

**Homework Assignment #6**  
**Due: July 7 at 10:00 p.m.**

1. The textbook has two routines that fix a single violation of the red-black tree properties introduced by insertions and deletions in a red-black tree (RBT):

- $\text{RB-INSERT-FIXUP}(T, z)$  fixes a tree  $T$  where  $z$  and its parent are both red, and
- $\text{RB-DELETE-FIXUP}(T, x)$  fixes a tree  $T$  that has double-black node  $x$ .

Suppose we have two RBTs  $T_1$  and  $T_2$  and a key  $k$  such that every key in  $T_1$  is less than  $k$  and every key in  $T_2$  is greater than  $k$ . We wish to design a JOIN operation that constructs a new RBT  $T'$  that contains  $k$  and all the keys in  $T_1$  and  $T_2$ .

Assume that each node in our RBTs has an additional field that stores the black height of that node. (We shall see in class soon how this field can be kept up-to-date without affecting the running time of operations.)

- [2] (a) Describe how to do  $\text{JOIN}(T_1, k, T_2)$  if the roots of  $T_1$  and  $T_2$  have the same black height.
- [4] (b) Suppose  $T_1$ 's root has smaller black height than  $T_2$ 's root. Then  $\text{JOIN}(T_1, k, T_2)$  will modify  $T_2$  to build  $T'$ . The JOIN replaces a node  $v$  in  $T_2$  with a new node  $v'$  whose left child is the root of  $T_1$  and whose right child is  $v$ . How would you choose  $v$ ? What would the key and colour of  $v'$  be so that this change introduces at most one violation of RBT properties?
- [2] (c) Describe the violation that could be introduced in part (b). How would you fix it?
- [2] (d) State the worst-case running time of JOIN if  $T_1$  has height  $h_1$  and  $T_2$  has height  $h_2$ . Briefly justify your answer.