Reg. No.															

Question Paper Code	13606

B.E. / B.Tech. - DEGREE EXAMINATIONS, APRIL / MAY 2025

Fourth Semester

Information Technology

(Common to Computer Science and Engineering, Computer Science and Engineering (AIML), Computer Science and Engineering (IOT), Computer Science and Engineering (Cyber Security) & M.Tech. - Computer Science and Engineering (5Years Integrated))

20ITPC401 - DESIGN AND ANALYSIS OF ALGORITHMS

Regulations - 2020

Duration: 3 Hours Max. Marks: 100

PART - A (MCQ) (10 × 1 = 10 Marks)

Answer ALL Questions

Marks K – Level CO

1. Which of the following is NOT a fundamental step in algorithmic problem-solving? *1 K1 CO1*
   (a) Understanding the problem
   (b) Designing an algorithm
   (c) Writing the algorithm in only one specific programming language
   (d) Testing and debugging
2. Which of the following best describes an algorithm? *1 K1 CO1*
   (a) A computer program written in a specific language.
   (b) A set of well-defined instructions for solving a problem.
   (c) A complex mathematical formula.
   (d) A hardware component of a computer.
3. What is the main characteristic of the brute force method? *1 K1 CO2*
   (a) It always finds the optimal solution using heuristics
   (b) It systematically tries all possible solutions
   (c) It avoids trying all possible solutions
   (d) It only works for small dataset.
4. Which of the following problems is NOT typically solved using brute force? *1 K1 CO2*
   (a) String Matching (b) Matrix Multiplication
   (c) Sudoku Solving (d) Convex Hull
5. Analyze how the divide and conquer approach solves problems: *1 K1 CO3*
   (a) Iterating through all possible solutions
   (b) Splitting the problem into independent subproblems
   (c) Using a single recursive call
   (d) Storing results of subproblems to avoid recomputation
6. Estimate the approximate number of comparisons in binary search for an array of size n: *1 K1 CO3*
   (a) $\log_2(n)$ (b) n/2 (c) n^2 (d) n log n
7. Floyd Warshall Algorithm belongs to ---? *1 K1 CO4*
   (a) Greedy method (b) Branch and Bound
   (c) Divide and Conquer (d) Dynamic method
8. Recognize which problem is NOT typically solved using dynamic programming. *1 K1 CO4*
   (a) 0/1 knapsack problem (b) Matrix chain multiplication problem
   (c) Edit distance problem (d) Fractional knapsack problem
9. Evaluate the role of heuristics in an iterative improvement algorithm: *1 K1 CO5*
   (a) They guarantee an optimal solution
   (b) They guide the search towards feasible solutions
   (c) They reduce the complexity of the search space
   (d) They are not applicable in iterative improvement

10. Identify the correct relationship between P, NP, NP-Complete, and NP-Hard problems. *1 K1 CO6*
(a) P ⊆ NP ⊆ NP-Complete ⊆ NP-Hard
(b) P ⊆ NP ⊆ NP-Hard ⊆ NP-Complete
(c) P ⊆ NP, NP-Complete ⊆ NP, NP-Hard ⊇ NP-Complete
(d) P = NP = NP-Complete = NP-Hard

PART - B (12 × 2 = 24 Marks)

Answer ALL Questions

11. Define the concept of an algorithm and list its desirable properties. *2 K1 CO1*
12. Compare best, worst, and average time complexities. *2 K2 CO1*
13. Recall the Brute Force approach. *2 K1 CO2*
14. Define a brute force algorithm to count the number of vowels in a given text. *2 K1 CO2*
15. Classify different types of problems that can be efficiently solved using the divide and conquer approach. *2 K2 CO3*
16. Write the formula to compute Multiplication of large integers. *2 K2 CO3*
17. Differentiate between the Greedy Method and Dynamic Programming. *2 K2 CO4*
18. Show the Principle of Optimality in Dynamic Programming. *2 K2 CO4*
19. List the three properties of flow networks. *2 K1 CO5*
20. Write brief notes on simplex method in linear programming. *2 K1 CO5*
21. Define the terms NP-Completeness and NP-Hard with appropriate examples. *2 K1 CO6*
22. Differentiate between P and NP problems with reference to time complexity. *2 K2 CO6*

