You are allowed to work in groups of two. Only one submission per pair is necessary, but please indicate who you are working together with. Submit your results as a zip-file to jpreusse@uni-koblenz.de until 04.06.2012, 11:59 pm.

1 Giant Connected Component (5 points)

1.1 Weakly/Strongly Connected Giant Component (3 points)

Write a function that computes the weakly or strongly connected giant component of a network \( G = (V, E) \) which may be either directed or undirected. The signature of your function should look like this:

\[
\text{function giantComponent = giantConnectedComponent(fileName, isDirected, weakOrStrong)}
\]

where \( isDirected \) is 0 if the network is undirected and 1 if it is directed. Further, \( weakOrStrong \) is 0 if the function should compute the weakly connected component and 1 if the strongly connected component is requested. The function should be called in the command line e.g. with

\[
v = \text{giantConnectedComponent('out.email-EuAll', 1, 1)};
\]

where \( v \in \{0, 1\}^{|V|} \) and

\[
v_i = \begin{cases} 
1, & \text{if } i \in \text{giant connected component}, \\
0, & \text{else}.
\end{cases}
\]

Thus, the sum of \( v \) should be the number of nodes contained in the strongly connected giant component

\[
\sum(v,1) = 34203.
\]

1.2 Big Data (2 points)

Inserting

\[
\text{time1 = cputime;}
\]

right at the second line of your function and inserting

\[
\text{ntime = cputime - time1; fprintf(1,'Time elapsed: %d\n',ntime);}\]

at the last line of your function, you output the computation time for your method. For above’s function call for the ‘out.email-EuAll’ data, your function should finish the computation within 3600 seconds.

Please write your elapsed time down, even if it is above the 3600 seconds. The best runtime\(^1\) will be rewarded with a bar of your favorite chocolate. My runtime is 2424 seconds, so this is the mark you have to beat for the sweets.

\(^1\)The runtime will be determined on my computer, so results are comparable.
2 Connected Components  (5 points)

2.1 Number of Connected Components and Size of Giant Component  (2 points)

Write a function that computes the size of the strongly and weakly connected giant component and the number of connected components of a network. The signature of your function should look like this:

function [sizeGiant, numComponents] =
connectedComponents(fileName, isDirected, weakOrStrong).

The enron is a directed network and contains parallel links of the following format:
senderID, receiverID, 1, timestamp

When calling your function, the result for the weakly connected case should be

[sizeGiant, numComponents] = connectedComponents('out.enron', 0, 0)
sizeGiant = 84384
numComponents = 1331

and for the strongly connected components

[sizeGiant, numComponents] = connectedComponents('out.enron', 1, 1)
sizeGiant = 9164
numComponents = 78058

2.2 Weak and Strong Ties  (3 points)

Now, you should use the previous function to implement the following method compareWeakStrong (2 points).

For the enron dataset you should implement the following

for i = 100:100:1000 do
% repeatedly remove i number of nodes from the original graph
% where i \in [100,1000] and stepsize = 100
    1. Remove the i weakest\(^2\) ties from the giant connected component of the original network. Compute
       size of giant connected component \(s_{\text{weak}}(i)\) and number of connected components \(c_{\text{weak}}(i)\).
    2. Remove the i strongest\(^3\) ties from the giant connected component of the original network. Compute
       size of giant connected component \(s_{\text{strong}}(i)\) and number of connected components \(c_{\text{strong}}(i)\).
endfor;

Before you look at the results, what do you expect to see when looking at \((s_{\text{weak}}(i), s_{\text{strong}}(i))\) and \((c_{\text{weak}}(i), c_{\text{strong}}(i))\) for different \(i\) and why?

Now, plot and compare \((s_{\text{weak}}(i), s_{\text{strong}}(i))\) and \((c_{\text{weak}}(i), c_{\text{strong}}(i))\) for different \(i\). Explain whether
the results match your expectations  (1 point).

---

\(^2\)Lowest edge weights, where weight of edge \((a, b)\) is defined by the number of times \(a\) mailed \(b\).

\(^3\)Largest edge weights, where weight of edge \((a, b)\) is defined by the number of times \(a\) mailed \(b\).