

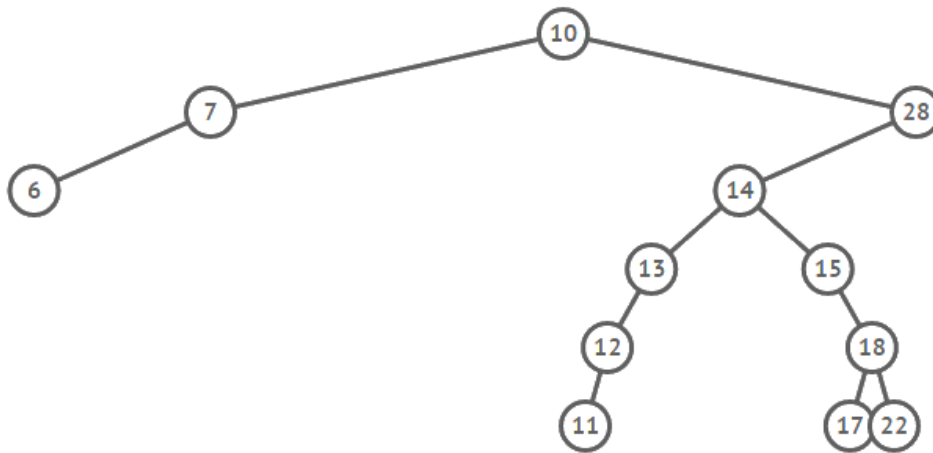
There are five questions in this exam. Answer all questions.

Q1. [4] Illustrate trees/heap at each step for the following:

(a) Build a Priority Queue using a Binary Heap for the following data  
 10, 13, 22, 16, 8, 7, 9, 6, 11, 12, 15.

(b) Remove the three smallest elements from the Priority Queue. Highlight which nodes you decide to sink or swim and re-draw the heap at each step.

Q2. [4] Observe the following Binary Search Tree. Convert this tree to a AVL tree. Re-draw the tree every time you decide to rotate it. Highlight the rotation type (case) for each rotation.



Q3. [3] Write a recursive method `public boolean checkAVL(AVLNode A)` to check if a tree with root at **A** is AVL. Assume all heights are included in the tree.

Q4. [2] Show how the array would be sorted using MergeSort (top-down). Trace all sort and merge operations.

10	13	22	16	8	7	9	6	11	12	13	16
----	----	----	----	---	---	---	---	----	----	----	----

Q5. [2] What is the run time (Big Oh notation) for the following operations in given data structures.

- Insertion of a key in Priority Queue implemented using a heap \_\_\_\_\_
- Removal for the last element in a Stack \_\_\_\_\_
- Worst case scenario for Removal of a key in Hash table implemented using Probing \_\_\_\_\_
- Number of swap operations using Insertion Sort in a sorted (descending order) array \_\_\_\_\_
- Average case scenario for removal of a node from a BST \_\_\_\_\_
- Average number of comparisons in selection sort \_\_\_\_\_
- Average number of element swaps in selection sort \_\_\_\_\_
- List the Order of all sub-arrays for bottom-up merge sort with array size  $n=11$ .