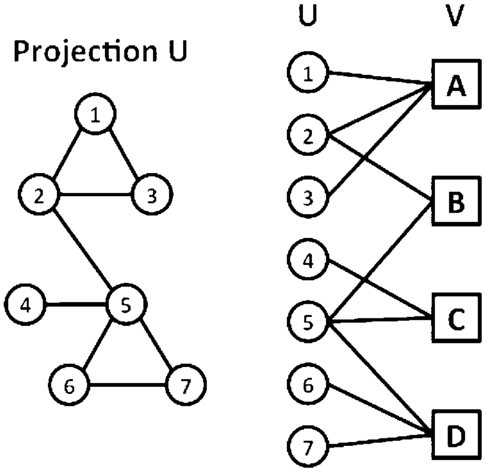
Question: MapReduce for Graph Analysis

Consider a large graph that is represented by a list of edges in a file where each row is in the format “[v1],[v2]” and represents an *undirected* edge between vertex [v1] and vertex [v2].

(a) Design a Map-Reduce program that computes the degree distribution for this graph, i.e., what is the degree for each vertex. You can assume you are reading the file line by line and each line is formatted as “[v1],[v2]”. Provide the pseudocode for the map and the reduce functions.

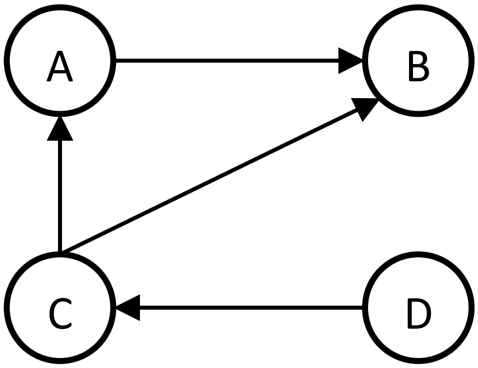
(b) For question (a), can you design a combiner (i.e., a *combine* function) function that would reduce network time by pre-aggregating values in the mapper?
Note: If you do need to revise the map or reduce functions in (a), describe the revised functions.

(c) You are given a large *undirected bi-partite* graph in the same format, i.e., rows of “[v1],[v2]”. As a reminder, a bi-partite graph is a graph whose vertices can be divided into two disjoint sets U and V such that every edge in the graph is connecting one vertex from set U with one vertex from set V. You can assume [v1] is always from set U, while [v2] is always from set V.
Write a map-reduce program that computes a projection of the bi-partite graph onto set U, i.e., a new graph whose vertices are the ones from set U and each two vertices in U are connected by an edge if they were both connected to the same vertex in V in the original graph. The image below shows an example for a bi-partite graph and its projection over U. The new graph should be written out in the same format as the input (i.e., a list of edges).



Question: PageRank

Consider the following graph. You would like to compute the PageRank score for each node using a simple power iterations method that takes in as input only the matrix M and solves $r = Mr$. Fortunately, the graph is small enough to keep the whole matrix in memory. You decide to set $\beta = 1.0$ and teleport immediately from dead-ends.



a. Provide the matrix M you will use for the power iterations method to compute PageRank.

	A	**B**	**C**	**D**
A				
B				
C				
D				

b. Run the power iterations for three steps (i.e., t=0,1,2) and report the obtained PageRank scores for each node (you can round scores to four decimal places).

c. You are given a graph G (you do not know the number of nodes and edges or the structure of the graph). PageRank scores were computed for all nodes in the graph using $\beta = 0.8$ and immediate teleporting out of dead-ends. You know that n_H is the node with the highest PageRank score while n_L is the node with the lowest PageRank score. **For each of the following, determine if it will increase/decrease/not change/unable to determine. Briefly justify your answer.**

i. Increasing β will ___________ the PageRank (PR) score of n_H

ii. Increasing β will ___________ the PR score of n_L

Question: Personalized PageRank

Consider an online repository with a large set of academic papers where each paper is associated with a set of authors. Users of the repository are using it to read papers of interest related to their research. You wish to build a recommender system that can recommend papers to researchers based on the papers they recently read using **personalized PageRank (random walk with restarts)**. The only information you can utilize to build your system is a simple text file with many rows of the format "#2323: [@98, @456, @718]" that indicates paper number 2323 was written by authors number 98, 456, and 718. You do not have access to the text of the paper or the names of the authors.

