MCAC 201 : Design and Analysis of Algorithms
Minor
March 2025

Problem 5-1 (Sorting/Searching) 15 points

(a) (5 points) Given a sorted array $A[1, \ldots, n]$ such that elements may not be distinct. Modify binary search such that for a given key, it returns the index of the first occurrence of key if it exists, else returns -1.

1. $first - occurrence = -1$
2. Compute the middle index "mid"
3. If $A[mid] = key$ then
   _____
   recurse on _____
4. If $A[mid] < key$ then recurse on _____
5. else recurse on _____
6. return $first - occurrence$

Also, give one line justification.

(b) (5 points) For each of the following inputs, state whether it is the worst case for Quick sort or not, where the last element is always chosen as the pivot? In either case, count the number of comparisons. Also, if it is not the worst case, modify it by changing the positions of at most two values - that means you can only swap one pair of numbers.

i. 6, 10, 15, 3, 27, 2, 32, 1
ii. 11, 6, 18, 8, 32, 7, 37, 15

(c) (5 points) The enrollment number of students in University of Delhi has the following format: first 4 digits for year of admission, next three letters (all small) program code followed by 3 digits of the serial number. For example: 2024mca058, 2022mcs025, 2019mph129 where mca, mcs and mph are codes for the MCA, M.Sc. (CS) and M.Sc.(Physics) programs respectively. Give a linear time algorithm to sort the students on their enrollment numbers. Argue that the running time of your approach is $O(n)$, where n is the number of students.

Problem 5-2 (Graphs) 12 points

(a) (4 points) Let $G = (V, E)$ be a directed graph, and let $s \in V$ be a chosen vertex. Give an $O(|V| + |E|)$ time algorithm to find the length of the shortest path from every vertex $v \in V$ to s. We define the length of a path as the number of edges on that path.

(b) (4 points) Consider a rooted tree T. By completing the following procedure, compute, for each node v in tree T, in the field $v.leaves$, the number of leaf nodes in v's subtree.

procedure LeafCount(v)
???
for each child w of v **do**
LeafCount(w)
???
end for
If ??? then ???
end procedure

Initial Call: LeafCount(T)

(c) (4 points) Given an undirected graph G with vertices already labeled with the DFS-start and DFS-finish times. Give a linear time algorithm to sort the vertices on their DFS-finish times.

(Hint: Think more - write answer in one or two lines)