

Name:

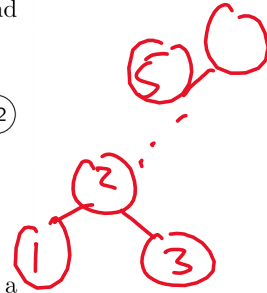
SID:

Please complete this worksheet during your lab, and turn it in to your TA by the end of your section. You are encouraged to work with your neighbors collaboratively.

Section Number:

- 01
- 02
- 03
- 04
- 05
- 06
- 07
- 08
- 09
- 10
- 11
- 12

1 BTrees

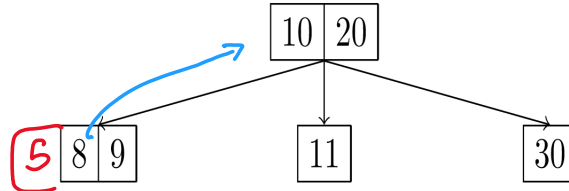


1.1 Given a Queue which returns items in sorted order, give the runtime of creating a *balanced* binary search tree, where N is the number of items initially in the Queue.

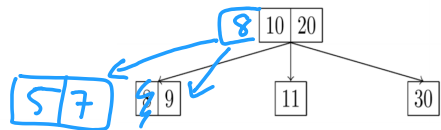
- $O(\log N)$
- $O(N)$
- $O(N \log N)$
- $O(N^2)$

1 2 3 (5) ...

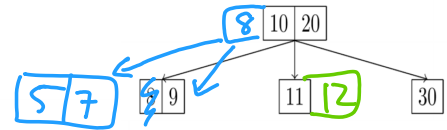
1.2 Given the following initial 2-3-4 tree, draw the result of performing each operation.



1. Insert 5 into this tree.



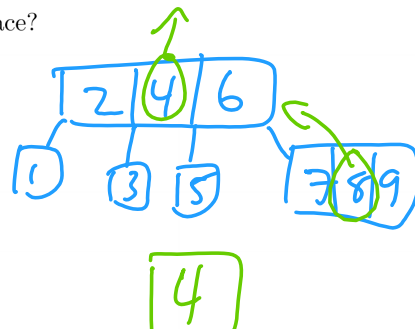
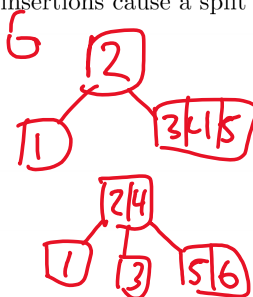
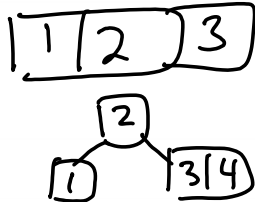
2. Insert 7 into the resulting tree.



3. Insert 12 into the resulting tree.

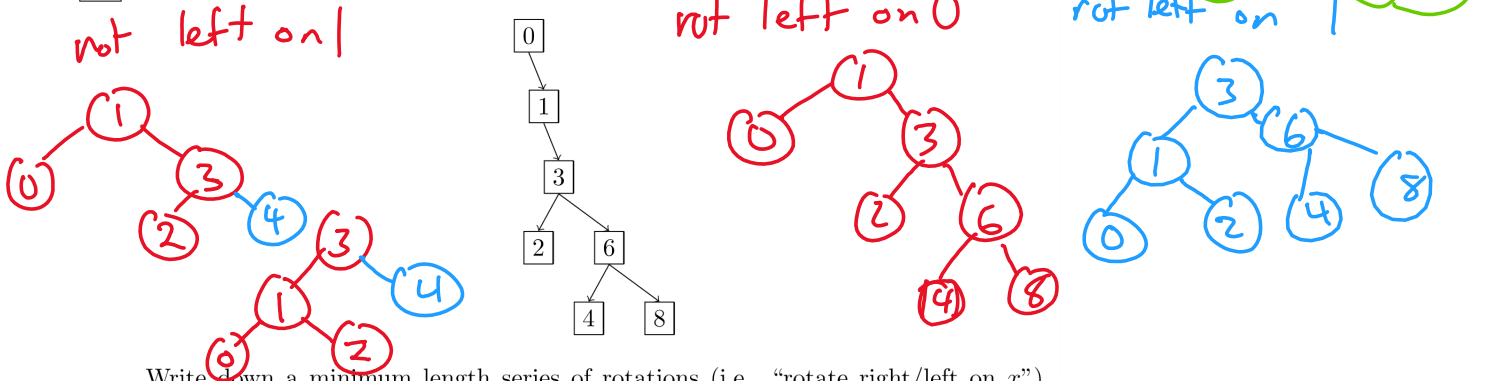
1.3 Suppose the keys 1, 2, 3, 4, 5, 6, 7, 8, 9, and 10 are inserted sequentially into an initially empty 2-3-4 tree. Which insertions cause a split to take place?

4, 6, 8, 10



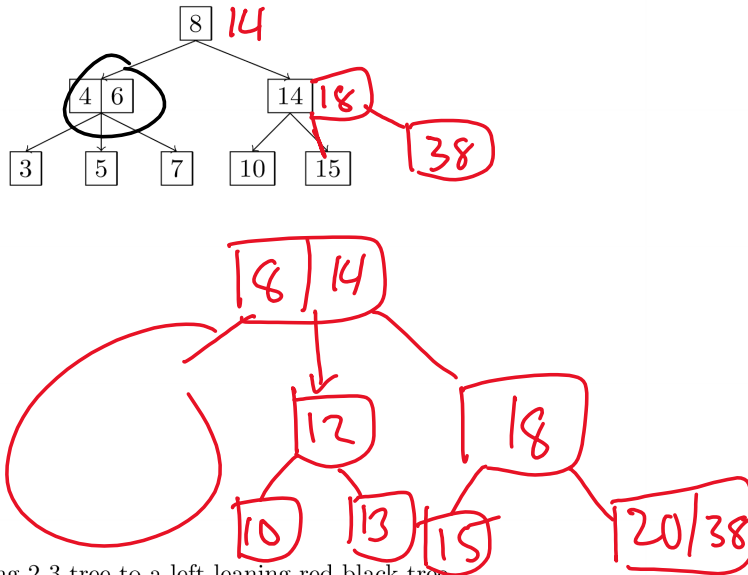
2 LLRBs

2.1 Let's say we are given an extremely unbalanced binary search tree:



Write down a minimum length series of rotations (i.e. "rotate right/left on x ") that will make tree balanced and have height of 2.

2.2 Draw what the following 2-3 tree would look like after inserting 18, 38, 12, 13, and 20.



2.3 Now, convert the resulting 2-3 tree to a left-leaning red-black tree.

