# TSKS11 Hands-On Session 1

Fall 2019

## Preparation for the TSKS11 labs

- Familiarize yourself with Gephi; see tutorials here,

  `https://gephi.org/users`.

- Also have a look at the Stanford Network Analysis Project (SNAP) project,

  `http://snap.stanford.edu`.

- To prepare for the hands-on sessions in the computer lab,

  1. From the command line, run

     ```
     module add prog/gephi/0.9.2
     module initadd prog/gephi/0.9.2
     module add prog/matlab
     module initadd prog/matlab
     ```

     It is sufficient to do this once, at the start of the course.

  2. Go to your home directory and run

     ```
     rm -rf TSKS11-sandbox
     cp -r /courses/tsks11/ht2019/TSKS11-sandbox .
     ```

     to copy the sandbox directory into your home directory. This is the directory you will work on in all the labs. If you want to start over the labs, for some reason, just redo these two steps.

  3. Some large data files required for the labs are stored in the read-only directory `/courses/tsks11/ht2019/data_and_fcns`. You may either (recommended) just read these files from `/courses/tsks11/ht2019/data_and_fcns`, or alternatively, copy the files into the appropriate subdirectory in your home directory.

- To start Gephi in the computer labs, run

  ```
  gephi &
  ```

  from the command line.

## Troubleshooting

- Occasional problems to re-start the Gephi and Mozilla Firefox software after a crash in the computer labs have occurred in the past. In case of problems, run in the home directory:

  ```
  rm -rf .gephi
  rm -rf .mozilla
  ```

- Before calling on support, also run

  ```
  rm -rf .local/lib
  rm -rf .cache
  ```

## Running the labs via remote login from a computer at home

From a Linux command line, run

```
ssh -X LiUID@ssh.edu.liu.se
```

to enter the same environment as in the computer labs. This requires a very fast and reliable Internet connection!

## Running the labs locally on a computer at home

It is perfectly possible to solve all lab problems on a personal laptop or home PC running a modern Linux distribution. A certain degree of familiarity with computers is required and path names will have to be modified manually.

The recommended approach is to use mount the LiU filesystem with

```
sshfs LiUID@ssh.edu.liu.se:/ ~/EDU
```

(the directory EDU must first be created)

Alternatively, use `scp` to copy the entire directory /courses/TSKS11 to your computer.

The software required for the labs has been developed on Linux (Ubuntu), but has not been tested on Windows or Mac (although in principle it should work).

# Task 1: Visualization of DBLP Network in Gephi

In this task we will perform some analysis of the DBLP network. This is a non-anonymous (with real names) co-authorship network in computer science, publicly available at `dblp.org`, consisting of some 3 million publications and more than a million authors. The goal is to run some community detection methods, evaluate some metrics and discuss the result, against the perspective and theory learned in class.

- Work in the subdirectory `TSKS11-sandbox/sesson1/task1`.
- Select two or three seeds in the DBLP program, `session1_task1`, by entering the seed names into the lines `IDvec.Add`. Then run (in the terminal)

  ```
  make session1_task1
  ```

  For the selection of seeds, any names may be used. Some suggestions are in the file `suggested-seeds.txt`.
- Run the DBLP-lab program (`session1_task1`), and open the file `X.NET` with Gephi. Make sure to import the network as undirected. Select node color according to modularity, node size according to degree centrality and run a layout algorithm to obtain a figure similar to that labelled "DBLP" in the class notes. Does the network have a clear community structure?
- Open the file X-centrality.csv into Matlab/Octave or a spreadsheet program (e.g., LibreOffice or MS-Excel). Plot the betweenness centrality (in logarithmic scale) against the clustering coefficient. Is there any correlation? Why?

# Task 2: Introduction to SNAP

- Work in the subdirectory `TSKS11-sandbox/session1/task2`.
- Run the Python script `session1_task2.py`.
- Go through the script, and map each function call to the SNAP documentation and explain what each line does.
- Open the .NET files, and visualize and analyze them with Gephi. Specifically, detect communities, and generate a figure similar to that in Task 1. Look at some metrics defined in class, e.g., the distribution of the length of shortest paths between nodes, clustering coefficient.

# Examination

- Collaboration on this homework in small groups is encouraged, but each student should individually demonstrate understanding of all tasks.
- Individual oral examination takes place in class (computer lab).