Roll No. ..... Total Pages : 3

CMTE/D-24 **24056**

ADVANCED DATA STRUCTURES

Paper–MT–CSE–20–12

Time allowed : 3 Hours] [Max. Marks : 75

Note : Attempt **five** questions in all, selecting **one** question from each unit. Question No. **1** is compulsory. All questions carry equal marks.

Compulsory Question

1. Attempt all questions: 6×2½=15
   (i) Describe a common application of range searching in real–world scenarios.
   (ii) Explain the impact of the load factor on the performance of a hash table.
   (iii) List two differences between AVL Trees and Red–Black Trees.
   (iv) What are the two main heuristics used in the Boyer–Moore algorithm?
   (v) How does a suffix trie help in string matching?
   (vi) Explain the advantage of using a priority search tree for range searching.

UNIT–I

2. (a) What is a hash function and what are its desirable properties? Give an example of a simple hash function. 7½

   (b) Explain the concept of rehashing and discuss when it should be applied in a hash table. 7½

3. What is a dictionary? How can you implement the dictionary in the computer? Explain by using suitable algorithms and examples. 15

UNIT–II

4. Explain the Skip List data structure in detail. Describe how search, insertion and deletion operations work in Skip Lists and discuss their average–case time complexities. 15

5. Discuss the structure of B–Trees and their importance in disk–based systems. Explain how insertion and deletion operations are managed in B–Trees and describe their advantages in large-scale data storage. 15

UNIT–III

6. (a) Explain the basic string operations used in text processing, such as concatenation, substring and comparison. Why are these operations fundamental? 7½

   (b) Define a standard trie and describe its structure. How are standard tries used in text processing? 7½

7. Discuss the Longest Common Subsequence (LCS) problem. Explain the dynamic programming approach to solving it, including the steps involved in constructing the solution matrix and retrieving the LCS. 15

UNIT–IV

8. (a) What is one-dimensional range searching and how is it implemented? Discuss its time complexity. 7½

   (b) Describe the structure of a priority search tree and its use in range searching. 7½

9. Explain the insertion and deletion operations in a k–D tree. Discuss how these operations affect the tree structure and balance in different dimensions? 15