Answer the following questions:

a. Describe the graph you would use: What are the nodes? What are the edges? Directed or undirected graph? Is it a Bipartite graph?

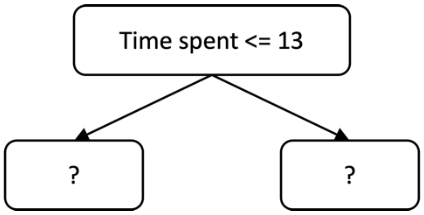
b. Assuming a user of the repository has recently read papers #255, #1023, #2047. You need to recommend five papers the user is likely to be interested in reading. How would you do that?

c. Will your system be able to recommend new articles that were just added to the repository and have not yet been read by any user? Justify your answer.

Question: Machine Learning

Recall the e-commerce dataset from the midterm. The dataset shows for each session on the website how long the customer spent on the website, whether they are new members, whether they were offered a discount, and whether they made a purchase.
You are training a decision tree to predict whether a customer will make a purchase.
The algorithm has already selected the root node split to be "Time spent <= 13". Next, you need to decide what should be the right and the left child nodes of the root node. Assume you cannot split on "Time spent" again and the only stopping criteria is when the leaf is pure. For each of the left and right nodes, decide whether it should split on "Discount Offered", split on "New Member", or be a leaf node. Justify your answer.

Time spent (minutes)	**Discount Offered**	**New Member**	**Purchase**
20	Yes	Yes	Yes
18	Yes	No	Yes
5	Yes	Yes	No
15	No	No	No
8	No	No	No
7	Yes	Yes	No
15	Yes	No	Yes
13	No	Yes	No
22	No	No	No



Question: Latent-Factor Recommender Systems

Consider using the specialized "SVD" method covered in class to decompose a Users-Movies ratings matrix M into two matrices: Q and P^T. Each cell is either contains the rating assigned by the corresponding user to the corresponding movie in a scale of 1-5 or contains no value if we the user has not rated a movie. We also have a held-out test set with 10% of the ratings that have been removed from the training matrix.

a. How do you expect the testing accuracy to change as a function of the number of components? Provide a sketch of a plot of the testing accuracy vs. number of components (the exact numbers are not important, just a sketch of the trends). Justify your plot.

b. After decomposition you obtain the following Q and P^T:

Matrix Q

	Factors		
Movies 1-5	-1.	1.8	.0
	1.1	1.5	1.4
	.3	1.2	1.5
	.7	2.4	-.4
	-.8	1.4	1.6

Matrix P^T

	Users 1-10									
Factors	.5	-.5	-.2	1.3	.1	.7	-.5	.0	-1.2	.7
	.7	.6	2.1	1.8	2.0	1.0	1.7	1.8	1.4	1.7
	2.2	0.9	-.3	1.0	1.0	.7	1.6	-.3	-.7	-.1

Based on Q and P^T, predict the rating for the following user-movie pairs (show your computation):

a. (User 3, Movie 2)

b. (User 8, Movie 4)

c. Your friend claimed that movies that the user likes are likely to be located near the user in learned latent factor space and therefore you can look at the nearest neighbors of each user to detect movies they have not watched and recommend them to the user. Do you think your friend's claim that movies that a user likes are likely to be located near the user in the learned latent factor space is correct? Justify your answer.

d. Some recommender systems have trouble making predictions of ratings for items that have just been added to the system and have only been rated by a small number of users. Would the latent-factor recommender system described above suffer from the same problem or not? Justify your answer.