# ADVANCED DATA STRUCTURES
## (CSEN 5101)

**Time Allotted : 2½ hrs** **Full Marks : 60**

***Figures out of the right margin indicate full marks.***

***Candidates are required to answer Group A and any 4 (four) from Group B to E, taking one from each group.***

***Candidates are required to give answer in their own words as far as practicable.***

## Group – A

1. Answer any twelve: **12 × 1 = 12**

*Choose the correct alternative for the following*

(i) What is the maximum possible height of a Red Black (RB) tree with 15 internal nodes?
(a) 2 (b) 4 (c) 6 (d) 8.

(ii) Two matrices, with number of non-zero elements k1 and k2 respectively, have been stored in their triple formats in sorted order of the row and column indices, respectively. The worst-case time complexity for adding the matrices is:
(a) O(min(k1, k2)) (b) O(max(k1, k2))
(c) O(k1 + k2) (d) O(k1 * k2)

(iii) Which of the following represents a binary Min-Heap?
(a) {10, 40, 60, 50, 80, 85} (b) {10, 40, 80, 60, 50, 85}
(c) {10, 40, 50, 60, 85, 80} (d) {10, 40, 80, 85, 60, 50}.

(iv) Suppose there is an AVL tree of height 4. What is the minimum possible number of nodes that can be present in the tree?
(a) 13 (b) 12 (c) 11 (d) 10.

(v) The maximum number of nodes possible in a Binary search tree of height h is
(a) $2^{h-1} - 1$ (b) $2^{h-1} + 1$ (c) $2^{h} + 1$ (d) $2^{h+1} - 1$.

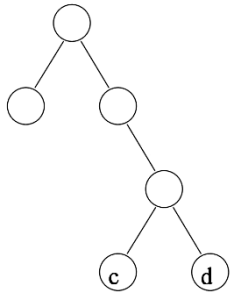
(vi) Which of the following tree can be most effectively represented by an array?
(a) AVL Tree (b) Almost complete Binary Tree
(c) Binary Search Tree (d) B-tree.

(vii) In an open addressing method with quadratic probing, the hash function is given by h(k) = k % 7, and the required probe sequences are $h_i(k) = (h(k) \pm i^2) \% 7$
i = 0, 1, 2, …, 6
The index of the hash table for $h_{-2}(15)$ is
(a) -3 (b) 3 (c) -4 (d) 4.

(viii) The time complexity of heapify operation on an array of n elements is
(a) O(log n) (b) O(n) (c) O(n * log n) (d) $O(n^2)$.

(ix) Which of the following is true for a B-tree of order 5?
(a) The root can have a maximum of 5 child nodes
(b) Each leaf node has at least 2 keys stored in it
(c) The heights of leaf nodes may differ
(d) None of the above.

(x) The computed code word for d from the tree in the adjacent figure using Huffman coding is
(a) 001 (b) 011 (c) 110 (d) 111.




*Fill in the blanks with the correct word*

(xi) If a hash table is 25% full, its load factor is _____.

(xii) The number of internal nodes of a 2-3 tree having 9 leaves is _____.

(xiii) If the postorder traversal of a binary search tree is 1, 4, 7, 6, 3, 13, 14, 10, 8, its inorder traversal after inserting the key 5 is _____.

(xiv) A positive integer n, when expressed in octal (i.e., radix 8) notation has 12 octal digits with no leading zeroes. If n is now expressed in hexadecimal (i.e., radix 16) notation, the number of hexadecimal digits will be _____.

(xv) If a node X is both the leftmost and rightmost node in a binary search tree, then the degree of X is _____.

## Group - B

2. (a) Show the steps of inserting the following key elements into a maximum priority queue, according to the given order of insertion:
6, 10, 20, 23, 13, 14, 16, 98. *[(CO2)(Analyse/IOCQ)]*

(b) Write the pseudo-code of a procedure to maintain the max-heap property in an array. *[(CO2)(Understand/LOCQ)]*

(c) An array contains the numbers as shown below. Show the steps to sort these numbers in ascending order using heapsort algorithm.

| 20 | 15 | 3 | 10 | 2 | 1 |
|---|---|---|---|---|---|

*[(CO2)(Understand/LOCQ)]*

**4 + 4 + 4 = 12**

3. (a) Insert the following 13 keys in the given sequence into a hash table T[0..16] of size 17, using the open addressing method with linear probing:
18, 15, 92, 44, 58, 26, 13, 24, 37, 74, 66, 82, 38. *[(CO3)(Analyse/IOCQ)]*

(b) Consider a hash table with 11 slots. The hash function is h(key) = key % 11. The collisions are resolved by chaining. The eleven keys are inserted in the order given below:

37, 23, 13, 21, 24, 7, 8, 15, 57, 43, 35

Show the hash table after all the keys are inserted. Find the maximum, minimum and average chain lengths in the hash table. What is the load factor (α)?

