

BACHELOR OF TECHNOLOGY (CBCS - 2023)
B. Tech. Sem-V Computer Science & Engineering : WINTER: 2025
SUBJECT: DESIGN & ANALYSIS OF ALGORITHM

Day : Monday
 Date : 08/12/2025

W-30724-2025

Time : 02:30 PM-05:30 PM
 Max. Marks : 60

NB :

1. All questions are COMPULSORY.
2. Figures to the right indicate FULL marks.
3. Draw neat labelled diagram WHEREVER necessary.

Q. 1 Describe how asymptotic notations help in comparing algorithm performance. Compare the best-case, average-case, and worst-case complexities of insertion sort and justify how input distribution affects running time. CO1 BT1 (10)

OR

Q. 1 Write short notes on the following (**any two**): CO1 BT3 (10)

- a) Role of growth of functions in algorithm efficiency
- b) Importance of pseudocode in algorithm design
- c) Relationship between time complexity and recurrence relations
- d) Differences between substitution, recursion tree, and Master's methods

Q. 2 Describe the Merge Sort algorithm in detail using the divide-and-conquer approach. Explain its working with a neat diagram showing recursive splitting and merging. Derive the recurrence relation for Merge Sort and solve it to find its time complexity. CO2 BT1 (10)

OR

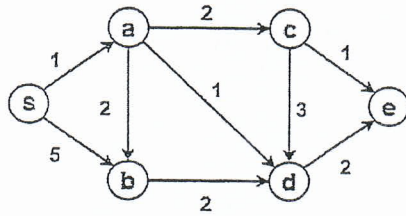
Q. 2 What is performance analysis in algorithms? Differentiate between the best case, average case, and worst-case performance of an algorithm. Illustrate your answer with suitable examples for:
 a) Linear Search
 b) Quick Sort
 Explain why worst-case analysis is important in algorithm design. CO2 BT1 (10)

Q. 3 Explain the Longest Common Subsequence (LCS) problem using dynamic programming. Construct the LCS table for the strings:
 X = "ABCDGH" and Y = "AEDFHR".
 Find the length of LCS and reconstruct the subsequence. CO3 BT2 (10)

OR

Q. 3 Explain Dijkstra's algorithm for finding the single-source shortest path in a weighted graph. Show the working of the algorithm step-by-step on the CO3 BT2 (10)

following graph:



Also analyse its time complexity using adjacency matrix and adjacency list.

- Q. 4 Explain the Backtracking technique in detail. Discuss the concepts of state-space tree, implicit constraints, explicit constraints, and bounding functions. Use 8-Queen Problem to illustrate how backtracking systematically explores and prunes solution spaces. CO4 BT2 (10)

OR

- Q. 4 Compare Backtracking and Branch and Bound techniques in terms of:
a) Approach and methodology
b) State-space tree representation
c) Use of bounding functions
d) Types of problems solved CO4 BT4 (10)

- Q. 5 Describe the KMP string matching algorithm. Explain how the prefix-function (LPS table) is constructed. Construct the LPS table for the pattern P = "ABABAC" and use it to perform the KMP matching on the text T = "ABABABACABABAC". Also analyse the time complexity of KMP. CO5 BT1 (10)

OR

- Q. 5 Explain the Rabin-Karp algorithm with the concept of rolling hash. Illustrate the hashing process and pattern matching for the pattern "TEST" in the text "THISISTESTTEXT". CO5 BT2 (10)

- Q. 6 Explain the following classes with suitable examples:
(a) Class P
(b) Class NP
(c) NP-Complete
(d) NP-Hard
Also explain the relationship between these classes with a neat diagram. CO6 BT2 (10)

OR

- Q. 6 What is polynomial-time reducibility? Explain how NP-Reducibility is used to prove that a problem is NP-Complete. Illustrate your answer with an example. CO6 BT1 (10)
