COS 435, Spring 2008 - Problem Set 2
Due at 5:00pm, Friday, Feb. 29, 2008

Collaboration Policy
You may discuss the general methods of solving the problems with other students in the class. However, each student must work out the details and write up his or her own solution to each problem independently.

Lateness Policy
A late penalty will be applied, unless there are extraordinary circumstances and/or prior arrangements:

* Penalized 10% of the earned score if submitted by 5:00 pm Saturday (3/1/08).
* Penalized 25% of the earned score if submitted by 1:30 pm Wednesday(3/5/08).
* Penalized 50% if submitted later than 1:30 pm Wednesday (3/5/08).

Problems

Problem 1.
In class we discussed the LSI example appearing in the original paper *Indexing by Latent Semantic Analysis* by Deerwester, Dumais, et. al. (Journal of the Society for Information Science, 41(6), 1990, 391-407). The term-document matrix and the matrices of the reduced-dimension approximation (k=2) are shown below.

Terms	**Documents**								
	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

$$\mathbf{V'}_k^T = \begin{pmatrix} 0.20 & 0.61 & 0.46 & 0.54 & 0.28 & 0.00 & 0.02 & 0.02 & 0.08 \\ -0.06 & 0.17 & -0.13 & -0.23 & 0.11 & 0.19 & 0.44 & 0.62 & 0.53 \end{pmatrix}$$

$$\mathbf{\Sigma'}_k = \begin{pmatrix} 3.34 & \\ & 2.54 \end{pmatrix}$$

$$\mathbf{U'}_k = \begin{pmatrix} 0.22 & -0.11 \\ 0.20 & -0.07 \\ 0.24 & 0.04 \\ 0.40 & 0.06 \\ 0.64 & -0.17 \\ 0.27 & 0.11 \\ 0.27 & 0.11 \\ 0.30 & -0.14 \\ 0.21 & 0.27 \\ 0.01 & 0.49 \\ 0.04 & 0.62 \\ 0.03 & 0.45 \end{pmatrix}$$

$$\mathbf{C}_k = \begin{pmatrix} 0.16 & 0.40 & 0.38 & 0.47 & 0.18 & -0.05 & -0.12 & -0.16 & -0.09 \\ 0.14 & 0.37 & 0.33 & 0.40 & 0.16 & -0.03 & -0.07 & -0.10 & -0.04 \\ 0.15 & 0.51 & 0.36 & 0.41 & 0.24 & 0.02 & 0.06 & 0.09 & 0.12 \\ 0.26 & 0.84 & 0.61 & 0.70 & 0.39 & 0.03 & 0.08 & 0.12 & 0.19 \\ 0.45 & 1.23 & 1.05 & 1.27 & 0.56 & -0.07 & -0.15 & -0.21 & -0.05 \\ 0.16 & 0.58 & 0.38 & 0.42 & 0.28 & 0.06 & 0.13 & 0.19 & 0.22 \\ 0.16 & 0.58 & 0.38 & 0.42 & 0.28 & 0.06 & 0.13 & 0.19 & 0.22 \\ 0.22 & 0.55 & 0.51 & 0.63 & 0.24 & -0.07 & -0.14 & -0.20 & -0.11 \\ 0.10 & 0.53 & 0.23 & 0.21 & 0.27 & 0.14 & 0.31 & 0.44 & 0.42 \\ -0.06 & 0.23 & -0.14 & -0.27 & 0.14 & 0.24 & 0.55 & 0.77 & 0.66 \\ -0.06 & 0.34 & -0.15 & -0.30 & 0.20 & 0.31 & 0.69 & 0.98 & 0.85 \\ -0.04 & 0.25 & -0.10 & -0.21 & 0.15 & 0.22 & 0.50 & 0.71 & 0.62 \end{pmatrix}$$

Not needed. Provided for your information.

Problem 1 Part a: Give the scores of the 9 documents for the query ***trees, minors*** using the dimension 2 (k=2) LSI approximation.

Problem 1 Part b: Give the scores of the 9 documents for the query ***trees, minors*** using the standard vector model. For both the documents and the query, start with 0/1 vectors, then normalize them to unit vectors and use the cosine similarity measure. It suffices to calculate the vector components that will actually affect the computation.

Problem 2

Part a. Implement the PageRank Algorithm in Java (preferred) or C as an *iterative* algorithm as discussed in class. Below are three graphs on which to test your algorithm. For each graph, run the algorithm for 100 iterations. After examining this data, state what you think the convergence criterion should be in terms of the difference between PageRank values of successive iterations. This convergence criterion should be *for all inputs*, not just a particular graph.

Part b. Modify your code to use the convergence criterion you determined in Part a. This code must print the PageRank values after each iteration.

