Purdue University | Gopal/Hambrusch
CS 381 | Spring 2008

**Assignment 6**

Due: Friday, March 21, 2008 (before class)

1) (30 points) Consider red-black trees supporting, in $O(\log n)$ time, the operations find(x), insert($x$) and delete($x$).

(i) Describe how to augment a red-black tree to support, in $O(\log n)$ time, the query Count($x, y$) which determines the number of elements $z$ currently in the tree with $x \leq z \leq y$. Include a discussion on how additional fields are maintained under insertions and deletions.

(ii) Describe how to augment a red-black tree to support, in $O(\log n)$ time, the queries increase($x, y, q$) and decrease($x, y, q$). In increase($x, y, q$), the value of every entry $w$ in the tree with $x \leq w \leq y$ is increased by $q$ (the decrease operation does the corresponding decrease). One cannot explicitly update the entries as this may take $O(n)$ time. Describe how all operations can be executed in $O(\log n)$ time.

2) (20 points) Assume that we use disjoint rooted trees to implement the MAKE-SET, UNION, and FIND-SET operations on disjoint sets. We perform a sequence of $m$ MAKE-SET, UNION, and FIND-SET operations, $n$ of which are MAKE-SET operations. Assume that **all** UNION operations occur **before** any of the FIND-SET operations.

Note that a UNION operation takes as input the representatives of two sets and combines them into one set in constant time. UNION operations use both union-by-rank (or union-by-size) and path compression heuristics.

(i) What is the maximum height of any tree just after all the UNION operations are finished. What is the worst case cost for one FIND-SET operation after the UNION operations are finished? Justify your answers.

(ii) Use the accounting method to show that the sequence of $m$ operations takes $O(m)$ time. (Hint: To pay for the FIND-SET operations, charge a UNION operation extra and use these quantities later to pay for the find operations.)