**Lab 6: HeapSort – AVL Trees**

**Question 1:** Draw a AVL tree for given sequence of numbers: 2, 9, 7, 6, 5, 8, 11, 4, 3

**Question 2**: Show the time efficiency classes of the priority queue's for following operations and data types.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Unsorted Array | AVL Tree | Heap |
| find maximum element |  |  |  |
| Delete maximum element |  |  |  |
| Add new value |  |  |  |

**Question 3:** Assume that we want to implement the priority-based job scheduler. Which is the best data structure to implement the priority-based operation system process scheduler (LinkedList, Queue, BST, etc.). What is the insertion performance of this data structure? Explain.

For example: I use an AVL Tree for process scheduler and, implement AVL Tree as an unsorted array. Insertion performance of the AVL Tree with an unsorted array is O(n).

**Question 4:** Why is the time complexity of Heap Sort O(n\*Log n)? The time complexity of which operation is O(logn) and which one is O(n)?

**Question 5:** Given a sequence of numbers: 2, 9, 7, 6, 5, 8

1. Draw a binary max-heap (in a tree form) by inserting the above numbers reading them from left to right
2. Show a tree that can be the result after the call to deleteMax() on the above heap