67978: A Needle in a Data Haystack
Introduction to Data Science
Homework #2: Similar Items, Graphs
Due: **28 Dec, 11:50pm, on Moodle**

THE HEBREW UNIVERSITY OF JERUSALEM

Instructions: Unless stated otherwise, submission is done **individually**. We rely on trust. You may discuss assignments verbally, but do not share solutions with other students. Cheating is a serious offense and will result in severe penalties.

Files should be zipped to **ex_2_First_Last.zip** (with your first and last name). The theoretical part should be a single file in pdf format only (no docx or jpg). If you're submitting handwritten answers, make sure they are crystal clear (both the handwriting and the scan – please avoid shaky cell pictures taken in a dark room :)). Submissions which fail to comply will not be graded.
Important: We are experimenting with having you submit some answers directly through the moodle (to make grading much faster!). Instructions coming soon.

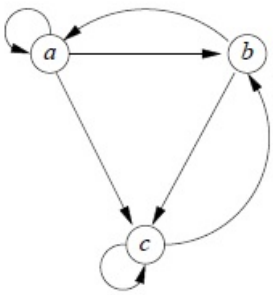
Problem 1 (Finding Similar Items).

(a) What are the first ten 4-shingles in the instructions above? (Use shingle = character, including spaces and punctuation. Ignore the "Instructions:" header and the spaces that follow it.)

(b) Prove or disprove: if the Jaccard similarity of two columns is 0, then minhashing always gives a correct estimate of the Jaccard similarity.

(c) One might expect that we could estimate the Jaccard similarity of columns without using all possible permutations of rows. For example, we could only allow cyclic permutations; start at a randomly chosen row r, which becomes the first in the order, followed by rows $r+1$, $r+2$, and so on, down to the last row, and then continuing with the first row, second row, and so on, down to row $r-1$. There are only n such permutations if there are n rows. However, these permutations are not sufficient to estimate the Jaccard similarity correctly. Give an example of a two-column matrix where averaging over all the cyclic permutations does not give the Jaccard similarity. Compute Jaccard and average similarity.

Problem 2 (PageRank).
Compute the PageRank of each page in the graph below:



(a) Assuming no teleports, $\beta = 0.7$.

(b) Assuming $\beta = 0.85$, regular teleports.

(c) Assuming $\beta = 0.85$ and the teleport set is $\{c\}$

THE HEBREW UNIVERSITY OF JERUSALEM

Problem 3 (Community Detection (Coding Question)).

Download the Facebook ego-network data. Each line represents an undirected edge: "881 858" means that nodes 881 and 858 are connected. This is not a multigraph (no parallel edges, no self-loops). Clean the data, if you need to. :)

You should submit two things: The code and the answers. You can use existing packages for handling graphs, but you need to write the clique percolation part yourself.

(a) How many nodes and edges are in the graph?

(b) How many connected components in the graph? How many nodes are in the largest connected component?

(c) Implement clique percolation (yourself) and apply it to the ego-network graph. How many communities are discovered after running clique percolation with with 4-cliques, and which nodes are their members? (Print them sorted, please)

Problem 4 (Meta). How long (in hours) did this assignment take?