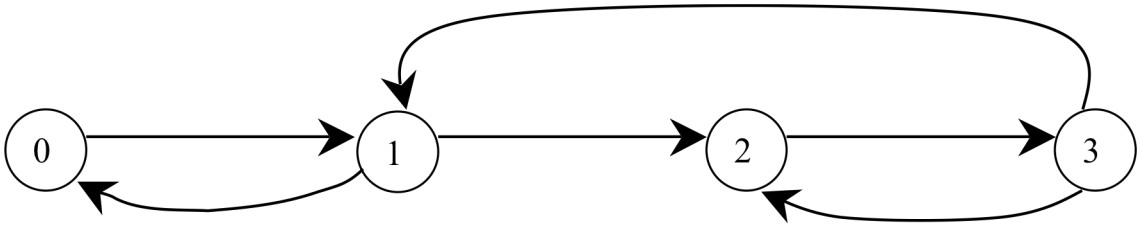
Part c. Examine the PageRank results for the three graphs. Do they match your intuition about the importance of various nodes? Explain. How does the difference in structure between Graph 2 and Graph 3 affect their PageRanks? Does this make sense?

What to hand in: Hand in your program for **Part b** and files or printouts containing the output of this program for each of the graphs. Also hand in your justification for your choice of convergence criterion based on **Part a** and your answers to the questions in **Part c**.

Save your code for this problem. You will need it next week.

Input for your programs. We are providing a simple text file for each graph. The first line of the file contains the number of nodes. Each successive line lists one edge by listing the node at which the edge starts followed by the node to which the edge goes. Nodes are numbered 0 to n-1. The input file contents are shown below each graph.

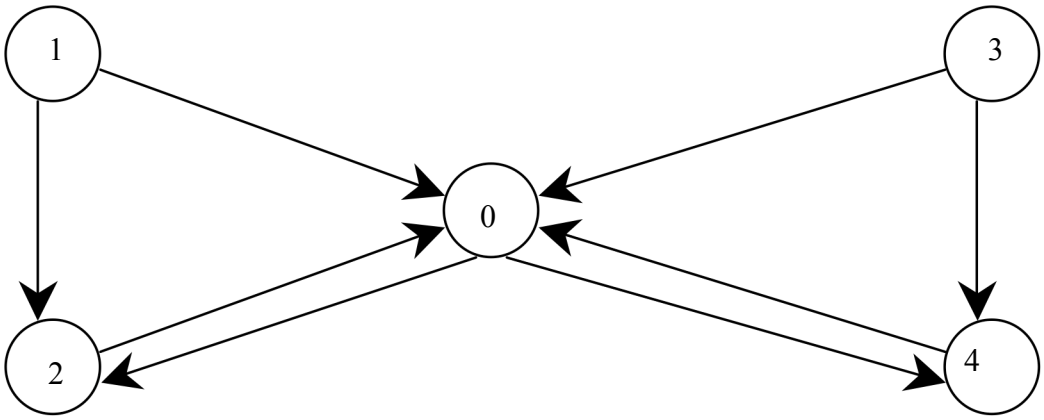
Graph 1



File HW2_graph1.txt:

```
4
0  1
1  2
1  0
2  3
3  1
3  2
```

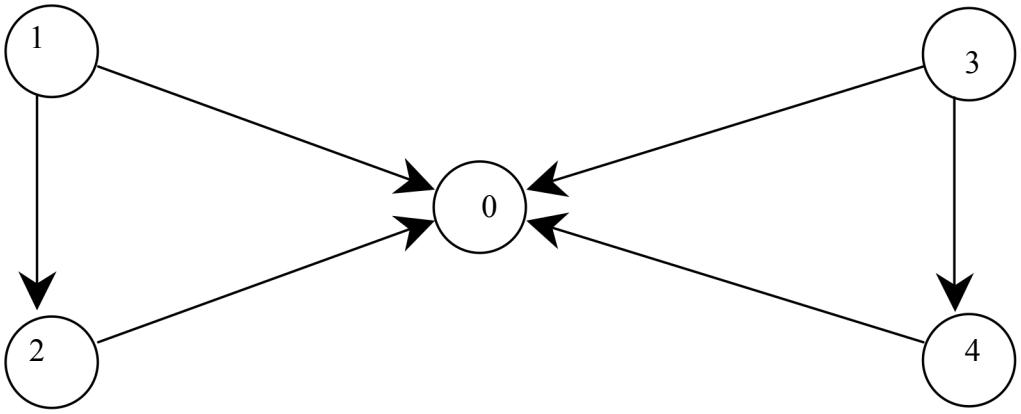
Graph 2



File HW2_graph2.txt:

```
5
0  2
0  4
1  2
1  0
2  0
3  0
3  4
4  0
```

Graph 3



File HW2_graph3.txt:

```
5
1  2
1  0
2  0
3  0
3  4
4  0
```