(*Please write your Exam Roll No.*) *Exam Roll No* ……

Bharati Vidyapeeth's

Institute of Computer Applications and Management

A-4, Paschim Vihar, New Delhi-63

SECOND SEMESTER [MCA] Internal Examination, February 2020

Paper Code: MCA–201	**Subject: Data and File Structures**

Time: 2 Hours **Maximum Marks: 45**

Note: Attempt THREE questions in all. Question No. 1 is compulsory, and attempt one question from each unit.

1. Answer all the following questions briefly: - 1.5 × 10 = 15

   (a) Examine the worst case complexity of following code snippets:

   a) for(i=0;i<n;i++)
   for(j=n;j>0;j=j/2)
   Statement;

   b) for(i=0;i<n;i++)
   for(j=10;j>0;j=j/2)
   Statement;

   (b) List any 5 applications of stack. Convert the expression (A+B)*(C/D^E) into prefix notation.

   (c) Establish a mathematical expression to compute the location where next element will be added in a static circular queue.

   (d) Determine the minimum and maximum height of a binary tree having 9 nodes. Also, determine that how many distinct binary trees could be constructed with 9 nodes.

   (e) Identify any two applications of a min heap. What is the worst case complexity of deleting an element from the heap?

   (f) Can we represent a binary tree using array? How can you find the left child and right child for i^{th} element?

   (g) Identify the situations when the worst case complexity of searching a node in BST becomes O(n). Discuss the mechanism you will use to reduce the complexity to O(log n)

   (h) Determine the minimum and maximum number of keys that can be accommodated in a node of B-tree having order 5.

   (i) Let us assume a circular queue is implemented through liked list with front and rear pointers. Identify the number of pointers which will be affected in performing insertion and deletion of items.

(j) We have traversed a BST and we found that the inorder and postoder traversal are same. Is it possible? If yes, then contract such a BST by assuming appropriate data items.

UNIT - I

2. (a) Design a data structure to accommodate two stacks in a single array. Write code snippet (single function) to perform push and pop operations in any of the stack based on choice parameter. 5

   (b) Design an algorithm to convert infix notation to postfix notation. 5

   (c) Create code snippet (function code) in 'C' to insert an element to its right position in a sorted singly linked list. 5

3. (a) Discuss different types of queues used in real-life scenarios. Identify the main problem in implementing a linear queue using array. Design an appropriate data structure to solve the problem of linear queue. 5

   (b) Create code snippet (function code) in 'C' to check the balance of parenthesis (opening and its corresponding closing parenthesis) using appropriate data structure. 5

   (c) Implement a linear queue (using array) which allows the insertion at front and deletion form rear. Write a function in 'C' to perform enqueue operation in the queue. 5

UNIT - II

4. (a) Construct an AVL tree for following input sequence: 15, 20, 24, 10, 13, 7, 30, 36, 25 5

   (b) Draw a binary tree whose inorder and preorder traversal are given below: 5
   Inorder: 2, 5, 6, 10, 12, 14, 15
   Preorder: 10, 5, 2, 6, 14, 12, 15
   After drawing the binary tree, find its postorder traversal.

   (c) Assume that we have implemented the threaded BST. Write a function in 'C' for inorder traversal without using recursion and stack. 5

5. (a) Explore the algorithms of max heap construction and deletion. Discuss the time complexity of retrieving and deleting an element from the max heap. 5

   (b) Construct a BST (step-by-step) by inserting the following input sequence: 5, 9, 11, 3, 6, 4, 7, 2, 8. After constructing the tree, delete the nodes having values 6 and 9. Write a function in 'C' to find the largest node in a BST. 5

   (c) Compare B-tree with BST. Construct a B-tree of order 5 for following input sequence: 10, 40, 30, 35, 20, 15, 50, 28, 25, 5, 60, 19, 12, 38, 27, 90, 45, 48. 5