Assignment 1

Available Since: April 23, 2015   Due Date: April 30, 2015, 12:00 a.m.
You are permitted and encouraged to work in groups of two.

Exercise 1: 5 Points

Show that the following statements are equivalent for an undirected graph $G = (V, E)$ with $n$ vertices and $m$ edges.

(a) $G$ is connected and does not contain any cycles.

(b) $G$ is connected and $m = n - 1$.

(c) There exists exactly one path between any two vertices of $G$.

Exercise 2: 5 Points

A graph is *chordal* if it has no induced cycles of length greater than three. A graph is *transitive orientable* if its edges can be oriented such that if there is an edge oriented from $u$ to $v$ and an edge oriented from $v$ to $w$ then there is also an edge oriented from $u$ to $w$. The complement $\overline{G}$ of a graph $G$ is the graph in which two distinct vertices are adjacent if and only if they are not adjacent in $G$.

In the following let $G$ be a graph that has an interval representation.

(a) Prove that $G$ is chordal.

(b) Prove that $\overline{G}$ is transitive orientable.
Exercise 3: 3 Points

The following exercise should make you familiar with the framework we are going to work in. You do not have to hand it in on paper, but please commit your changes to the SVN.

- Download and setup Eclipse together with an svn plugin (e.g., subclipse).
- Checkout the following project into your workspace:
  https://svn.uni-konstanz.de/algo/gd_s15
- Copy GraphDrawer and resource from the material folder into your own group folder (lastname1_lastname2) (which will be accessible from Friday).
- Make yourself familiar with the RandomLayout class and make it an option in the toolbar when starting the GraphDrawer class. This will make it possible to layout a graph, for example after generating it using BinaryTreeGenerator or RandomGraphModel).

Exercise 4: 7 Points

The following exercise should make you even more familiar with the framework we are going to work in. You do not have to hand it in on paper, but please commit your solution to the SVN.

- Write your own layout class that implements a circular layout, in which all vertices are evenly spread over the contour of a circle.
- Use class name lastname1_lastname2.a01.CircularLayout.

For more information on the yFiles library used in this project, check docs.yworks.com/yfiles/doc/developers-guide/.