Computational Methods in Internet Economy
TUTORIAL 1

1. ADMINISTRATIVE INFORMATION

Announcement date: 14/11/2011
Submission date: **16/12/2011**
Contact person: Josif Grabocka, room C142 Spl
Submission: cmie@ismll.de

2. DATA DESCRIPTION

Data from SNAP: http://snap.stanford.edu/data/

i. wiki-Vote: http://snap.stanford.edu/data/wiki-Vote.html
   - what: Wikipedia who-votes-on-whom network
   - nodes: 7,115
   - edges: 103,689
   - giant component has 99.3% of nodes
   - average cluster coefficient: 0.21
   - diameter: 7
   - 90-percentile effective diameter: 3.8 (relates to average distance)
   - It is directed: **transform it to undirected** (for every one-way edge insert the symmetrical)

ii. Enron email network: http://snap.stanford.edu/data/email-Enron.html
   - what: Email communication network from Enron
   - nodes: 36692
   - edges: 367662
   - giant component has 91.8% of nodes
   - average cluster coefficient: 0.48
   - diameter: 12
   - 90-percentile effective diameter: 4.8 (relates to average distance)

iii. General Relativity and Quantum Cosmology collaboration network: http://snap.stanford.edu/data/ca-GrQc.html
   - what: Collaboration network of Arxiv General Relativity (co-authorships)
   - nodes: 5242
   - edges: 28980
   - giant component has 79.3% of nodes
   - average cluster coefficient: 0.53
   - diameter: 17
   - 90-percentile effective diameter: 7.6 (relates to average distance)

3. TASKS

a) Implement **a graph representation** in Java:

- Different possible representations: http://en.wikipedia.org/wiki/Graph_(data_structure)#Representations
- We are interested in large, sparse, undirected graphs
- Either develop your own representation (see for example: http://www.algolist.net/Data_structures/Graph/Internal_representation)
- or install and use Java Universal Network/Graph Framework (JUNG): http://jung.sourceforge.net/
    - JUNG Manual: http://sourceforge.net/apps/trac/jung/wiki/JUNGManual
    - JUNG Tutorial: http://www.grotto-networking.com/JUNG/JUNG2-Tutorial.pdf
- Note: If you cannot handle the size of the SNAP data in your computer (e.g., memory problems) then first consider a more efficient representation (e.g., for undirected graphs you can reduce the memory by 2 if you ignore the direction). If still you cannot handle the SNAP data, then proceed with sampling nodes and/or edges and also explain in your report how you did it.

b) **Visualize** the SNAP data:

- If you use JUNG for representing the graphs, you can use the inherent support for visualization: http://jung.sourceforge.net/doc/JUNGVisualizationGuide.html
- If you use your own graph representation in Java, please use a tool such as Pajek (http://pajek.imfm.si/doku.php) for visualizing the data.

c) Perform **basic network analysis** for each SNAP data:

- Report the size of the giant component (in percentage of total number of nodes).
- Report the diameter of the graph
- Report the average clustering coefficient (sum of CC's for each node divided by total number of nodes).
- Report the distribution of node degrees (create an equi-width histogram with your selected number of bins, see: http://en.wikipedia.org/wiki/Histogram).
- Report the distribution of Betweenness Values for the edges (create an equi-width histogram with your selected number of bins, see: http://en.wikipedia.org/wiki/Histogram).
    - If you use your own graph representation, you can use the algorithm explained in Chapter 3.6 of the book: http://www.cs.cornell.edu/home/kleinber/networks-book/
    - If you use JUNG representation, you can alternatively consider using existing features, such as the BFSDistanceLabeler (see: http://jung.sourceforge.net/doc/api/index.html).

4. REPORT

Write a brief report avoiding source code inside the report. The source code should be send as a separate zip file along your report to the submission mail cmie@ismll.de .