CSC265 Fall 2011 Homework Assignment 4

due Wednesday, December 7, 2011, 10:15am

1. A white-black heap is a set of binomial trees that satisfies the following properties:

   - The roots of the binomial trees in the set all have different degrees.
   - Each node in the binomial trees is either white or black. The root of every binomial tree is white.
   - If a white node is not the root of a binomial tree, its key is greater than or equal to the key of its parent.
   - The key of a black node is less than the key of its parent.
   - The black nodes all have different degrees.
   - The degree of a black node is 1 less than the degree of its parent, i.e. the black node is the first child of its parent.

   Recall that a node in a binomial heap has degree k if and only if the subtree rooted at that node is a binomial tree with 2^k nodes.

   We say that a set of binomial trees is a k-white-black heap if it satisfies all the properties of a white-black heap except that it has at most two black nodes of degree k and at most one of its black nodes of degree k has a parent of degree greater than $k+1$. Thus for every natural number k, every white-black heap is a k-white-black heap.

   (a) Suppose you have a set H of binomial trees that satisfies all the properties of a white-black heap except that it has two black nodes of degree k both of which have parents of degree $k+1$. Give a transformation that converts H into a set of binomial trees on the same set of nodes that is a k'-white-black heap for some $k' > k$.

   (b) Suppose you have a set H of binomial trees that satisfies all the properties of a white-black heap except that it has exactly one black node of degree k and its parent has degree greater than $k+1$. Suppose that the sibling of degree $k+1$ of this black node is white. Give a transformation that converts H into a set of binomial trees on the same set of nodes that is a k'-white-black heap for some $k' > k$.

   (c) Suppose you have a set H of binomial trees that satisfies all the properties of a white-black heap except that it has exactly one black node of degree k and its parent has degree greater than $k+1$. Suppose that the sibling of degree $k+1$ of this black node is black. Give a transformation that converts H into a set of binomial trees on the same set of nodes that is a k'-white-black heap for some $k' > k$.

   (d) Suppose you have a set H of binomial trees that satisfies all the properties of a white-black heap except that it has exactly two black nodes of degree k, one of whose parents has degree greater than $k+1$. Give a transformation that converts H into a set of binomial trees on the same set of nodes that is a k'-white-black heap for some $k' > k$.

   (e) Explain how to perform DECREASE-KEY in a white-black heap so that its amortized cost is $O(1)$. Use the number of black nodes in the white-black heap as your potential function or, equivalently, maintain the invariant that there is one token on each black node.

2. A node in a directed graph is a *sinkhole* if every other node has an edge to it, but it has no edge to any node.

   (a) Prove that every directed graph has at most one sinkhole.

   (b) Suppose you are given the adjacency matrix representation of a directed graph with n nodes. Give an algorithm that determines whether the graph has a sinkhole and, if so, returns it. Your algorithm must run in $O(n)$ time. Prove that your algorithm is correct and runs in the desired time.

   (c) Suppose you are given the edge list representation of a directed graph with n nodes and m edges. Give an algorithm that determines whether the graph has a sinkhole and, if so, returns it. Your algorithm must run in $O(n + m)$ time. Prove that your algorithm is correct and runs in the desired time.

   (d) Prove that, for any algorithm which determines whether a graph, represented by its edge list, has a sinkhole, there is a graph with n nodes on which the algorithm takes $\Omega(n^2)$ steps.

3. (a) Suppose that, during the BFS of an undirected graph $G = (V, E)$, node a is first visited before node b, which is first visited before node c. Prove that if $\{a, c\} \in E$, but $\{a, b\} \notin E$, then there exists a neighbour d of b which is visited before node a.

   (b) Suppose that, during the DFS of an undirected graph $G = (V, E)$, node a is first visited before node b, which is first visited before node c. Complete the following sentence and prove it is correct: If $\{a, c\} \in E$, but $\{a, b\} \notin E$, then there exists a neighbour d of b which is visited . . .

4. Two trees are edge-disjoint if there is no edge appearing in both of them.

   (a) Let G be a graph with 2 edge-disjoint spanning trees. What is the least number of vertices, n', that G can have? Give an example of a graph on n' vertices which has 2 edge-disjoint spanning trees.

   (b) For general $n \geq n'$, describe a graph on n vertices and a weight function such that the graph has two edge-disjoint minimum spanning trees, but not three minimum spanning trees that are pairwise edge-disjoint. Explain why your construction is correct.

   (c) Prove that if a graph has fewer than $2k$ vertices, then it cannot have k spanning trees such that every pair of them is edge-disjoint, when $k \geq 3$.

   (d) Prove that if there is a graph on n vertices that has k, but not $k + 1$, pairwise edge-disjoint minimum spanning trees, then there is also a graph on $n + 1$ vertices which has k, but not $k + 1$, pairwise edge-disjoint minimum spanning trees.