Models of Computation, an Algorithmic Perspective

Assignment 9

Tue 14.12.2010

This assignment is due on January 5/7 in your respective tutorial groups. You are allowed (even encouraged) to discuss these problems with your fellow classmates. All submitted work, however, must be written individually without consulting someone else’s solutions or any other source like the web.

Exercise 1 Consider a tree $T$ (on vertex set $V = \{1, ..., n\}$) in the Euler tree representation. For any vertex $r \in V$, we let $T_r$ denote the rooted tree obtained by rooting $T$ at $r$. For vertices $u$ and $v$, least common ancestor of $u$ and $v$ ($LCA(u, v)$) is defined as the node furthest from the root that is an ancestor of both.

We want to answer the following type of queries: Given three vertices $r, u, v \in V$, what is $LCA(u, v)$ in $T_r$?

a) Design an algorithm that finds the answer to such a query with depth $O(\log n)$ and work $O(n \log n)$. [Hint: The following seemingly unrelated problem might be relevant: Given an array $A[1..n]$ and two indices $i_1$ and $i_2$, what is $\min_{i_1 \leq j \leq i_2} A[j]$?]

b) Assume now that we want to answer many such queries for the same root $r$. Design a data structure that represents $T_r$ in such a way that the least common ancestor of two vertices $u, v$ can be found with constant depth and work. The construction of the data structure should take only depth $O(\log n)$ and work $O(n \log n)$. [Hint: Precompute the answer for an appropriately chosen set of $O(n \log n)$ queries by dynamic programming!]

Exercise 2 Let $G = (V, E)$ be a planar graph with edge lengths given by $\ell : E \to \mathbb{N}$. Show that for any $0 < \varepsilon < 1/2$, we can preprocess the instance with depth $O(\sqrt{n} \log^2 n + n^{2\varepsilon} \log n)$ and work $O(n^{3(1-\varepsilon)} \log n)$ such that the distance between any two vertices can be computed in sequential time $O(n^{2\varepsilon} \log n)$. 