Course Code: 18CS0516 **R18**

SIDDHARTH GROUP OF INSTITUTIONS:: PUTTUR
(AUTONOMOUS)
Siddharth Nagar, Narayanavanam Road – 517583

QUESTION BANK (DESCRIPTIVE)

Subject with Code: DAA(18CS0516) **Course & Branch**:B.Tech - CSIT

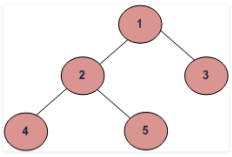
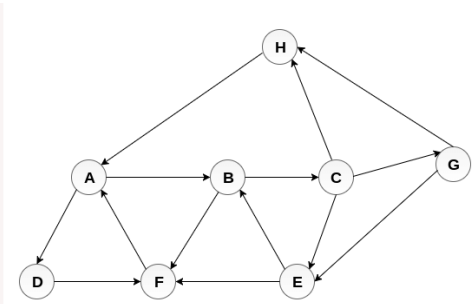
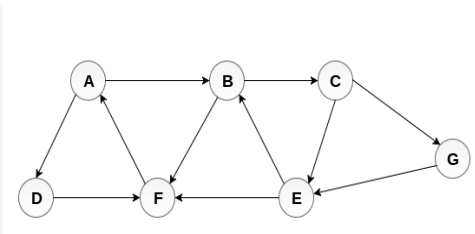
Year & Sem: III-B.Tech & I-Sem **Regulation:** R18

UNIT –I
INTRODUCTION , DISJOINT SETS

1	**a**	What is an algorithm?	[L1][CO1]	**[2M]**
	b	Write the For LOOP general format.	[L1][CO1]	**[2M]**
	c	Arrange the following function in increasing order. $n, \log n, n^2, n^3, n\log n, 2^n$	[L1][CO1]	**[2M]**
	d	Solve that $1/2n^2-3n=\theta(n^2)$.	[L3][CO1]	**[2M]**
	e	List out the steps that need to design an algorithm.	[L1][CO1]	**[2M]**
2	**a**	What is asymptotic notation? Explain different types of notations with examples?	[L2][CO1]	**[6M]**
	b	Illustrate an algorithm for (i) Finding factorial of n number (ii)Sum of n natural numbers	[L2][CO1]	**[4M]**
3		Simplify steps involved in performance analysis with example.	[L2][CO1]	**[10M]**
4	**a**	What do you mean by algorithm? List some of the properties of it?	[L1][CO1]	**[5M]**
	b	Apply the Master's theorem. Solve the following Recurrence relations i) $T(n) = 4T(n/2) + n$ i000i) $T(n) = 2T(n/2) + n\log n$	[L3][CO1]	**[5M]**
5	**a**	Classify the rules of Pseudo code for Expressing Algorithms?	[L2][CO1]	**[7M]**
	b	Solve the given function -If $f(n)= 5n^2+ 6n+ 4$ then prove that $f(n)$ is $0(n^2)$.	[L3][CO1]	**[3M]**
6	**a**	Explain the collapsing rule for Find algorithm with example.	[L6][CO1]	**[5M]**
	b	Solve the following Recurrence relation i) $T(n) = 4T(n/3) + n^2$ ii) $T(n) = 6T(n/3) + n^2 \log n$	[L3][CO1]	**[5M]**
7		Estimate the recurrence relations: i) $x(n) = x(n-1) + 5$ for $n>1$, $x(1) = 0$ ii) $x(n) = 3x(n-1)$ for $n>1$, $x(1) = 4$ iii) $x(n)= x(n/2) + n$ for $n>1$, $x(1) = 1$(solve for $n = 2^k$) iv) $x(n)= x(n/3) + 1$ for $n>1$, $x(1) = 1$(solve for $n = 3^k$)	[L6][CO1]	**[10M]**
8	**a**	Determine in steps of Union and Find algorithms with example.	[L5][CO1]	**[5M]**
	b	Explain space complexity in detail.	[L2][CO1]	**[5M]**
9	**a**	Define disjoint sets? Explain different types of disjoint sets operations with examples?	[L2][CO1]	**[6M]**
	b	Solve the following recurrence: i) $T(n)=7T(n/3) + n^2$ ii) $T(n)=3T(n/2) + n$	[L3][CO1]	**[4M]**
10		Explain two types of recurrences in detail with suitable example.	[L6][CO1]	**[10M]**

