­­ADM – Homework4

*Giacomo Parmendola(1462237)*

*Vigèr Durand Azimedem Tsafack(1792126)*

*Mattia Podio()*

* **First Step**: Graph creation

1. Here we process the JSON file and create a graph with all the authors as nodes. Two nodes are connected if they share, at least, one publication and the weight of an edge is computed in this way: .

Find the python code in the file *graph\_creation.py.*

*output:*



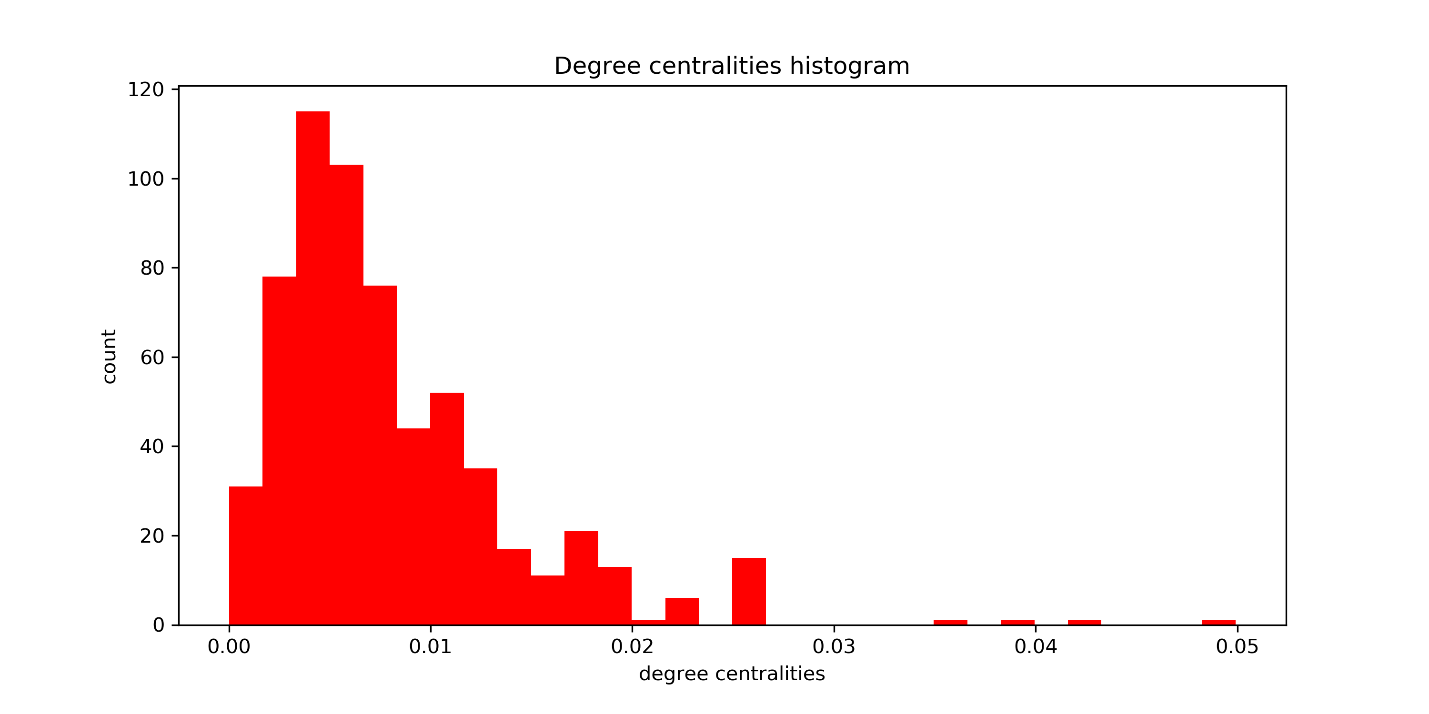
* **Second Step**: Some statistics

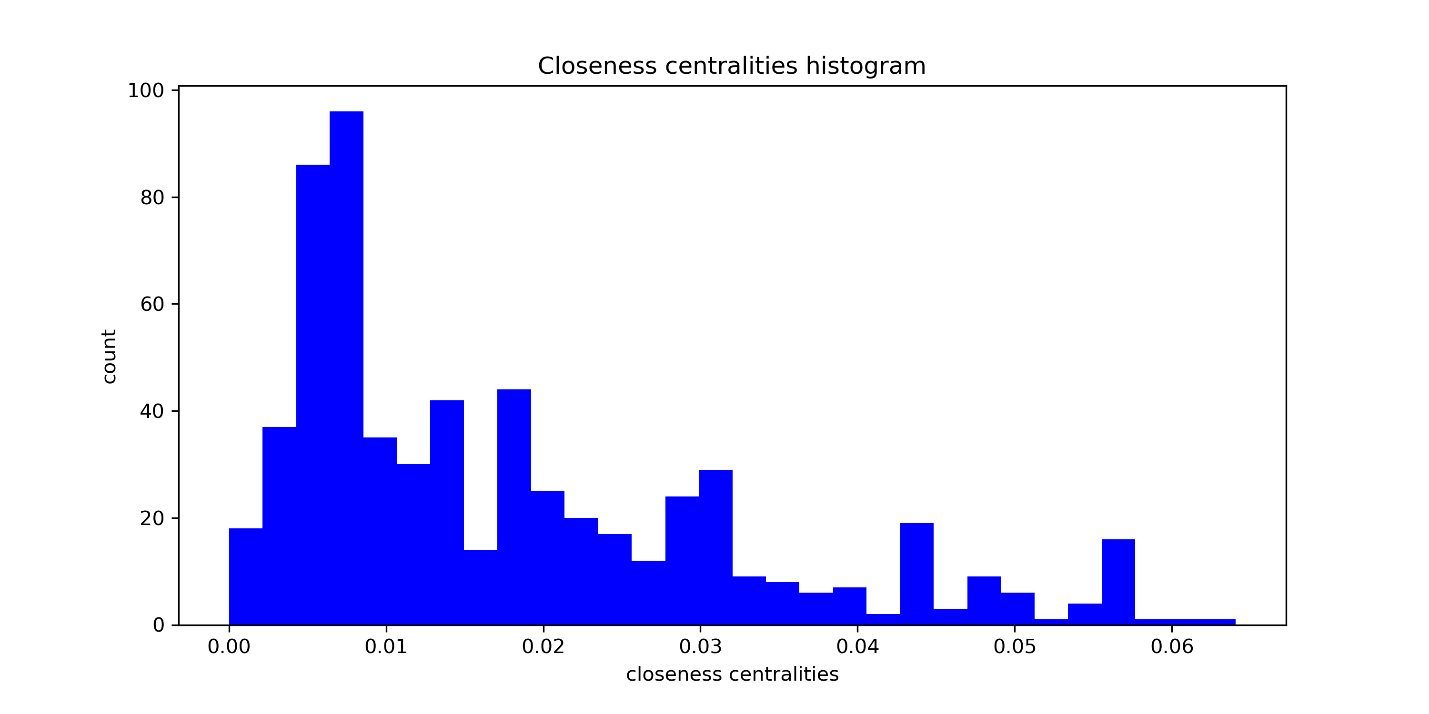
1. Given a conference in input, return the subgraph induced by the set of authors who published at the input conference at least once: Find the python code in the file *graph\_creation.py*

