Homework - Tree Data Structures

April 14, 2018

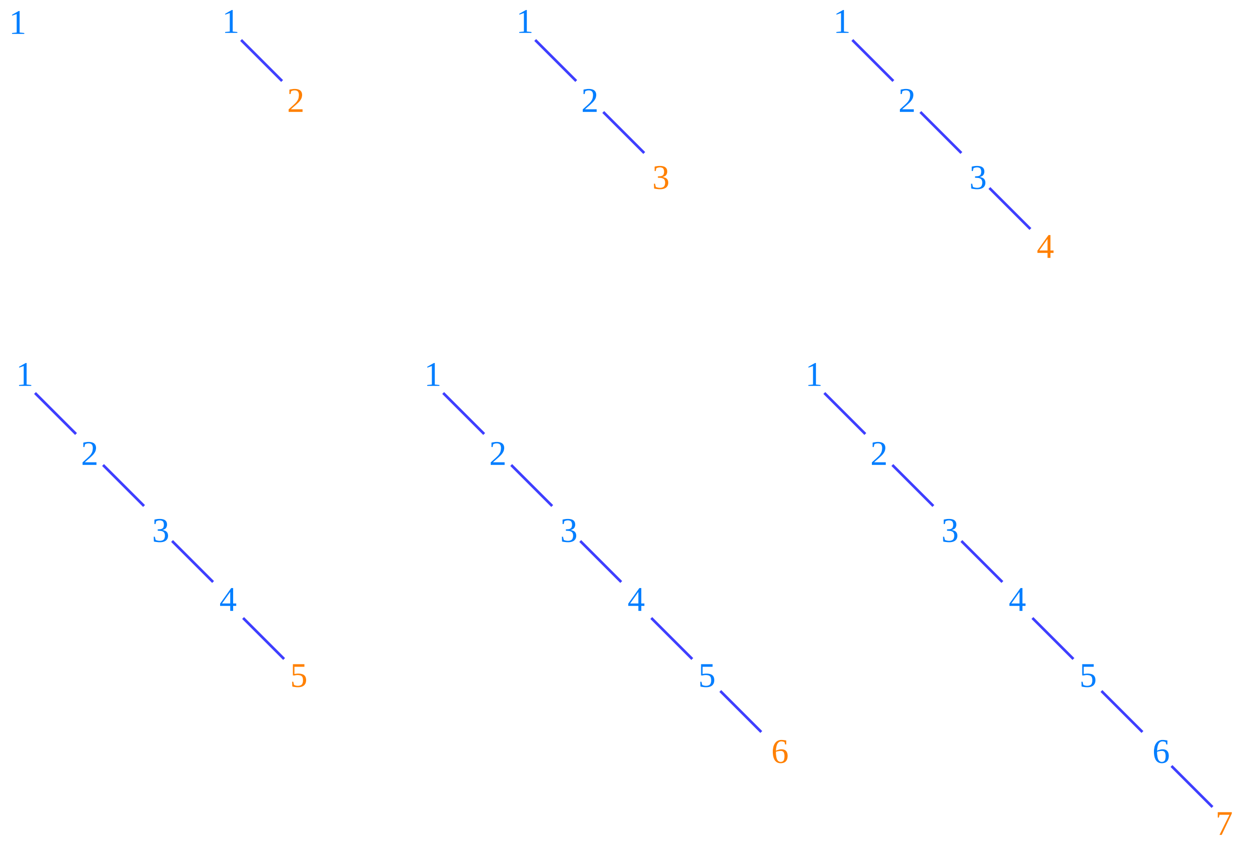
The completed assignments can be submitted either on blackboard or handed to me in class on Wednesday 18th April, 2018.

Each of the questions carry 20 points.

The last question involves searching and sorting. Whereas we have not gone through those techniques in class, I would like you to research and read about them and answer the questions.

