

**Algorithm Design**  
***M. T. Goodrich and R. Tamassia***  
John Wiley & Sons  
**Solution of Exercise C-3.11**

For each node of the tree, maintain the size of the corresponding subtree, defined as the number of internal nodes in that subtree. While performing the search operation in both the insertion and deletion, the subtree sizes can be either incremented or decremented. During the rebalancing, care must be taken to update the subtree sizes of the three nodes involved (labeled  $a$ ,  $b$ , and  $c$  by the restructure algorithm).

To calculate the number of nodes in a range  $(k_1, k_2)$ , search for both  $k_1$  and  $k_2$ , and let  $P_1$  and  $P_2$  be the associated search paths. Call  $v$  the last node common to the two paths. Traverse path  $P_1$  from  $v$  to  $k_1$ . For each internal node  $w \neq v$  encountered, if the right child of  $w$  is in not in  $P_1$ , add one plus the size of the subtree of the child to the current sum. Similarly, traverse path  $P_2$  from  $v$  to  $k_2$ . For each internal node  $w \neq v$  encountered, if the left child of  $w$  is in not in  $P_2$ , add one plus the size of the subtree of the left to the current sum. Finally, add one to the current sum (for the key stored at node  $v$ ).