PART - C (6 × 11 = 66 Marks)

Answer ALL Questions

23. a) Formulate a general plan for analyzing the time efficiency of recursive algorithms. Apply recurrence to find the number of moves required for the Towers of Hanoi problem for n disks. *11 K2 CO1*

OR

b) Develop a general plan for analyzing the time efficiency of non-recursive algorithms and determine the time complexity for finding the largest element in an array. *11 K2 CO1*

24. a) Analyze the concepts of the following: *11 K2 CO2*
(i) Brute Force String Matching Algorithm.
(ii) Convex Hull problems using the brute force approach.

OR

b) Apply the brute force approach to find the optimal solution for the given knapsack problem: *11 K2 CO2*

Item	Weight	Value
1	2	$1
2	3	$2
3	4	$8
4	5	$6

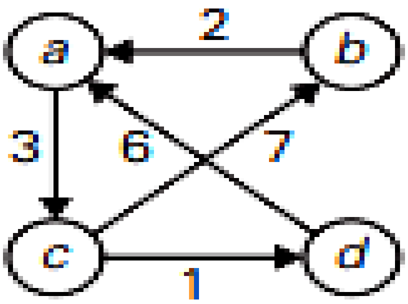
Capacity W=8.

25. a) Illustrate the working of the binary search algorithm with a step-by-step dry run on the following sorted array: *11 K2 CO3*
Array: [2, 4, 7, 10, 15, 18, 20, 25, 30, 35]
Search Element: 15

OR

b) Define Divide and Conquer strategy and explain the closest-pair with suitable example. *11 K2 CO3*

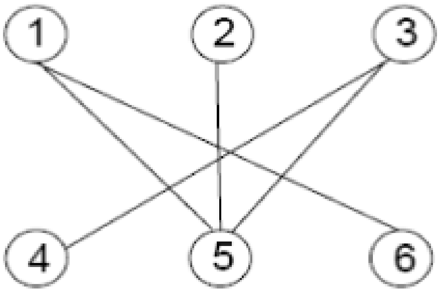
26. a) Implement Floyd’s Algorithm for All-Pairs Shortest Path using the given weighted graph. Analyze its time complexity. *11 K3 CO4*



OR

b) Solve the 0/1 Knapsack Problem using the Greedy Approach for the following items are given: *11 K3 CO4*
Items: {A, B, C, D}
Weights: {2, 3, 4, 5}
Values: {3, 4, 5, 6}
Knapsack Capacity = 5

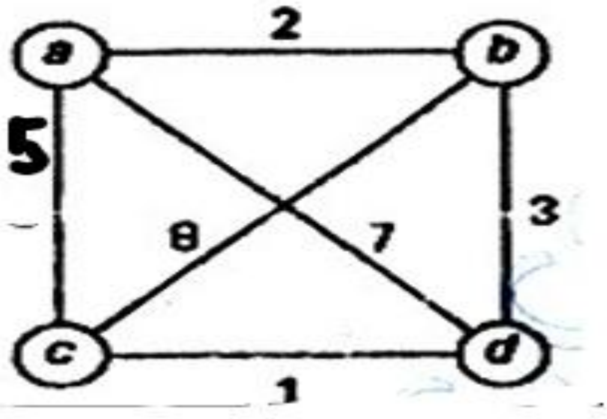
27. a) (i) Write the algorithm for maximum matching in Bipartite Graphs and prove the theorem with example. *11 K3 CO5*
(ii) Apply the maximum matching algorithm to the following bipartite graphs.



OR

b) Explain the algorithm for stable marriage problem and prove the theorem with Example. *11 K2 CO5*

28. a) Construct a solution to the Hamiltonian Circuit Problem using the backtracking technique and demonstrate the process. *11 K3 CO6*



OR

b) Apply an approximation algorithm for the Travelling Salesman Problem (TSP). Given the following distance matrix for 4 cities: *11 K3 CO6*

	A	B	C	D
A	-	10	15	20
B	10	-	35	25
C	15	35	-	30
D	20	25	30	-

- Use the Nearest Neighbor Algorithm to approximate the TSP tour starting from city A.
- Show the tour, the path taken, and the total distance traveled.