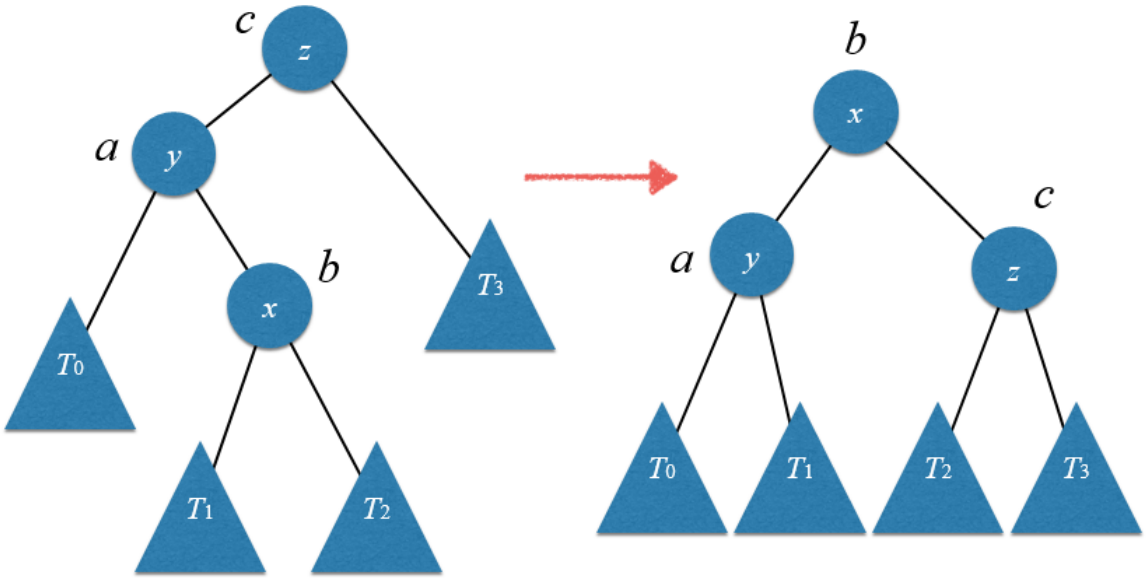
CSC 226 FALL 2016
ALGORITHMS AND DATA STRUCTURES II
ASSIGNMENT 2 - WRITTEN
UNIVERSITY OF VICTORIA

1. Draw the 2-3 trees that result when you insert the keys Y L P M X H C R A E S in that order into an initially empty tree. There should be 11 trees in all. Use the final tree to construct the corresponding red-black tree.

2. Consider Case 2 (taken from slide 50 in Lecture-5-AVLTrees.pdf and shown below). On the left is the subtree S' rooted at z that results from inserting key k into an AVL tree T. On the right is the subtree S^* rooted at node x which is the result of restructuring at node z in S'. Prove that the height of S^* is equal to the height of S, the subtree rooted at z prior to inserting key k.



3. An inversion in a sequence is an out-of-order pair; i.e., $i < j$ but $a_i > a_j$. Inversions are discussed briefly in the book on page 252. For example, the sequence $(5,3,2,1,4)$ has 7 inversions. What is the minimum number of inversions of a permutation of $1,2,\ldots,n$? What is the maximum number of inversions of a permutation of $1,2,\ldots,n$?

4. Explain carefully how to use red-black trees to compute the number of inversions in a permutation in time $O(n \log n)$. Effectively, you may need to modify the code for Algorithm 3.4 on page 439. Explain in detail any changes that you would make to method put().

5. In the programming portion (full specification to come) you are going to write a program that computes the percentage of red nodes in a given red-black tree. You will test your program by running 100 trials of the experiment of inserting N random integer keys into an initially empty tree, for $N = 10^4, 10^5$, and 10^6. Here you will report your findings and formulate a hypothesis about the percentage of red nodes in any red-black tree based on your results.