*[(CO3)(Analyse/IOCQ)]*

**5 + (5 + 2) = 12**

## Group - C

4. (a) For any nonempty binary tree (T), if $n_{leaf}$ is the number of leaf nodes and $n_2$ is the number of nodes with degree 2, then prove that

$$n_{leaf} = n_2 + 1.$$

*[(CO1)(Understand/LOCQ)]*

(b) Suppose a node N of a binary search tree has a degree of 2. Write an algorithm to compute its inorder successor. *[(CO4)(Create/HOCQ)]*

(c) The in-order and post-order traversal results of a binary tree are given below:

In-order: 17, 19, 39, 41, 46, 52, 61, 63, 74, 85, 88, 96

Post-order: 17, 39, 41, 19, 46, 61, 74, 88, 96, 85, 63, 52

Construct the binary tree, showing each step. *[(CO2)(Analyse/IOCQ)]*

**3 + 4 + 5 = 12**

5. (a) Derive a formula to find out the number of all possible Binary Search Trees, given N distinct elements. *[(CO2)(Analyse/HOCQ)]*

(b) There are two Binary Search Trees ($B_1$ and $B_2$), representing the same set of values. The height of $B_1$ is $h_1$ and that of $B_2$ is $h_2$. Given $h_1 < h_2$, which one ($B_1$ or $B_2$) is more efficient? Justify your answer. *[(CO2)(Analyse/HOCQ)]*

(c) Show the necessary steps of constructing a binary search tree using the following key elements according to the given order of insertion:

20, 15, 30, 5, 17, 25, 37, -5, 7, 23, 33, 45

Now delete the following keys in sequence from the tree:

7, 20, 15, 37, 30, 5. *[(CO1,CO2,CO3,CO4)(Apply/IOCQ)]*

**4 + 2 + (3 + 3) = 12**

## Group - D

6. (a) Construct an AVL (height-balanced) tree T of height = 5 on the following set S of 14 keys and indicate the balance factor at each node.

S = {12, 15, 22, 23, 34, 37, 49, 51, 55, 60, 68, 70, 75, 79}. *[(CO3)(Analyse/IOCQ)]*

(b) Write a pseudo code for deleting an element in a skip list.

*[(CO2,CO3,CO4,CO6)(Evaluate/HOCQ)]*

**(3 + 3) + 6 = 12**

7. (a) Define a B-Tree. When does a B-Tree become a 2-3 Tree? *[(CO3) (Remember/LOCQ)]*

(b) Show the necessary steps of constructing a B-Tree of order 5, using the following key elements according to the given order of insertion as follows:

30,45,70,95,5,10,20,32,35,40,42,50,60,85,90,100,120,37

*[(CO2,CO3,CO4,CO6)(Apply/IOCQ)]*

(c) Show the necessary steps of constructing a Red-Black Tree using the following key elements according to the given order of insertion:

22, 10, 44, 56, 98, 32, 64,75 . *[(CO6)(CO2)(CO3)(CO4)(Apply/HOCQ)]*

**(2 + 1) + 4 + 5 = 12**

## Group - E

8. (a) What is a k-D tree? If your search space has n number of points, what will be the height of the k-D tree? Draw the k-D tree that results from inserting the following points:

[A(1, 2), B(3, 1), C(3, 4), D(2, 2), E(0, 4), F(3, 3), G(0, 0)]

Show each intermediate step. *[(CO2,CO3)(Understand/LOCQ),(Apply/IOCQ)]*

(b) Show the necessary steps and data of searching the pattern "ABABCD" in the text "DCABCCABABCCABABCDABCD", using Knuth Morris Pratt Algorithm. *[(CO1,CO5,CO6)(Apply/IOCQ)]*

**(2 + 4) + 6 = 12**

9. (a) State the difference between a Trie data structure and a Binary Search Tree. Out of the two options – Hash Table and Trie to keep the data, which one is preferable? Justify your answer. *[(CO3)(Remember,Understand/LOCQ)]*

(b) How is one dimensional range searching performed? Analyse its time complexity. *[(CO2)(Understand/LOCQ)]*

(c) This question relates to computational geometry. Let Q = {$p_1$, $p_2$, $p_3$, $p_4$} be a set of 4 points in the X-Y plane. We want to determine whether the line segment joining $p_1$ and $p_2$ intersects the line segment joining $p_3$ and $p_4$. Describe briefly a method that can be used to solve the problem. For illustration, use the points $p_1$ = (10, 5), $p_2$ = (5, 10), $p_3$ = (6, 8), $p_4$ = (15, 3). *[(CO6)(Apply/HOCQ)]*

**(2 + 1) + 3 + 6 = 12**

---

| Cognition Level | LOCQ | IOCQ | HOCQ |
|---|---|---|---|
| Percentage distribution | 20.83 | 53.13 | 26.04 |

**Course Outcome (CO):**

After the completion of the course students will be able to:

CSEN5101.1 Remember definitions and notations of basic terminologies used in data structures.

CSEN5101.2 Learn and understand abstract data types and its significance; differentiate between linear and non-linear data structures for solving real world problems.

CSEN5101.3 Understand and apply some of the special trees, Tries data structure and various hashing techniques.

CSEN5101.4 Design modular algorithms on linear and non-linear data structures for solving engineering problems efficiently.

CSEN5101.5 Understand and analyse the basic principles of different string-matching algorithms and identify their advantages and disadvantages.

CSEN5101.6 Evaluate the performance of different data structures with respect to various applications.

**LOCQ: Lower Order Cognitive Question; IOCQ: Intermediate Order Cognitive Question; HOCQ: Higher Order Cognitive Question.*