A4, Q5 (a)

```
function [p, iters] = PageRank(G, alpha)

    outlinks = sum(G);
    d = (outlinks==0);
    D = diag(1./(outlinks+d));

    P = G * D;   % scale columns by 1/deg(j)
    R = size(P,1);

    e = ones(R,1);
    p = ones(R,1) ./ R;   % initial probability vector

    iters = 0;
    err = 1.0;
    maxIters = 10000;

    while err > 1e-8 && iters < maxIters
        % Note: For efficiency, do (d*p) before e*d
        Mp = alpha*(P*p + e*(d*p)/R) + (1-alpha)/R*e;
        err = norm(p-Mp, inf);
        p = Mp;
        iters = iters + 1;
    end

    if iters == maxIters
        disp('Failed to converge');
    end
```

A4, Q5(b)

```
%% Google script

if 1  % Note: 1 means true, 0 means false

    clear G;
    clear U;

    % Place your code to create G and U here

    U{1} = 'a';
    U{2} = 'b';
    U{3} = 'c';
    U{4} = 'd';
    U{5} = 'e';
    U{6} = 'f';

    G(1,3) = 1;
    G(1,4) = 1;
    G(2,1) = 1;
    G(2,6) = 1;
    G(3,2) = 1;
    G(3,5) = 1;
    G(4,2) = 1;
    G(5,4) = 1;
    G(6,3) = 1;
    G(6,5) = 1;

else

    % Load a larger graph
    load cs.mat

end

%% Page Rank algorithm

alpha = 0.85;
[p, iters] = PageRank(G, alpha);


%% Output

disp(['Converged in ' num2str(iters) ' iterations']);

% Display the first 6 scores
p(1:6)

bar(p);

[y I] = sort(p, 'descend');

for n = 1:min(length(I),10)
    disp([num2str(n) ': ' U{I(n)}]);
end
```
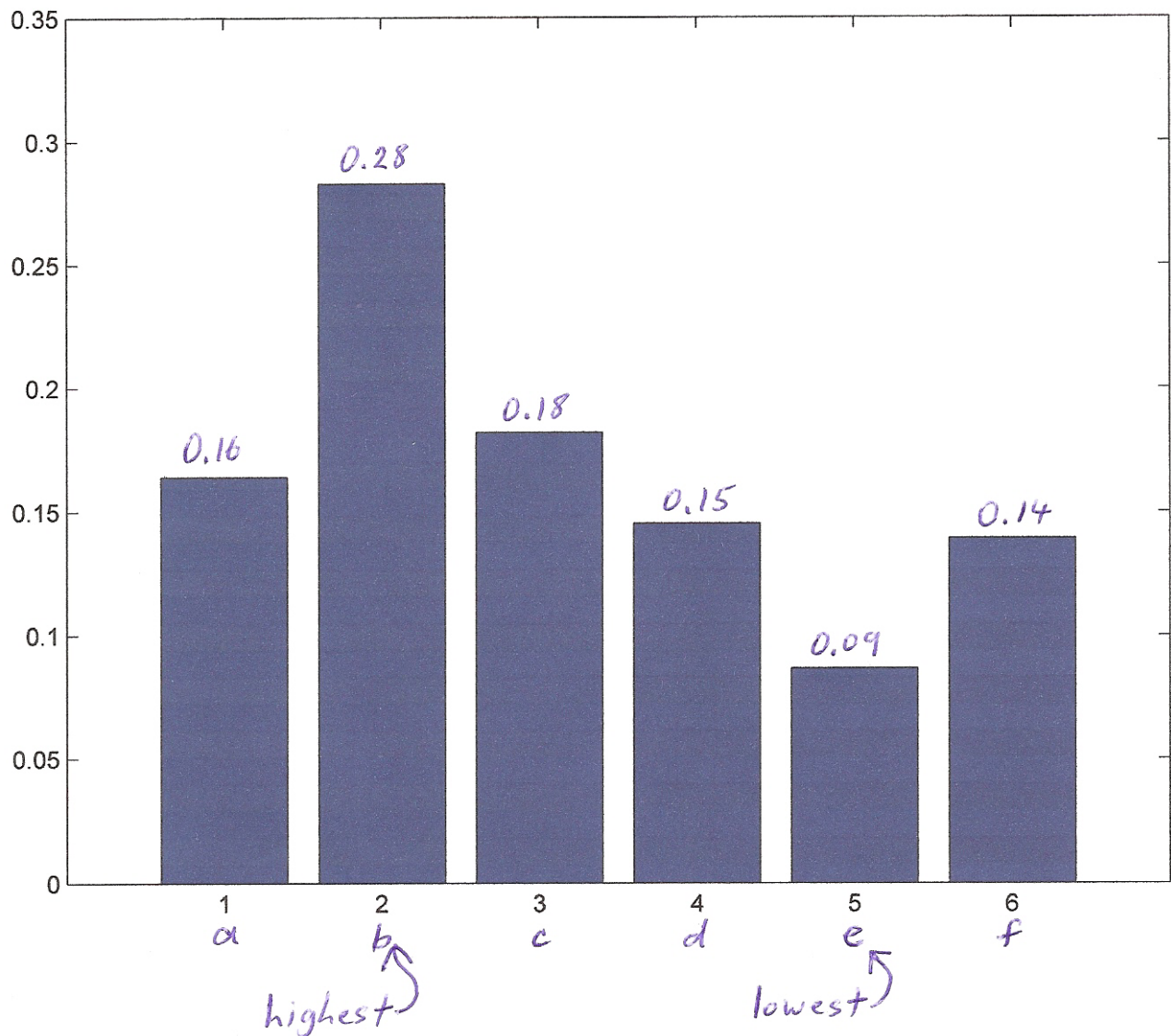
A4, Q5 (b) output   (19)

Converged in 48 iterations

ans =

```
    0.1641
    0.2828
    0.1820          } the 6 scores
    0.1452
    0.0867
    0.1392
```

```
1: b
2: c
3: a          } Rank order of the nodes
4: d
5: f
6: e
>>
```


0.16   0.28   0.18   0.15   0.09   0.14

1 a   2 b   3 c   4 d   5 e   6 f

highest   lowest

A.4, Q5(c), Part(i)

For example, we get the following results for the number of iterations.

α	#iterations
0.0	1
0.1	6
0.4	14
0.7	35
0.85	72
0.95	189
1.0	804

We see that the number of iterations goes up rapidly as α increases.

Part(ii)

A large α gives better information about the network topology, but for α near 1 it over-emphasizes cycles.

A small value of α leads to faster convergence, but gives less information about the network topology.