Problem Session

1. Consider an array of size '5'. What is the worst case number of comparisons by (note: in asymptotic sense, it is $O(1)$, however, we focus here on 'exact number')
   - Insertion Sort
   - Merge Sort
   - Is it possible to sort in 7 comparisons.
2. Write an algorithm to find both the smallest (min) and largest (max) elements in a set of n-elements.
   - In $2n - 3$ comparisons
   - In $\frac{7n}{4} - 2$ comparisons
   - In $\frac{5n}{3} - 2$ comparisons
   - In $\frac{3n}{2} - 2$ comparisons
3. Write an algorithm to find the second largest element (second max).
   - In $2n - 3$ comparisons
   - In less than $2n - 3$ comparisons
4. You are given a very large array (theoretically, infinite sized array) and you do not know the size of the array. The array is filled with integers in increasing order till some position (i.e., array is sorted till some position) and after that it is filled with (remaining all locations) very large integer (infinity). You are asked to search an element x in this array. How efficiently can you perform search to locate x.
5. You have 70 gold coins that are all expected to be of the same weight (say 1 gram), however you know that one coin is fake (whose weight is 0.9 gram). You are given a balanced scale which has two plates and one can place any number of coins on each side at one time, and the scale will tell you if both sides weigh the same or which side is lighter. Present an algorithm to identify the faulty coin in least number of weighings. What would be the number of weighings if there are 27 coins.
6. You are given 9 boxes ($B_1, \ldots, B_9$) all containing gold coins (say 1 gram each). Although, all B_i's are expected to be identical, one box is faulty (i.e., all coins in that weighs 1.1 gram). What would be your approach (algorithm) to identify the faulty one. This time, you are given an electronic scale with one plate and the scale will tell you the weights of the coins placed. For example, if you place 4 coins from the faulty box, scale will show up 4.4.
7. An integer array $A[1..n]$ contains all integers from 0 to n except one. Write an algorithm to identify the missing integer.
8. Present an efficient algorithm to merge k sorted lists into one sorted list. n_i represents the size of the i^{th} list and n denotes the total number of elements.
9. In a max-heap, where can we find second maximum, k^{th} maximum, minimum.
10. Describe a $\theta(n \log n)$ algorithm that, given a set S of n-integers and another integer x, determine whether or not there exist two elements in S whose sum is exactly x.
11. Present an efficient algorithm to sort n integers in the range 0 to k, k is a fixed integer.
12. A sorting algorithm is stable, if numbers with the same value appear in the output array as they do in the input array. i.e., Order of appearance is preserved between input and output. Which of the following algorithms are stable; bubble, insertion, merge, quick, heap.
13. What modification would you do to quick sort so that it runs in $\theta(n \log n)$ in worst case.