Homework 2: A MapReduce Application for Company Recommendation (Due at 5:15pm April 13 Wed.)

Suppose there is a graph for company connections, where each node represents a company (eg. C1, C2 …) and each edge represents exiting connection/business between two companies. Suppose companies want to expand their business. They are looking for a **MapReduce** application to find companies which they don't direct connection to but are only **2 hops** away in the graph. In the following, we will call them 2-hop companies. Moreover, for a company C, **the connectivity of its 2-hop company is defined to be the number of distinct 2-hop paths between them.**

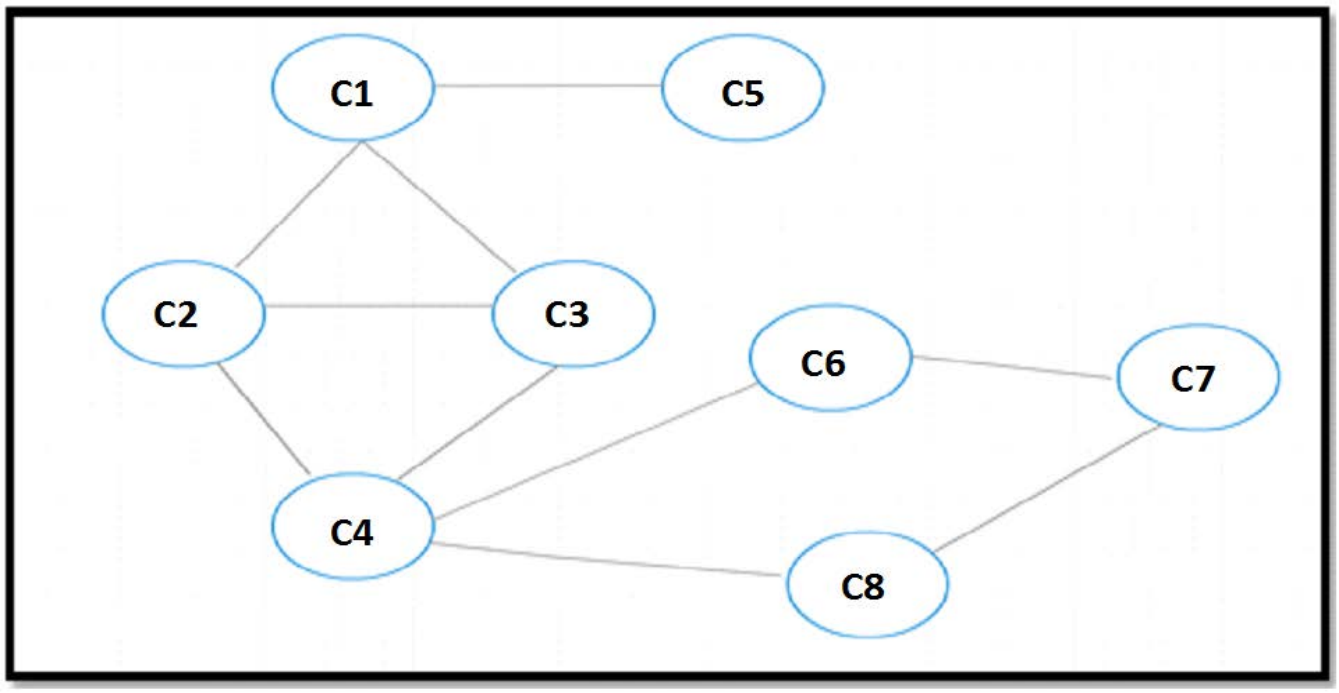
For example,

For C1, below are the two hop paths:
C1-C2-C4, C1-C3-C4, C1-C2-C3, C1-C3-C2.

Since C2 and C3 have direct connection with C1, so these are NOT 2-hop companies of C1. There are two distinct 2-hop paths for C4: C1-C2-C4 and C1-C3-C4, so for C1, the connectivity of C4 is 2.

Requirements:

1. For each company, recommend **2-hop companies.**
2. Output should be **sorted in the descending order of connectivity. If there is a tie, the node with smaller ID wins.**
   Eg. For C8, we have following 2-hop companies C6 (2), C2 (1) and C3 (1), the output should be: [C6:2],[C2:1],[C3:1] =>C8
   Note that the number besides each NodeID is its connectivity.



INPUT (the direct neighbors of each company):

C1:C2,C3,C5
C2:C1,C3,C4
C3:C1,C2,C4
C4:C2,C3,C6,C8
C5:C1
C6:C4,C7
C7:C6,C8
C8:C4,C7

OUTPUT:

[C4:2]	=>C1
[C5:1], [C6:1], [C8:1]	=>C2
[C5:1], [C6:1], [C8:1]	=>C3
[C1:2], [C7:2],	=>C4
[C2:1], [C3:1],	=>C5
[C8:2], [C2:1], [C3:1]	=>C6
[C4:2]	=>C7
[C6:2], [C2:1], [C3:1]	=>C8

Note that:

1. **You need to write your program using Java. Put the output file right at the root directory of the project.**
2. **You can test your program with the given input.txt; but the TAs will use a different input file for grading.**
3. **Every student needs to include the source code and the executable file in a single ZIP file "YourSUID-HW2.zip" and submit it to Blackboard before deadline. A hard copy should be turned in right before the class on the due date.**
4. **Sample code from the WordCount project is given for your reference.**
5. **You MUST write your own program. Copying codes from others or some online sources will result in serious consequences!!!**