

```
%-----  
% Tony Hyun Kim  
% CS 246, PS 3, Problem 1  
% Computing similarity by "hand"  
%-----  
clear all;  
  
% Adjacency matrix  
R = [0 1 1 0 1;  
      0 1 0 1 0;  
      1 0 0 0 0];  
  
% Initialize the similarity matrices  
nA = 3; sA = eye(nA);  
nB = 5; sB = eye(nB);  
  
% Decay constants  
C1 = 0.8;  
C2 = 0.8;  
  
Niter = 3;  
for iter = 1:Niter  
  
    % Compute the next sA matrix  
    sAnew = 0.5*eye(nA);  
    for i = 1:(nA-1)  
        for j = (i+1):nA  
            Oi = find(R(i,:));  
            Oj = find(R(j,:));  
            for oi = Oi  
                for oj = Oj  
                    sAnew(i,j) = sAnew(i,j) + sB(oi,oj);  
                end  
            end  
            sAnew(i,j) = C1/(length(Oi)*length(Oj))*sAnew(i,j);  
        end  
    end  
    sAnew = sAnew + sAnew'; % Symmetric matrix  
  
    % Compute the next sB matrix  
    sBnew = 0.5*eye(nB);  
    for i = 1:(nB-1)  
        for j = (i+1):nB  
            Ii = find(R(:,i));  
            Ij = find(R(:,j));  
            for ii = Ii'  
                for ij = Ij'  
                    sBnew(i,j) = sBnew(i,j) + sA(ii,ij);  
                end  
            end  
            sBnew(i,j) = C2/(length(Ii)*length(Ij))*sBnew(i,j);  
        end  
    end  
    sBnew = sBnew + sBnew';  
  
    % Batch update  
    fprintf('Iter %d\n',iter);  
    sA = sAnew;  
    sB = sBnew;  
    disp(sA);  
    disp(sB);  
end
```