**Network Alignment**

If you want to do either the **graph database and network modeling**, or the **biological network alignment** project, you need to know what a graph is and how to work with them, especially how to code with them. Your task is the following: you're given a text file representing a network. The first line of the file is N, the number of nodes. You will name the nodes from 0 through N-1. The remaining lines will have two integers per line, representing an edge. You don't know in advance how many edges there are, you just keep reading until you reach end of file. Your task is to compute and output the number of *connected componets* (you can ignore duplicate edges). If you are a grad student (or applying to grad school with me) then also compute a second number which is the number of STRONGLY CONNECTED COMPONENTS. Below I provide some sample inputs. I don't care what language you use. In addition, in your write up, include a histogram of the distribution of DEGREES of nodes; in the undirected case, just count all degrees; in the strongly connected (directed) case (if you are doing that one), provide histograms for both the in-degree and out-degree. (That means you'll have three histograms per graph.) The histogram shows how many nodes have degree zero, degree 1, etc., up to the max degree. **If you don't know any of these terms, look them up, don't ask me.** The data for this project is [here](http://www.ics.uci.edu/~wayne/research/students/graphs.zip). Analyze all the graphs in the zip file. **Extra work for those who already have a degree:** Read the [GRAAL paper](https://www.ics.uci.edu/~wayne/papers/GRAAL.pdf) and count the number of graphlets of size 3 in all the networks.

Specification:

you're given a text file representing a network. The first line of the file is N, the number of nodes. You will name the nodes from 0 through N-1. The remaining lines will have two integers per line, representing an edge. You don't know in advance how many edges there are, you just keep reading until you reach end of file.

Task :

1. Your task is to compute and output the number of *connected componets* (you can ignore duplicate edges).
2. compute a second number which is the number of STRONGLY CONNECTED COMPONENTS
3. include a histogram of the distribution of DEGREES of nodes, The histogram shows how many nodes have degree zero, degree 1, etc., up to the max degree
4. distribution of in-degree
5. distribution of out-degree