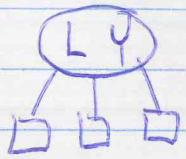
CSC226 ASSIGNMENT 2 - SOLUTIONS

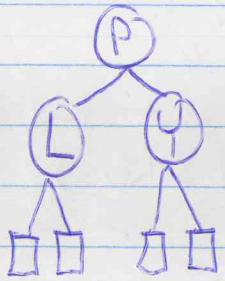
1.

1. insert(Y):

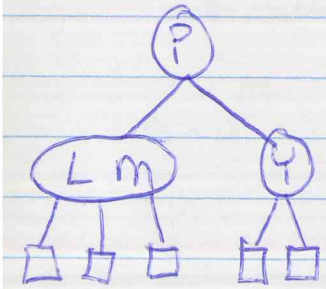
2. insert(L):



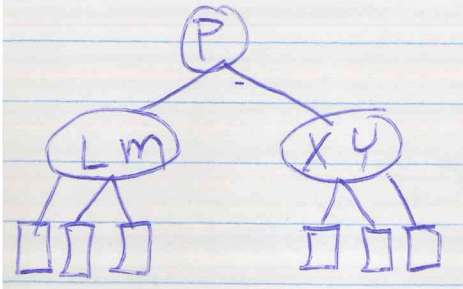
3. insert(P):



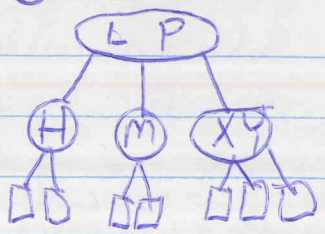
4. insert(M):



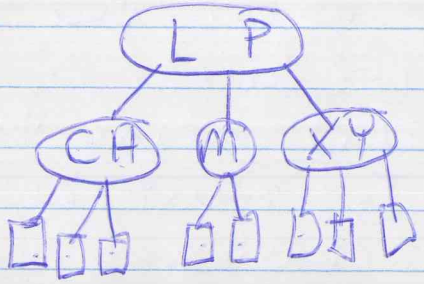
5. insert(X):



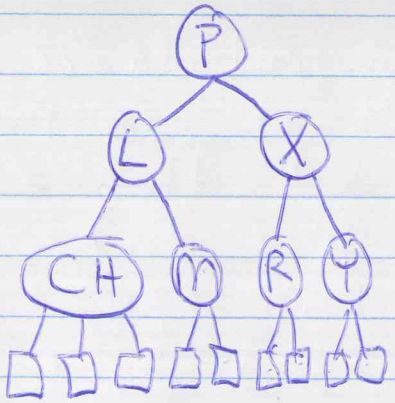
6. insert(H):



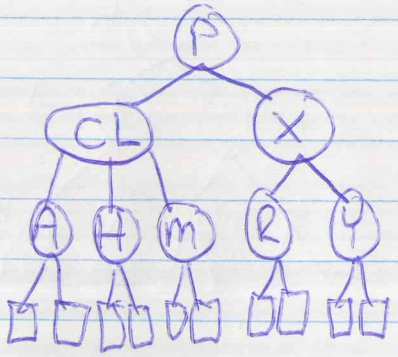
7. insert(C):



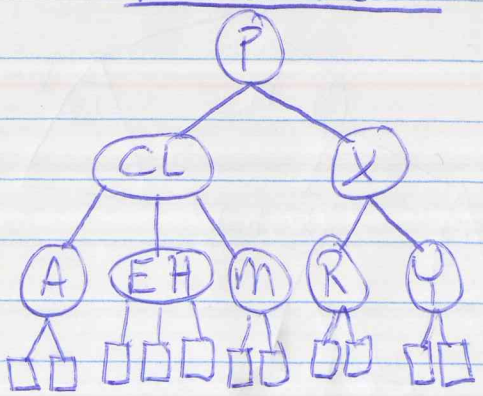
8. insert(R):



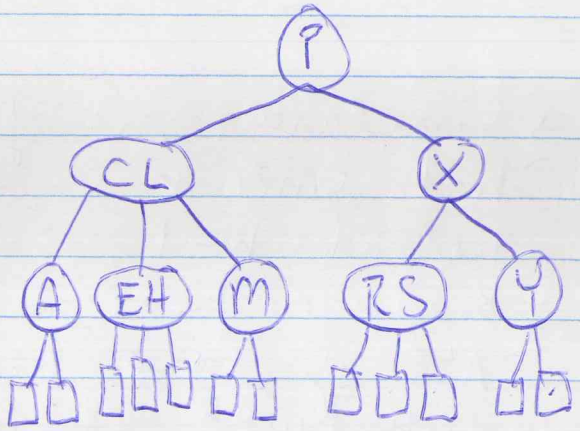
9. insert(A):



