# Homework #5: Computing Edge Betweenness Using Spark

TA handling this homework: Xueping Li [xuepingl@usc.edu](mailto:xuepingl@usc.edu)

## Due: December 6, Friday

## 100 points

The task of this homework is to implement the GN algorithm in parallel using Spark. One way to proceed is to compute the DAG graphs and partial fractions in parallel. Recall that if the graph has no nodes, then no DAG graphs will need to be generated, and partial scores for the edge betweenness are collected from each graph.

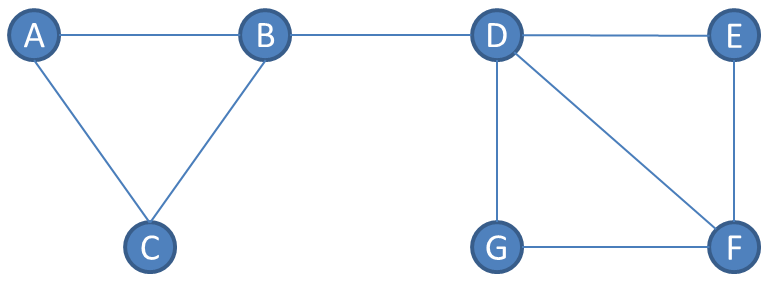
You may use the networkx package and its BFS implementation. **However, you may not use its implementation of edge betweenness.**

Name your script **firstname\_lastname\_gn.py** You will lose points if you do not do this.

And you only need to submit **firstname\_lastname\_gn.py** file.

### Input format

The input graph will be provided in a text file where each line represents an edge in the graph, with end points separated by comma. For example, the input for the graph below is as follows.



A,B

A,C

…

### Output format

You should print the betweenness of edges to **stdout** in a format as follows.

Sort your results by first node then second node in alphabetically increasing order. You will lose points if you do not do this.

(A, B), 5.0

(A, C), 1.0

(B, C), 5.0

(B, D), 12.0

(D, E), 4.5

(D, F), 4.0

(D, G), 4.5

(E, F), 1.5

(F, G), 1.5

### Execution format

### spark-submit firstname\_lastname\_gn.py <input-data-file>