University of Calcutta
Mode of Examination: Online
M.Sc. Semester – I Examination, 2020
Subject: Computer Science
Paper Code & Name: CSM 103 (Advances in Data Structure)
Full Marks: 70
Date: 20.03.2021 **Time and Duration: 12.00 PM – 3:00 PM (3:00 Hours)**

Please note the following instructions carefully:
Promise not to commit any academic dishonesty.
Marks will be deducted if the same/similar answers are found in different answer-scripts.
Candidates are required to answer in their own words as far as applicable.
The figures in the margin indicate full marks.
Each page of the answer scripts should have your University Roll # on the right-top corner.

The name of the scanned copy of the answer script will be of the following format:
CSM-103-ADS-University-Roll-Number.pdf
(Example: CSM-103-ADS-C91-CSC-201001.pdf)
The subject of the mail should be the file name only.
The scanned answer-script is to be sent to cucse2020@gmail.com
The report should have the top page (Page #1) as an index page; mention page number(s) against each question number's answer.
The answer-script may not be accepted after the scheduled time.

Answer Question #1, Question #2, and any four from the remaining.

1. Answer ***any five*** questions: [2 × 5]

(a) Differentiate little-oh from Big-Oh.

(b) Assuming an array $X[50, 60]$ with four words per memory cell and the base address as 150, compute the address of $X[40, 6]$ in row-major order.

(c) By how many comparisons one can radix-sort n integers, where each integer has d digits, and each digit belongs to the set $\{1, 2, \ldots, k\}$?

(d) How many leaf nodes can be there in a 5-way tree that has eight internal nodes and each internal node has precisely five children?

(e) What is the running time of BFS that works on a graph represented by an adjacency matrix?

(f) Explain with reasons the searching method used to find a word in a dictionary.

(g) What do you mean by *mergeable heaps*? Give examples.

(h) What is a disjoint-set forest? Give an example of union operation using the disjoint-set forest.

2. Answer ***any five*** questions: [4 × 5]

(a) How a k-ary tree, $k \geq 3$, could be represented as a binary tree? Give a suitable example to support your answer, and characterize the binary tree you obtained. [2 + 2]

(b) Define tree edge, forward edge, back edge, and cross edge related to DFS on a graph. Show that DFS on an undirected graph does not result in any cross edge. [2 + 2]

(c) Is *QuickSort* truly the fastest sorting algorithm? Justify your answer in support of its name. Write the PARTITION algorithm of *QuickSort*. [2 + 2]

(d) Devise a mechanism to identify the isolated vertices, pendant vertices, vertices with degrees two and three of a graph of at least 14 vertices. What is the complexity of your mechanism? [3 + 1]

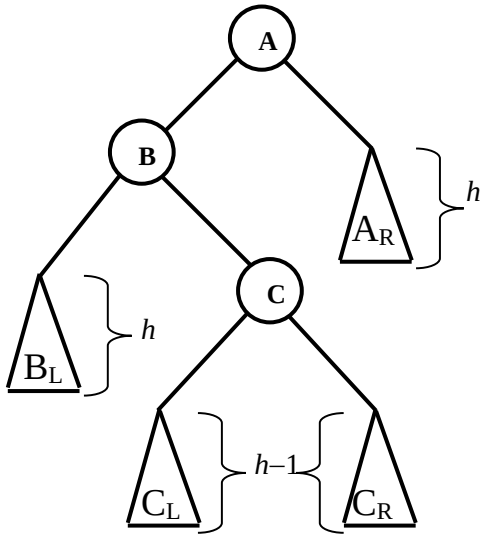
(e) Draw the 2-3 tree that comprises of a linearly ordered set, S = {{1.5, 2.5, 3.5}, {5.5, 6.5, 7.5}, {8.5, 9.5}}. Obtain the 2-3 tree S', after inserting 4.5 into S. Also, obtain the 2-3 tree after deleting 8.5 from S'. [1 + 1½ + 1½]

(f) "The preorder and postorder sequences of a binary tree are not sufficient to uniquely construct the tree." —Prove or disprove with justification and support. [4]

(g) Show the hash table that results when the letters in "COMPUTER SCIENCE" are stored in the given order using the linear probe collision resolution method. Assume a hash table of size 19 and use the hash function, H(k) = k mod 19 for the *alphabet's k-th letter*. [4]

(h) What do you mean by amortized analysis of algorithms? How is it differentiated from worst-case analysis and average-case analysis? State the role of the credit balance function in computing the amortized cost of some algorithm. [1 + 2 + 1]

3. (a) Define the AVL tree, and state its importance. Suppose we have an instance of a binary tree as shown below.



Now suppose a key is inserted in C_L, and its height becomes h. Show that the tree is no longer an AVL tree. Briefly state the steps to be taken to make it an AVL tree, and compute the total cost necessary to do it. [2 + (1 + 2 + 1)]

(b) Build a max-heap using the array $A = \langle 6, 4, 18, 11, 85, 20, 7, 23, 10 \rangle$, and accordingly sort the given sequence. [4]

4. (a) Write down the algorithm(s) for evaluating an infix expression and show how the algorithm(s) work(s) for a given expression of at least twelve characters (comprising operands, operators, and parentheses). [6]

   (b) Distinguish between binomial heap and Fibonacci heap along with their properties and memory representations. [4]

5. (a) Construct a Red-Black tree inserting the following keys into an initially empty tree. [6]

   41, 38, 33, 14, 19, 8, 27, 5

   (b) Consider a Red-Black tree formed by inserting n nodes. Argue that if $n > 1$, the tree has at least one red node. [4]

OR

Mathematically define sorting. If Π is a permutation of at least twelve distinct numbers, then clearly show the steps how *GraphSort* may work to sort the numbers in Π in descending order. Define stable sorting and justify if *GraphSort* is a stable sorting algorithm. [2 + 5 + 3]

6. (a) Construct a B-tree of order 3 by inserting into an initially empty B-tree the following keys in the order given.
   L, P, B, M, O, V, W, S, F, D, I [5]

   (b) Delete from the final B-tree obtained after 6(a) above, the keys M, W, and S one after another and compute the B-trees. [3]

   (c) Are B-trees of order 2 full binary trees? —Justify. [2]

7. (a) Construct a *lexicographic search tree* for the following numbers: [3]
   124, 1246, 1249, 299, 3792, 3799, 4724, 4781,480, 555, 78311, 78312, 8, 99

   (b) Prove that a binomial tree B_k has exactly ${}^kC_i$ nodes at depth i for $0 \leq i \leq k$. [3]

   (c) Consider the problem of incrementing n-digit binary integers and compute the amortized cost of the problem in terms of n, if applicable. [4]

8. (a) Write a short note on *threaded binary tree*. [4]

   (b) On an empty binomial heap, carry out the following sequence of operations:
   insert (28), insert (15), insert (18), insert (21), insert (24), insert (12), insert (9), insert (13), insert (5), insert (17), insert (41). Then delete the minimum element present in a heap. Clearly show the steps to be executed and draw the resulting structure of the heap after each operation. [6]