 1;
    c = x(:,1);
   for t =2:T
        a1 = randi([1 N],1,1);% Generate randomly chosen neuron for the
asynchronous update%2


        s = W(a1, :) * c;


        beta = 2;
        prob = 1/(1+exp(-2*beta*s));

        r = rand%0.1*(randi(11)-1);

        if (le(r,prob))
            out = 1;%3
        else
            out = -1;
        end

        S=c;
        S(a1)=out;


        m(t) = (1/N)*S'*x(:,1);%4
        c = S;

    end%5

    m_sum = (1/T)*sum(m);%6
    m_experimentnumber(i) = m_sum;
end%7
```

```
%%
avg=(1/experimentnumber)*sum(m_experimentnumber);%8
fprintf('%.3f\n',avg)


%0.144
```