Assignment 2

Available Since: 02 May 2013  Due Date: 16 May 2013, 08:00 a.m.
You are permitted and encouraged to work in groups of two.

Exercise 1: 3 Points

Let $T$ be a rooted tree with $n$ vertices and the property that any vertex is either a leaf or has at least two children.

(a) Give a tight lower bound for the number of leaves of $T$.

(b) Give a tight upper bound for the height of $T$.

Exercise 2: 3 Punkte

(a) Design and explain an efficient algorithm which determines the corresponding tree from given pre- and inorder sequences (the nodes are labeled uniquely). Show the corresponding asymptotic runtime.

(b) Determine the tree with your developed algorithm (a) from the following sequences and draw it.

preorder: A C B D E G H F I
inorder: B C E D A F H I G

Exercise 3: 4 Points

An unrooted tree $T$ can be drawn by choosing an arbitrary vertex of $T$ as the root and applying a tree layout to the thus rooted tree. A possible criterion for choosing the root is to minimize the height of the resulting rooted tree.

Consider the following definitions for a graph $G = (V, E)$. The distance $d_G(v, w)$ between two vertices $v$ and $w$ is the length of a shortest path in $G$ between $v$ and $w$. The eccentricity of a vertex $v$ is $e_G(v) := \max_{w \in V} d_G(w, v)$ and the radius of $G$ is $r_G := \min_{v \in V} e(v)$. The center of $G$ is the set $Z_G := \{v \in V : e_G(v) = r_G\}$. Note that the height of a tree is minimized if and only if its root is chosen from its center.
(a) Let $T = (V, E)$ be a tree, $L \subseteq V$ the set of leaves of $T$, and $T'$ the subgraph of $T$ induced by $V \setminus L$. Show that $Z_T = Z_{T'}$.

(b) Give an efficient algorithm that computes the center of a tree and examine its runtime? Can this problem be solved in linear time?

(c) How many elements does the center of a tree have at most?

**Exercise 4:**

Implement the Reingold-Tilford algorithm presented in the lecture. Ensure linear-time complexity of your implementation!

- use a node from the center, as described in Exercise 3, as the root of the tree (use `graph.firstNode()` only if you cannot compute the center)
- use class name `lastname1.lastname2.u02.ReingoldTilford`
- consider using `y.base.NodeMap` to store contour lists and x-offsets

Test the efficiency of your implementation by generating random trees, node number between 1 000 and 25 000 (step width 200). Measure your running times (not those of the generator) and create a plot (i.e. using gnuplot, R, Excel) that documents your results. Submit the plot together with the theoretical exercises and the implementation via svn.

**General Hints:**

- download and setup Eclipse together with an svn plugin (i.e. subclipse)
- checkout the following project into your workspace: https://svn.uni-konstanz.de/algo/lehre/gd13
- copy the classes and content from `material.groupZero` to your own group folder, e.g., `lastname1.lastname2` (make sure that your copied `GraphDrawer` class is working and that it doesn’t use classes from `groupZero`.
- make yourself familiar with the `CircularLayout` class which can be used as an option in the toolbar when starting the `GraphDrawer` class.
- consider using `BinaryTreeGenerator`.
- feel free to add further icons or extend the project with further functionality