# Tutorial 4.

1. In the ordinary BST, let us have an additional variable to be stored at the node, which is the height of the subtree rooted there. Thus every node x has the following attributes: left, right, value, height. Write pseudo-code for insert and delete in BST with the updation of height. Be careful with delete.
2. This is to ensure that you understand the analysis of the "random permutation". In Quicksort, we had said that if [a(1),a(2),...,a(n)] is a random permutation, then the probability that a(1)=i is precisely 1/n. Now, for a permutation as above, let b=a(a(1)). In other words, if a=[2,3,1,4,5] then b=3, but if a=[5,4,3,2,1] then b=1 and so on. If I were to select a random permutation, then what is the probability p(i) that b=i. Warning: p(i) is not the same for every i.
3. Consider the structure below. Is the tree structure AVL? If we wish to store the set 1, . . . , 22, label each node with the correct number.
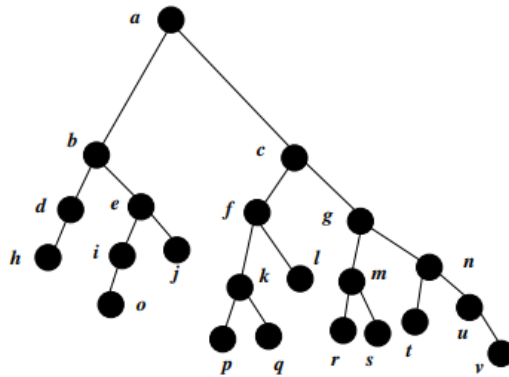
4. Now add 23 to the set and then delete 1. Also do the same in the reverse order. Are the answers the same? When will the answers be the same?
5. Why is it that in insert, a single or double rotation is sufficient to balance the tree, but not in deletion?