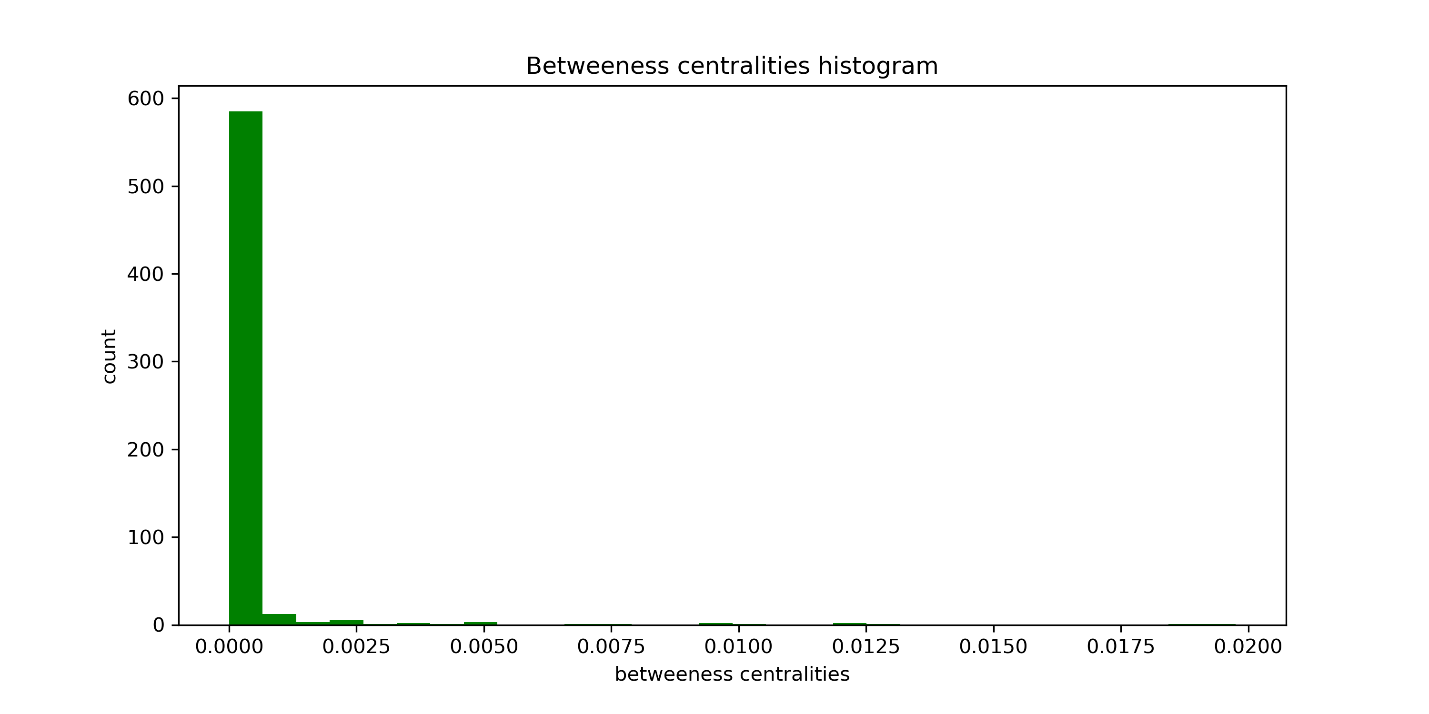
First, we ask the user to insert some conference ID:



*Output graph:*

**

**

**

*Computation time:*



1. given in input an author and an integer d, get the subgraph induced by the nodes that have hop distance at most equal to *d* with the input author: find the python code in the file *graph\_creation.py*.

