In the lab-classes this week we revise graphs by using BFS, DFS and disjoint-sets functionality.

**Update the environment:** Update the repository via

```
git pull
```

or, if you didn’t create it before, create a (new) clone by

```
git clone git://github.com/OKullmann/CS-242-Algorithms.git
```

Change to the subdirectory for week 7:

```
cd ~/CS-242-Algorithms/201112/Week07/
```

**Basic setup:** Two graphs \( G_1, G_2 \) are given via the files \( G1, G2 \):

1. Both graphs have 30 vertices.
2. The vertices are numbers \( 0, \ldots, 29 \).
3. Each line gives the two endpoints of one edge.

The task is to use from previous weeks the programs `BreadthFirstSearch`, `BreadthFirstSearch_forest`, `DepthFirstSearch_forest` and `LinkedLists` to compute basic parameters of these graphs:

- the number of connected components
- whether the graph is connected or not
- whether the graph is acyclic (a forest) or not
- diameter and radius.

All four programs read from standard input. Recall that via e.g. `cat G1 | ...` you can use the content of \( G1 \) instead.

You might want to look again at the labsheets for weeks 4 and 6: you find them on the course homepage [http://cs.swan.ac.uk/~csoliver/Algorithms201112/index.html](http://cs.swan.ac.uk/~csoliver/Algorithms201112/index.html).

## 1 The number of edges

We have \(|V(G_1)| = |V(G_2)| = 30\). You need to determine the number of edges:

\[
\begin{align*}
|E(G_1)| & = ~? \\
|E(G_2)| & = ~?
\end{align*}
\]

You can use the command-line tool `wc` (use `man wc` for information) for that.
2 The number of connected components

For both graphs you need to determine the number of connected components, and this in three ways:

1. Use BFS (the forest-version is most convenient).
2. Use DFS.
3. Use disjoint sets (you need to stare a bit at the output).

If there is more than one component, then you also need to determine the sizes of the components.

Drawing the graphs is likely not really useful, however for each graph you best produce a factsheet, where you write down the various pieces of information you gathered.

Now, given the number of connected components, it is easy to answer:

- Is $G_1$ connected?
- Is $G_2$ connected?

3 Cycles

Determine:

- Is $G_1$ acyclic?
- Is $G_2$ acyclic?

Use two methods:

1. As discussed in the lecture, disjoint sets give an answer (stare at the full output).
2. Also from the number of connected components and the number of edges the answer can be deduced:

   (a) Consider a graph $G$.
   (b) Let $c$ denote the number of connected components, let $m := |E(G)|$ denote the number of edges, and let $n := |V(G)|$ denote the number of vertices.
   (c) If $c = 0$, then $m = 0$ and the graph is acyclic.
   (d) If $c = 1$, then the graph is acyclic if and only if $m = n - 1$.
   (e) Can you guess the generalisation: $G$ is acyclic (for arbitrary $c$) if and only if $m = n - ?$.

4 Radius and diameter

Look up at Wikipedia the notions “radius” and “diameter”. They only apply to connected graphs. Determine these parameters, by using BFS (with various start vertices). Note that BreadthFirstSearch works only with directed graphs, so you need to transform the undirected graph into a directed graph (appropriately!). A little awk-program print_fields.awk is provided, which prints both fields of a line, called e.g. by awk -f print_fields.awk G1; adapting this program (note that several edges can be put on one line) you can create the desired directed graph, which you might store via “>” into a file.