1 Binary Search Tree

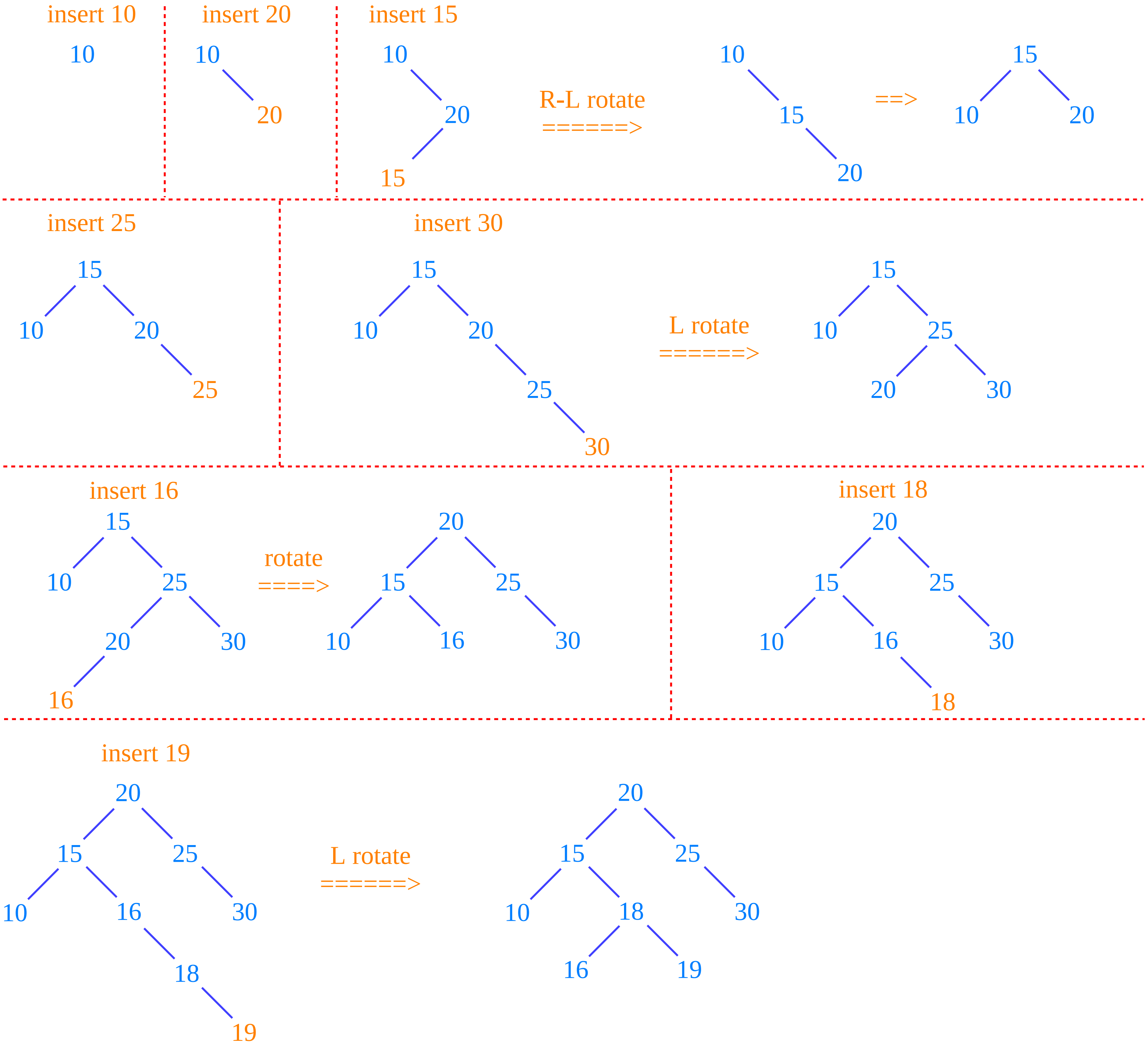
Draw the binary search tree obtained when the keys 1, 2, 3, 4, 5, 6, 7 are inserted in the given order into an initially empty tree. What is the problem of the tree you get? Why is it a problem? How could you modify the insertion algorithm to solve this problem. Justify your answer.



2 AVL Tree

i. Insert the following sequence of elements into an AVL tree, starting with an empty tree: 10, 20, 15, 25, 30, 16, 18, 19.

ii. Delete 30 in the AVL tree that you got.



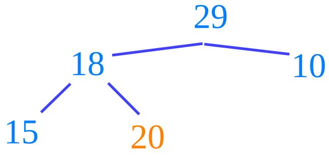
3 Max -Heap

Consider the array A = {29, 18, 10, 15, 20, 9, 5, 13, 2, 4, 15}

i. Does A satisfy the max-heap property? If not, fix it by swapping elements

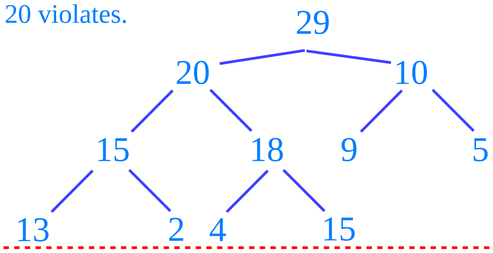
ii. Using array A (possibly corrected), illustrate the execution of the heap-extract-max algorithm, which extracts the max element and then rearranges the array to satisfy the max-heap property. For each iteration or recursion of the algorithm, write the content of the array A.

i.



A does not satisfy the max-heap property because:
element 20 violates.

swap
18 <=>20



ii.

4 Min-Heap

Consider the array: A = {4, 33, 6, 90, 33, 32, 31, 91, 90, 89, 50, 33 }

i. Is A a min-heap? Justify your answer by briefly explaining the min-heap property.

ii. If A is a min-heap, then extract the minimum value and then rearrange the array with the min-heapify procedure. In doing that, show the array at every iteration of min-heapify. If A is not a min-heap, then rearrange it to satisfy the min-heap property

5 Search & Sorting techniques

i. Define and contrast the following search and sorting techniques: Insertion Sort, Heap Sort, Merge Sort, Quick Sort, Sequential Search, Binary Search

ii. For each of the mentioned search and sorting techniques, specify the worst-case and average-case Big-Oh complexity, assuming an input array of size N (You can draw a table with a column for worst-case and average-case):