TEL AVIV UNIVERSITY
Department of Computer Science
0368.4281 – Advanced topics in algorithms
Spring Semester, 2012/2013

Homework 2, April 7, 2013

Due on April 23. Please put a hardcopy in my mailbox and keep a copy of the homework!

1. We say that a flow f is acyclic if there is no cycle with positive flow along it. Given a flow f from s to t in a directed graph $G = (V, E)$, show how to get from f an acyclic flow f' such that $|f| = |f'|$ (try to get an algorithm that runs in $O(m \log n)$ time.).

2. Assume that we modify the push-relabel algorithm so that we never push-relabel from active nodes of distance label greater than or equal to n.
a) Prove that when we stop, the value of the flow into the sink is the value of the maximum flow.
b) Give an efficient algorithm to convert the preflow when the modified algorithm stops, to a flow. What is the complexity of your algorithm ?
c) Suppose that we modify the algorithm so it keeps track of the number of active nodes of distance label i for each i, $1 \leq i < n$. When we relabel the last vertex of distance label i we also relabel all vertices that currently have distance label greater than i and smaller than n, to n. Prove the correctness of this modified implementation and analyze its running time.

3. In this question, you'll make precise some classical facts about finger search trees. Please answer rigorously. (Assume that all items in the trees below have distinct keys.) Recall that a 2-4 tree is a B-tree in which each node has either 2,3, or 4 children, and 1,2, or 3 items, respectively. Insertion is done by adding the item to the appropriate leaf, splitting this leaf if it has 4 items, adding an item to its parent, and keep splitting (the parent) until the parent has a legal number of items. Deletion is done by first replacing the item to be deleted with its predecessor or successor so that it is in a leaf. Then we delete the item from the leaf, if the leaf remains with only one item we either steal an element from one of its siblings if a sibling has > 2 elements, or merge it with its sibling. As a result of such merge the parent looses a child and we continue recursively to fix the parent if it remains with one child.

In a finger 2-4 tree we access the tree from both ends instead of from the root. We search in parallel going up from both ends until we identify the subtree containing the key and continue searching the target as in a regular 2-4 tree in that subtree.

a) Describe precisely an implementation of concatenate, and split for 2-4 finger search trees with fingers at both ends. Recall that a concatenate takes two trees T_1 and T_2 such that all the elements in T_1 are smaller than all the elements in T_2 and returns one tree T containing all the elements that are either in T_1 or in T_2. Split takes a tree T and a key x and returns two trees T_1, containing all elements smaller than x in T, and T_2 containing all elements larger or equal to x in T.

b) Prove that your implementation has the following properties:
1) Concatenate and split take $O(\log n)$ worst-case time, where n is the total number of elements

in the tree/trees participating in the operation.

2) Consider a sequence of inserts, concatenate, and split operations starting from an empty tree. Concatenate of a tree T_1 with n_1 items to a tree T_2 with n_2 items takes $O(\log(\min\{n_1, n_2\}))$ amortized time. Split at x_i or insert of x_i, takes $O(\log(\min\{d_i, |T_i| - d_i\}))$ amortized time, if x_i is the d_i^{th} smallest element in T_i.

c) Improve the bound on catenation in (b2) to $O(1)$ amortized time, the bound for the other operations remains the same.

4. (a) Given a minimum cost circulation in a graph $G = (V, E)$, show how to compute node potentials $\pi(v)$, $v \in V$, such that all reduced costs with respect to π ($c^\pi(e)$) of residual edges are non-negative. What is the running time of your algorithm ?

(b) Let $\pi(v)$, $v \in V$, be potentials such that there exists a feasible circulation f whose residual edges have non-negative reduced costs with respect to π. Given π show how to find a minimum cost circulation whose residual edges have non-negative reduced cost with respect to π. What is the running time of your algorithm ?

Try to find efficient and simple algorithms.