Course Code: 18CS0516 **R18**

UNIT –II
BASIC TRAVERSAL AND SEARCH TECHNIQUES, DIVIDE AND CONQUER

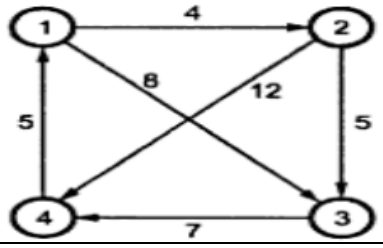
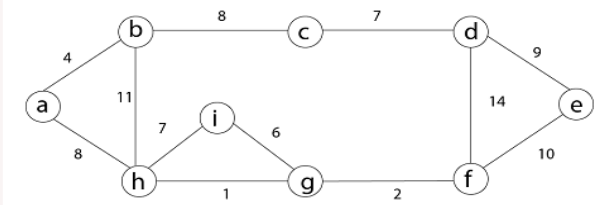
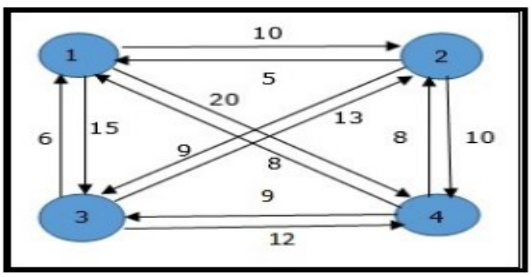
1	**a**	Define the divide and conquer method.	[L1][CO2]	**[2M]**
	b	Give the recurrence relation of divide-and-conquer.	[L1][CO2]	**[2M]**
	c	List out the formulas for Strassen's matrix multiplication.	[L1][CO2]	**[2M]**
	d	Write the recurrence relation for quick sort and analyze time complexity?	[L1][CO2]	**[2M]**
	e	Find the In order and preorder and post order tree traversal for the following binary tree. 	[L1][CO2]	**[2M]**
2		What is divide and conquer strategy? Explain the working strategy of Binary Search and find element 60 from the below set by using the above technique: { 10, 20, 30,40,50, 60,70}. Analyze time complexity for binary search.	[L2][CO2]	**[10M]**
3		Analyze the working strategy of merge sort and illustrate the process of merge sort algorithm for the given data: 43, 32, 22, 78, 63, 57, 91 and 13.	[L4][CO2]	**[10M]**
4		A=$\begin{bmatrix} 9 & 4 & 6 & 7 \\ 7 & 8 & 1 & 4 \\ 4 & 3 & 2 & 6 \\ 5 & 3 & 0 & 2 \end{bmatrix}$ B=$\begin{bmatrix} 7 & 6 & 2 & 1 \\ 3 & 9 & 0 & 3 \\ 2 & 5 & 2 & 9 \\ 3 & 2 & 4 & 7 \end{bmatrix}$. Create Stassen's matrix multiplication on A and B find the Resultant matrix	[L6][CO2]	**[10M]**
5	a	Sort the records with the following index values in the ascending order using quick sort algorithm. 9, 7, 5, 11, 12, 2, 14, 3, 10, 6.	[L2][CO2]	**[5M]**
	b	Write and explain the control abstraction for Divide and conquer.	[L2][CO2]	**[5M]**
6		Explain the Strassen's algorithm for matrix multiplication and analyze time complexity.	[L5][CO2]	**[10M]**
7		Explain DFS algorithm and trace out minimum path for DFS for the following example. 	[L5][CO2]	**[10M]**
8		Summarize an algorithm for quick sort. Provide a complete analysis of quick sort for given set of numbers 12, 33, 23, 43, 44, 55, 64, 77and 76.	[L2][CO2]	**[10M]**
9		Elaborate BFS algorithm and trace out minimum path for BFS for the following example. 	[L6][CO2]	**[10M]**
10	**a**	Compare between BFS and DFS techniques.	[L4][CO2]	**[4M]**
	b	Solve an algorithm for techniques of binary trees with examples.	[L3][CO2]	**[6M]**

Course Code: 18CS0516 **R18**

UNIT –III
GREEDY METHOD, DYNAMIC PROGRAMMING

1	**a**	What is meant by feasible solution?	[L1][CO3]	[2M]
	b	Write the general algorithm for Greedy method control abstraction.	[L1][CO3]	**[2M]**
	c	What is Knapsack problem?	[L1][CO3]	**[2M]**
	d	Define optimal solution.	[L1][CO3]	**[2M]**
	e	Define dynamic programming.	[L1][CO3]	**[2M]**
2	Construct an optimal solution for Knapsack problem, where n=7, M=15 and (p1,p2,p3,p4,p5,p6,p7)=(10,5,15,7,6,18,3)and(w1,w2,w3,w4,w5,w6,w7)=(2,3,5,7,1,4,1) by using Greedy strategy.		[L3][CO3]	**[10M]**
3	Explain any one application of greedy method with an example?		[L2][CO3]	**[10M]**
4	Elaborate job sequencing with deadlines by using greedy method where given the jobs, their deadlines and associated profits as shown below. Calculate maximum earned profit.		[L6][CO3]	**[10M]**

