Reg. No. : | | | | | | | | | | |

Question Paper Code: 31424

B.E. / B.Tech. DEGREE EXAMINATION, MAY 2016

Fourth Semester

Computer Science and Engineering

01UCS404 - DESIGN AND ANALYSIS OF ALGORITHMS

(Regulation 2013)

Duration: Three hours Maximum: 100 Marks

Answer ALL Questions

PART A - (10 x 2 = 20 Marks)

1. Write a note on measures used for finding efficiency of an algorithm.
2. Differentiate recursive and non-recursive algorithms.
3. Give the benefit of application of brute force technique to solve a problem.
4. Estimate the average number of comparisons for the unsuccessful search.
5. Draw the table of the dynamic programming algorithm for constructing an optimal binary search tree.
6. Compute the space complexity of optimal binary search tree algorithm.
7. Show the mathematical formulation to solve a max flow problem.
8. State the usage of stable marriage problem?
9. Determine the additional features required in branch-and-bound when compared to backtracking.
10. Define Hamiltonian circuit problem in an undirected graph.

PART - B (5 x 16 = 80 Marks)

11. (a) Formulate a non recursive algorithm for finding the Fibonacci sequence and show that n3logn is w(n3) derive its time complexity. (16)

Or

(b) Explain briefly Big-oh Notation, Omega Notation and Theta Notations. Give examples. (16)

12. (a) Write a pseudo code for divide and conquer algorithm for merging two sorted arrays into a single sorted one. Explain with an example. Set up and solve a recurrence relation for the number of key comparisons made by the pseudo code. (16)

Or

(b) (i) Distinguish between Quicksort and mergesort and arrange the following numbers in increasing order using mergesort (18, 29, 68, 32, 43, 37, 87, 24, 47, 50). (8)

(ii) The worst-case time of procedure MERGESORT is O(nlogn). What is its time in the best case? Can we say that the time for merge sort is O(nlogn)? (8)

13. (a) Solve the following instance of the Knapsack problem by the greedy method.

Item	Weight	Profit
1	2	$ 10
2	3	$ 5
3	5	$ 15
4	7	$ 7
5	1	$ 6
6	4	$ 18
7	1	$ 3

The Knapsack capacity m = 15. (16)

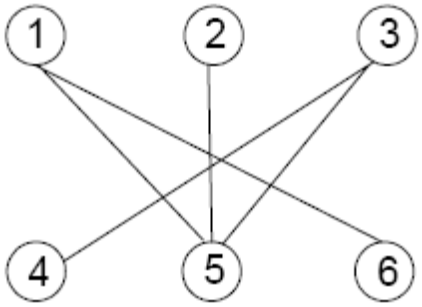
Or

(b) (i) Construct the optimal binary search tree for the following key values. (8)

Key	A	B	C	D
Probability	0.1	0.2	0.4	0.3

(ii) Write a linear time algorithm, that generates the optimal binary search tree for the root table. (8)

14. (a) Apply the maximum matching algorithm to the following bipartite graphs. (16)



Or

(b) Explain briefly about the maximum-flow problem with an example. (16)

15. (a) (i) What are the problems that can be solved using branch and bound technique? Also give a general template for branch and bound technique. (8)

(ii) Demonstrate the method to solve 0/1 knapsack problem using branch and bound technique. Apply it for the instance p = {11, 21, 31, 33, 43, 53, 55, 65}, w = {1, 11, 21, 23, 33, 43, 45, 55}, m = 110 and n = 8. Construct the portion of the state space tree. (8)

Or

(b) (i) Explain NP hard and NP complete problems with example and analyze the complexity. (8)

(ii) Write an algorithm to solve an assignment problem using branch and bound technique. Also apply it to solve the following instances: (8)

	Job 1	Job 2	Job 3	Job 4
Person 1	4	3	8	6
Person 2	5	7	2	4
Person 3	16	9	3	1
Person 4	2	5	3	7