Master of Computer Applications
Department of Computer Science & Engineering,
Motilal Nehru National Institute of Technology Allahabad.

Mid Semester Examination 2016-17 (odd)

Subject: Analysis of Algorithms (CA 3304) **Duration:** 1.5 HRS **Max. Marks:** 20

Note: Be specific and **to the point** in your answers. Make assumptions wherever necessary and quote it. All questions are compulsory. Neat answers, without cuttings will be appreciated. It is advisable to design the solution in rough before writing the final answers. Answer the questions serially.

```
RANDOMIZED-SELECT (A, p, r, i)
1  if p == r
       return A[p]
2  q = RANDOMIZED-PARTITION (A, p, r)
3  k = q-p+1 // total elements on the left of the partition +1
4  if i == k   // the pivot value is the answer
       return A[q]
5  else if i < k
       return RANDOMIZED-SELECT (A, p, q-1, i)
6  else
       return RANDOMIZED-SELECT (A, q+1, r, i-k)
```

Q1. Answer following questions in context of algorithm RANDOMIZED-SELECT: **[3x2=6]**

a. If $T(n)$ be the running time of the algorithm, write the recurrence relation for $T(n)$ and solve it using Master's Method.

b. Provide an iterative version of RANDOMIZED-SELECT.

Q2. COUNTING SORT algorithm assumes that each of the n input elements is an integer in the range 0 to k, for some integer k. When $k=O(n)$, the sort runs in $\Theta(n)$ time. Change COUNTING SORT algorithm for the numbers lying in the range p to q where $(q-p)$ is still $O(n)$. Also analyze the new complexity. **[6]**

Q3. *K*-WAY MERGE algorithms or MULTIWAY MERGES are a specific type of sequence merge algorithms that specialize in taking in multiple sorted lists and merging them into a single sorted list. These merge algorithms take in a number of sorted lists greater than two. Design a fast algorithm for merging k sorted lists with its complexity analysis. **[6]**

Q4. Justify that the complexity of *BUILD-MAX-HEAP* over an array of size n is $O(n)$. **[2]**