Jobs	**J1**	**J2**	**J3**	**J4**	**J5**	**J6**
Deadlines	5	3	3	2	4	2
Profits	200	180	190	300	120	100

5	a	Explain in detail about greedy method and its applications.	[L2][CO3]	**[5M]**
	b	Simplify the algorithm for Knapsack problem and analyze time complexity.	[L4][CO3]	**[5M]**
6	Construct an algorithm for All pairs of shortest path and calculate shortest path between all pairs of vertices by using dynamic programming method for the following graph. 		[L6][CO3]	**[10M]**
7	Apply the minimum spanning tree of the following graph using Kruskals algorithm and prims algorithm . 		[L3][CO3]	**[10M]**
8	Explain 0/1 knapsack problem by using dynamic programming with an examples.		[L2][CO3]	**[10M]**
9	Analyze the minimum cost tour forgiven problem using travelling sales person Concepts. 		[L4][CO3]	**[10M]**
10	Build any one application of dynamic programming with an example.		[L6][CO1]	**[10M]**

Course Code: 18CS0516 **R18**

UNIT –IV
BACKTRACKING, BRANCH AND BOUND

1	**a**	State Sum of Subsets problem.	[L1][CO4]	**[2M]**
	b	What is graph coloring?	[L1][CO4]	**[2M]**
	c	Define state space tree.	[L1][CO4]	**[2M]**
	d	Define Branch-and-Bound method.	[L1][CO4]	**[2M]**
	e	Choose the searching techniques that are commonly used in Branch-and-Bound method.	[L1][CO4]	**[2M]**
2		Explain sum of subsets by using backtracking with an example.	[L5][CO4]	**[10M]**
3		Discuss the Hamiltonian cycle algorithm with step by step operation with example.	[L6][CO4]	**[10M]**
4	**a**	Explain the principles of FIFO branch and bound.	[L2][CO4]	**[5M]**
	b	Recall the graph coloring. Explain in detail graph coloring with an example.	[L5][CO4]	**[5M]**
5	**a**	Explain the properties of LC-search.	[L2][CO4]	**[5M]**
	b	Give brief description about the general method of branch and bound.	[L2][CO4]	**[5M]**
6		Select any one application of backtracking with an example.	[L3][CO4]	**[10M]**
7		Construct the LC branch and bound search. Consider knapsack instance n=4 with capacity M=15 such that pi={10,10,12,18},wi={2,4,6,9}apply LC branch and bound technique.	[L6][CO4]	**[10M]**
8		Simplify 0/1 knapsack problem and design an algorithm of LC Branch and Bound and find the solution for the knapsack instance of n = 4,(p1, p2, p3, p4) = (10, 10, 12, 18), (w1,w2, w 3, w4) = (2, 4, 6, 9) and M = 15.	[L4][CO4]	**[10M]**
9		Evaluate 0/1 knapsack problem using branch and bound with an example.	[L5][CO4]	**[10M]**
10		Distinguish in detail 8-queens problem using back tracking with state space tree.	[L4][CO4]	**[10M]**

Course Code: 18CS0516 **R18**

UNIT –V
NP-HARD AND NP-COMPLETE PROBLEMS

1	**a**	Define class P.	[L1][CO5]	**[2M]**
	b	Define NP- hard problem.	[L1][CO5]	**[2M]**
	c	What is Non-deterministic algorithm?	[L1][CO5]	**[2M]**
	d	What is a decision problem?	[L1][CO5]	**[2M]**
	e	Define NP.	[L1][CO5]	**[2M]**
2		Construct the non-deterministic algorithms with example.	[L3][CO5]	**[10M]**
3		Distinguish between deterministic and non-deterministic algorithms.	[L4][CO5]	**[10M]**
4		Construct the non-deterministic sorting algorithm and also analyze its complexity.	[L6][CO5]	**[10M]**
5		Explain the class of P and NP with example?	[L2][CO5]	**[10M]**
6		Differentiate between NP- complete and NP-hard problems?	[L4][CO5]	**[10M]**
7		State and explain cook's theorem?	[L2][CO5]	**[10M]**
8		Estimate the strategy to prove that a problem steps of NP-hard.	[L6][CO5]	**[10M]**
9		Illustrate the satisifiability problem and write the algorithm.	[L2][CO5]	**[10M]**
10		Determine the classes NP-hard and NP-complete problem with example.	[L5][CO5]	**[10M]**

Prepared by: Mr. V GOPI