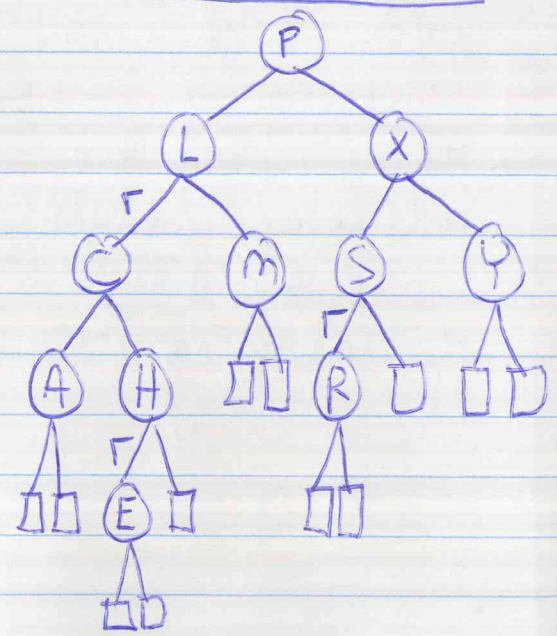
10. insert(E):



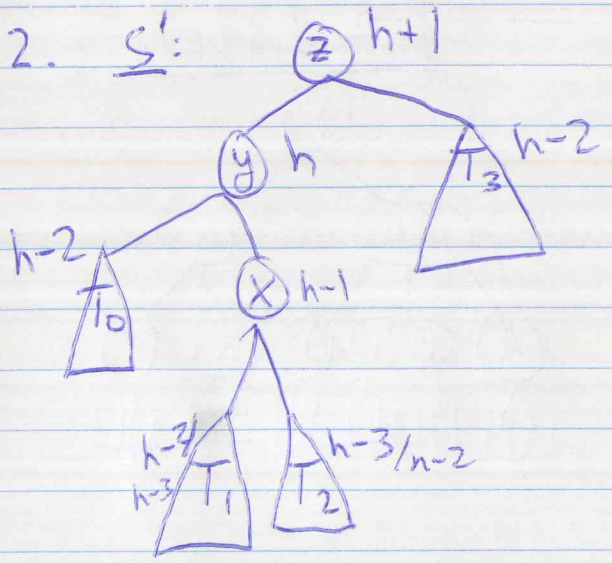
11. insert(S):



Red-Black Tree:



2. S':



→ Let height of S (before insertion) be h. Since there is an imbalance at z, the height of S' is h+1

→ One of y or T_3 must be at height h. Since y was the "highest" child it's height is h.

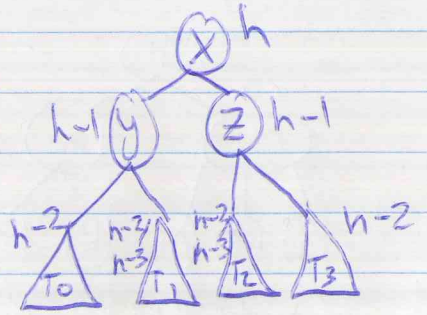
→ That means the height of T_3 must be h-2, since z has an imbalance

→ Similarly, one of T_0 or x must be at height h-1. Since x is the higher child, it has height h-1.

→ Because there is an imbalance at z but not y, the height of T_0 must be h-2 (otherwise, if it's h-1, then z could not have had height h in S.)

→ One of T_1 & T_2 has height h-2 and the other is height h-3. Also, otherwise height of S would have been h+1.

→ Thus, after the restructure we have the following heights, since the heights of T_0, to T_3 do not change.



3. The minimum number of inversions occurs in the permutation $(1, 2, \ldots, n)$ which is 0.

The maximum number of inversions occurs in the permutation $(n, \underbrace{n-1}_{1}, \ldots, \underbrace{2}_{n-2}, \underbrace{1}_{n-1})$ which is

$$\sum_{i=1}^{n-1} i \cdot 1 + 2 + \cdots + n-1 = \frac{(n-1)n}{2} = \frac{n^2-n}{2}$$

4. If you insert the elements from a permutation of $1, 2, \ldots, n$ in order into a red-black tree, then inversions occur when elements in the tree are bigger than the element you are inserting. So, every time you do a put you need to count how many nodes in the tree contain larger keys than the current element. We do this by keeping track of the size of the subtree (nodes) at the right child of the node h, you are comparing when the current key k is less than $h.key$. That is, when $k < h.key$ let

$$count \mathrel{+}= 1 + size(h.right)$$

5. The actual ~~~~ Percentage of red nodes is a red-black tree is between 25 and 25.5% somewhere. Their experimental values should be consistent with this. They may have a smaller range of values based on the number of input tests they do but their range should be in ~~25% to 25.5%~~ 25% to 25.5%