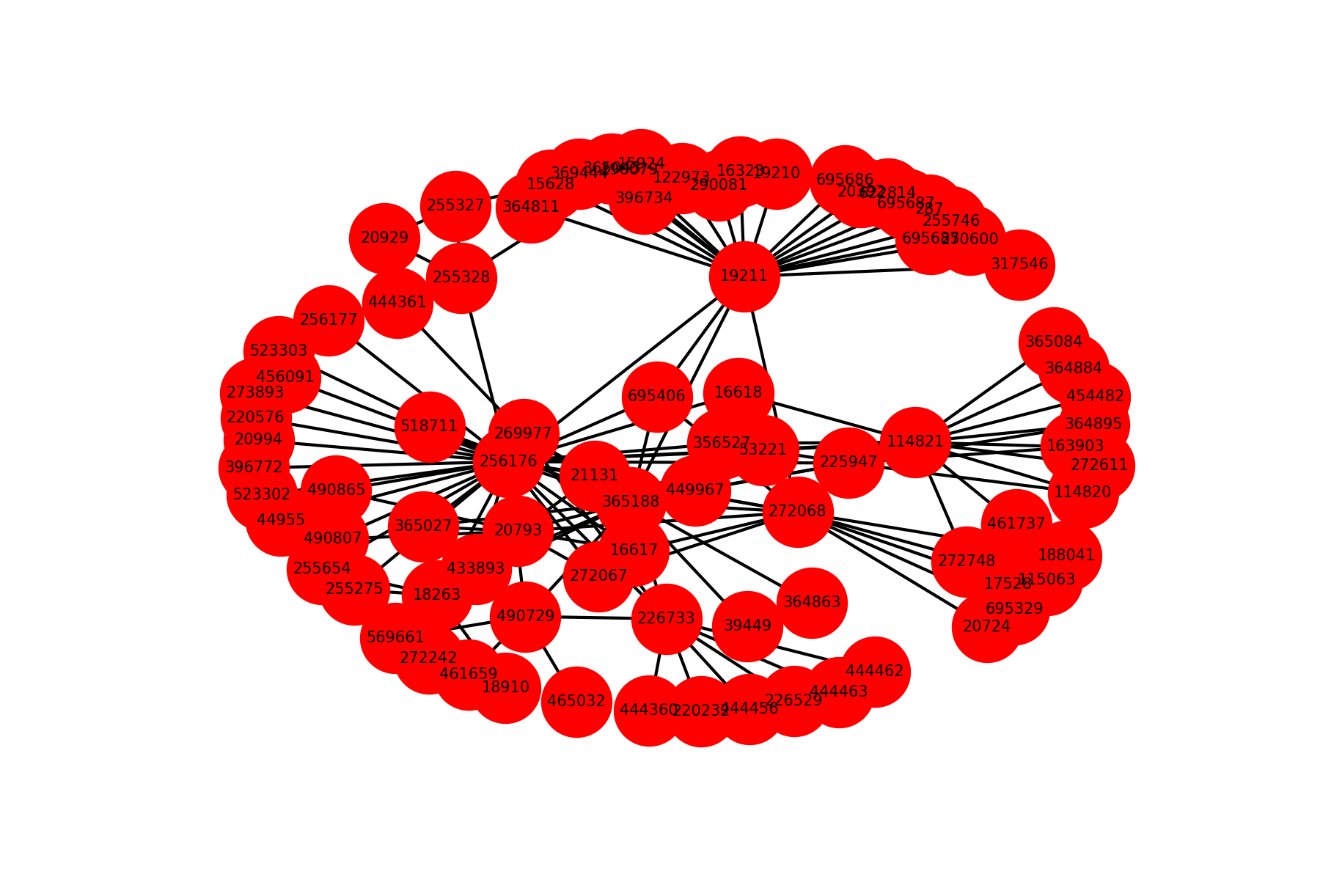
First, we ask the user to insert some author ID:



And then we also ask the user for an integer *d* which is going to the max for the hop-distance:



*Output graph:*





* **Third Step**: Erdős number

1. Here we compute the weight of the shortest path that connects some input author with Aris. We use as measure of distance the weight of the edges. Find the python code in the file *graph\_creation.py* and the shortest\_path function in the file *Libhw4.py*

First, we ask the user to insert some author ID:



And we call our shortest\_path function which returns:



After that, we call the *nx.dijkstra\_path\_length* function which returns the same result with a little difference in the computation time.

Computation time of our function:



Computation time of the nx function:



1. Here we compute for each node of the graph, its GroupNumber, defined as follow: *GroupNumber(v) = minu∈I{ShortestPath(v,u)}.*

First, we ask the user to insert some author ID: