INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal, Hyderabad - 500 043

COMPUTER SCIENCE AND ENGINEERING

QUESTION BANK

Course Title	**DESIGN AND ANALYSIS OF ALGORITHMS**				
Course Code	ACSC13				
Program	B.Tech				
Semester	IV	CSE			
Course Type	Core				
Regulation	IARE - UG20				
Course Structure	Theory			Practical	
	Lecture	Tutorials	Credits	Laboratory	Credits
	3	-	3	-	-
Course Coordinator	Ms. C Bhagyashree, Assistant Professor				

COURSE OBJECTIVES:

The students will try to learn:

I	Mathematical approach for Analysis of Algorithms.
II	Methods and techniques for analyzing the correctness and resource requirements of algorithms.
III	Different paradigms of algorithm design including recursive algorithms, divide-and-conquer algorithms, dynamic programming, greedy algorithms, Backtracking , Branch and Bound and graph algorithms.
IV	Strategies for solving problems not solvable in polynomial time.

COURSE OUTCOMES:

After successful completion of the course, students should be able to:

CO 1	**Find** the (worst case, randomized, amortized) running time and space complexity of given algorithms using techniques such as recurrences and properties of probability.	Remember
CO 2	**Apply** divide and conquer algorithms for solving sorting, searching and matrix multiplication.	Apply
CO 3	**Make Use of** appropriate tree traversal techniques for finding shortest path.	Understand

CO 4	**Compare** the efficiencies of problems using different algorithms such as searching, sorting and graph traversal..	Understand
CO 5	**Apply** greedy algorithm for finding solutions of minimization and maximization problems.	Apply
CO 6	**Apply** dynamic programing algorithms for calculating optimised solutions.	Apply
CO 7	**Utilize** backtracking and branch and bound techniques to deal with traceable and in-traceable problems..	Apply
CO 8	**Describe** the classes P, NP, NP-Hard, NP-complete for solving deterministic and non deterministic problems.	Understand
CO 9	**Develop** efficient algorithms for common computer engineering design problems.	Apply
CO 10	**Apply** knowledge and skills for employability and to succeed in national and international level competitive exams.	Apply

QUESTION BANK:

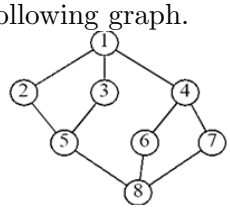
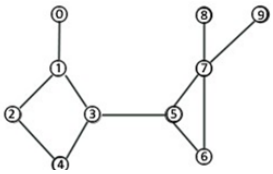
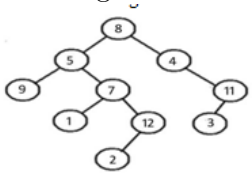
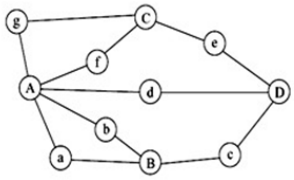
MODULE I				
DESIGN AND ANALYSIS OF ALGORITHMS				
PART A-PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS				
Q.No	**QUESTION**	**Taxonomy**	**How does this subsume the level**	**CO's**
1	Solve the following recurrence relation T(n)=2 T(n/2) + n, and T(1)=2.	Remember	Learner to recall the concept of recurrence relation relate the rules of recurrence relationandsolve the given recurrence relation.	CO 1
2	Solve the following recurrence relation T(n) = 7T(n/2)+cn2	Remember	Learner to recall the concept of recurrence relation thenrelate the rules of recurrence relationandsolve the given recurrence relation.	CO 1
3	Solve the recurrence relation T(n)=T(1), n=1 T(n)=T(n/2) + c, n¿1 and n is a power of 2	Remember	Learner to recall the concept of recurrence relation thenrelate the rules of recurrence relationandsolve the given recurrence relation.	CO 2

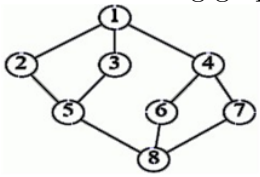
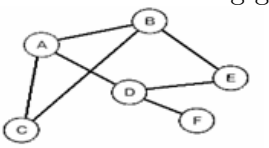
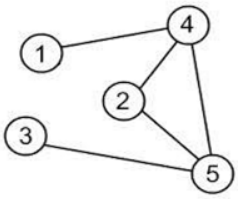
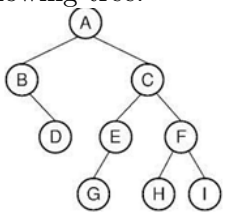
4	Apply quick sort algorithm and simulate it for following data sequence: 3 5 9 7 1 4 6 8 2	Remember	Learner to recallthe concept of divide and conquer then understandquick sort method and solvegiven data sequence.	CO 2
5	Identifythe tracing steps of merge sort and quicksort and analyze the time complexity for the following data: 33, 44, 2, 10, 25,	Remember	Learner to recallthe concept of divide and conquer then understand quick sort, merge sort method and solvegivendata.	CO 2
6	Organize the steps in merge sort to arrange following data in non-decreasing order1,2,5,6,9,8,7	Remember	Learner to recallthe concept of divide and conquer thenunderstand, merge sort method and solvegivendata.	CO 1
7	Organizemerge sort on following letters H, K, P,C,S,K,R,A,B,L	Remember	Learner to recallthe concept of divide and conquer then understand merges sort method and solvegivendata.	CO 2
8	ExplainStrassen's method outperforms the traditional matrix multiplication method. How many number of multiplication operations is required during multiplication of two matrices with size of 32 x 32 in Stressen's method.	Remember	Learner to recall concept of Strassen's matrix multiplication and Comparewith normal matrix multiplication for the any two 32 x 32	CO 2
9	Explain recurrence relation for Strassen's matrix.	Remember	Learner to recallthe concept of recurrence relation Then explainsthe concept on Strassen's matrix.	CO 2
10	Solve the following recurrence relation T (n) =2 T (n/2) + 1, and T (1) =2.	Remember	Learner to recall the concept of recurrence relation relate the rules of recurrence relationandsolve the given recurrence relation.	CO 1
PART B-LONG ANSWER QUESTIONS				
1	Definevarious asymptotic notations used for best case,average case and worst case analysis of algorithms.	Remember	—	CO 1

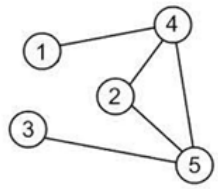
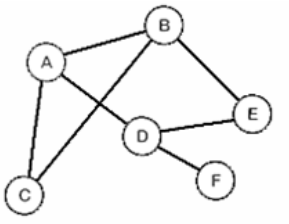
2	Explain difference between priori analysis and posteriori analysis.	Remember	Learner to recallasymptotic notationsand Compare priori analysis and posteriori analysis	CO 1
3	Explainbinary search algorithm and analyze its time complexity.	Remember	Learner to recall concept of divide and conquer and demonstrate binary search algorithm and analyze its time complexity	CO 2
4	Explain quick sort algorithm and simulate it for the following data: 20, 35, 10, 16, 54, 21, 25	Remember	Learner to recall concept of divide and conquer and demonstrate quick sort algorithm and analyze its time complexity	CO 2
5	Define iterative binary search algorithm.	Remember	—	CO 1
6	Illustrate merge sort algorithm and discuss time complexity in both worst case and average cases.	Remember	Learner to recallconceptof divide and conquer and demonstrate merge sort algorithm and analyze its time complexity, average and best cases.	CO 2
7	Explainthe advantage of Strassen's matrix multiplication when compared to normal matrix multiplication for the any two 16 x 16.	Remember	Learner to recall concept of Strassen's matrix multiplication and Comparewith normal matrix multiplication for the any two 16 x 16	CO 2
8	Explain amortized analysis and discuss how amortized complexity and actual complexity related.	Remember	Learner to recall concept of amortized analysis and outline how amortized complexity and actual complexity related.	CO 2
9	Defineprobabilistic analysis and randomized algorithms.	Remember	—	CO 1
10	Organizesorted list of numbers using merge sort: 78, 32, 42, 62, 98, 12.	Remember	Learner to recallthe concept of divide and conquer then understand merges sort method and solvegiven list of numbers.	CO 2
11	Define an algorithm that sorts a collection of greater than one elements of arbitrary type.	Remember	—	CO 1

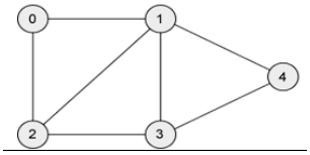
12	Explain the recurrence relation using substitution method T(n)= { T(1) n=1 aT(n/b)+f(n) n¿1 ,where a=5,b=4,and f(n)=cn2	Remember	Learner to recall concept of substitution method then explain the concept with the given example of recurrence relation	CO 1
13	Define the Pseudo code conventions for specifying algorithms of recursive and an iterative algorithm to compute n!.	Remember	—	CO 1
14	Explain the frequency counts for all statements in the following algorithm segment. i=1; while(i¡=n) do x=x+1; i=i+1;	Remember	Learner to recall concept of frequency countthen explain the concept with the given example algorithm	CO 2
15	What is stable sorting method? Is merge sort a stable sorting method? Justify	Remember	—	CO 2
16	What is Bubble sorting method? Is bubble sort a stable sorting method? Justify	Remember	—	CO 2
17	What is Quick sort? Is quick sort a best sorting method? Justify	Remember	—	CO 2
18	What is Bubble sorting method? Is bubble sort a stable sorting method? Justify	Understand	—	CO 2
19	Compare different asymptotic notations	Apply	—	CO 2
20	Differentiate time and space complexity? Justify	Apply	—	CO 2
PART-C - SHORT ANSWER QUESTIONS				
1	Define the term algorithm	Remember	—	CO 1
2	Define order of an algorithm.	Remember	—	CO 1
3	List asymptotic notations for big ‘Oh’, omega and theta.	Remember	—	CO 1
4	What do you mean by probability analysis?	Remember	—	CO 1

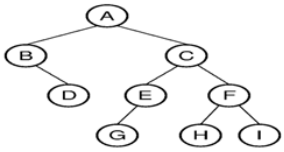
5	Findthe best case and worst case analysis for linear search.	Remember	—	CO 1
6	If f(n)=5n2 + 6n + 4, then Find that f(n) is O(n2).	Remember	—	CO 1
7	Define the recurrence equation for the worst case behavior of merge sort.	Remember	—	CO 1
8	Find the average case time complexity of quick sort.	Remember	—	CO 1
9	Define algorithm correctness.	Remember	—	CO 1
10	Define best case, average case and worst case efficiency of an algorithm.	Remember	—	CO 1
11	Define the term amortized efficiency.	Remember	—	CO 1
12	Define order of growth.	Remember	—	CO 1
13	How do you measure the runtime of an algorithm?	Remember	—	CO 1
14	Define the role of space complexity and time complexity of a program.	Remember	—	CO 1
15	What is the use of design technique?	Remember	—	CO 1
16	Explain step count method and analyze the time complexity when two n×n matrices are added.	Remember	Recallstep count method and summarize time complexity of given matrix.	CO 1
17	What is meant by divide and conquer? Give the recurrence relation for divide and conquer.	Remember	—	CO 1
18	Explaincontrol abstraction of divide and conquer.	Remember	This would require learner to recall the concepts of divide and conquer and demonstrate control abstraction of divide and conquer.	CO 1
19	Find out any two drawbacks of binary search algorithm.	Remember	—	CO 1
20	Find out the drawbacks of Merge Sort algorithm..	Remember	—	CO 1

MODULE II				
SEARCHING AND TRAVERSAL TECHNIQUES				
PART A-PROBLEM SOLVING AND CRITICAL THINKING				
Q.No	**QUESTION**	**Taxonomy**	**How does this subsume the level**	**CO's**
1	Build BFS traversal tree of following graph. 	Apply	Learner to recallthe concept of graph traversal thenunderstand breath first search technique and construct spanning tree for a given graph.	CO 4
2	Indentify the articulation points from the following graph 	Apply	Learner to recall the concept of connected and biconnetedcomponents thenunderstand methodology of articulation point andIdentifyarticulation points for the given graph.	CO 4
3	Organizeinorder, pre order, post order traversal of the following tree 	Apply	Learner to recalltree traversal techniques then understand inorder, preorder and postorder concepts and Organize the given tree.	CO 6
4	Construct DFS and BFS traversal trees of following graph 	Apply	Learner to recallthe concept of graph traversal technique then understand BFS and DFS technique and construct spanning trees for a given graph.	CO 6

5	Construct DFS traversal tree for following graph 	Understand	Learner to recallthe concept of graph traversal technique then understand DFS technique and construct spanning trees for a given graph.	CO 4
6	Construct BFS traversal tree of the following graph 	Apply	Learner to recallthe concept of graph traversal technique then understandBFS techniqueand construct spanning trees for a given graph.	CO 4
7	Identify the articulation points from the following graph 	Apply	Learner to recall concept of connected and biconnetedcomponents thenunderstandmethodology of articulation point andIdentifyarticulation points for the given graph.	CO 4
8	Constructinorder, preorder, post order traversal of the following tree. 	Apply	Learner to recall idea of tree traversal techniques then understand inorder, preorder and postorder concepts and constructthe given tree.	CO 4

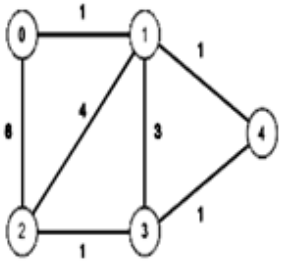
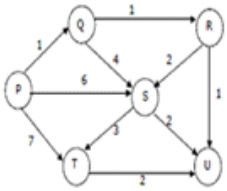
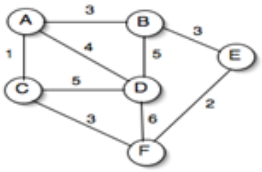
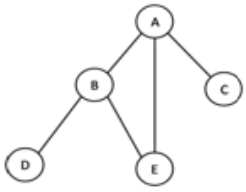
9	Construct BFS and DFS traversal treesof following graph. 	Apply	Learner to recallthe concept of graph traversal technique then understand BFS and DFS technique and construct spanning trees for a given graph	CO 4
10	Construct DFS traversal of following graph 	Apply	Learner to recallthe concept of graph traversal then understand DFStechniqueand construct spanning trees for a given graph.	CO 4
PART B-LONG ANSWER QUESTIONS				
1	Explain breadth first search algorithm with example.	Understand	Learner to recall the concept of breadth first search algorithm and Explain with the help of example.	CO 4
2	Explain depth first search algorithm with example	Understand	Learner to recall the concept of depth first search algorithm and Explain with the help of example	CO 4
3	Explain iterative versions of binary tree traversal algorithms (inorder, preorder and post order).	Understand	Learner to recall the concept of tree traversal algorithms and Contrast inorder, preorder and post order for iterative versions of binary tree	CO 4
4	Compare the approaches of BFS and DFS methods and derive the time complexities of both methods for the inputs of adjacency lists and adjacency matrix separately.	Apply	Learner to recall the concept of graph traversal algorithms thenExplain DFS, BFS methodsand Solve the time complexities of both methods for the inputs of adjacency lists and adjacency matrix separately.	CO 4

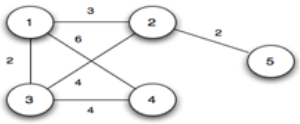
5	Explain BFS and spanning trees in detail.	Understand	Learner to recallthe concept of spanning trees, graph traversal and demonstrate breadth first search technique.	CO 4
6	Explain weighting rule for finding UNION of sets and collapsing rule	Understand	Learner to recall the concept of disjoin set operations and demonstrate weighting rule for finding UNION of sets and collapsing rule	CO 6
7	How to construct a binary tree from inorder and preorder traversals.	Remember	—	CO 6
8	Explain about DFS and spanning trees	Understand	Learner to recall the concept of spanning trees, graph traversal and demonstrate (Relate to)depth first search technique	CO 6
9	Illustrate how to identify given graph is connected or not	Understand	Learner to recallthe concept of connected components and outline the given graph is connected or not	CO 5
10	Explain the concept of biconnected component with an example	Understand	Learner to recall the concept of bi connected component and Explain with an example.	CO 5
11	Develop a program to print all the nodes reachable from a given starting node in a bigraph using BFS method. 	Apply	Learner to recall the concept of graph traversals then understand the BFS technique and developa program to print all the nodes reachable from a given starting node in a bigraph.	CO 6

12	Develop a program to perform various tree traversal algorithms for a given tree. 	Apply	Learner to recall the concept of tree traversals then understands inoder, preoder and postoder concepts and developsa program for a given tree.	CO 6
13	Construct binary tree from the following Inordersequence:DBEAFC and Preorder sequence: A B D E C F	Apply	Learner to recallthe concept of tree traversal technique then understandbinary tree traversal and outlinethe binary tree from the given inoder and preorder sequence.	CO 4
14	Illustrate the advantage of collapse find over simple find with example.	Understand	Learner to recall the concept of disjoin set operations and explain advantage of collapse find over simple find with example	CO 4
15	Construct binary tree from the following Inorder sequence: 4, 8, 2, 5, 1, 6, 3, 7 and Postorder sequence: 8, 4, 5, 2, 6, 7, 3, 1	Apply	Learner to recallthe concept of tree traversal thenunderstand binary tree traversal technique and outline the binary tree from the given inoder and postorder sequence.	CO 4
16	Explain step count method and analyze the time complexity when two n×n matrices are added.	Remember	Recallstep count method and summarize time complexity of given matrix.	CO 1
17	What is meant by divide and conquer? Give the recurrence relation for divide and conquer.	Remember	—	CO 1
18	Explaincontrol abstraction of divide and conquer.	Remember	This would require learner to recall the concepts of divide and conquer and demonstrate control abstraction of divide and conquer.	CO 1

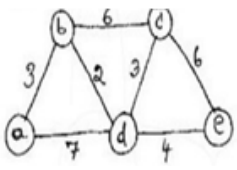
19	Find out any two drawbacks of binary search algorithm.	Remember	—	CO 1
20	Find out the drawbacks of Merge Sort algorithm..	Remember	—	CO 1
PART C-SHORT ANSWER QUESTIONS				
1	Define union operation on sets.	Remember	—	CO 2
2	Define find operation on sets.	Remember	—	CO 2
3	Define spanning tree and minimal cost spanning tree.	Remember	—	CO 2
4	Find time complexities of depth first search for the inputs of adjacency list and adjacency matrix.	Understand	Learner to recallthe concept of depth first search and explain thetime complexities.	CO 2
5	Find time complexities of breadth first search for the inputs of adjacency list and adjacency matrix.	Understand	Learner to recallthe concept of breadth first search and explain thetime complexities.	CO 4
6	Comparebreadth first search and depth first search.	Understand	Learner to recallthe concept of breadth first search, depth first searchandCompare theirproperties.	CO 4
7	What do you mean by weighted union?	Remember	—	CO 4
8	What is collapsing find?	Remember	—	CO 4
9	Define an articulation point	Remember	—	CO 4
10	Define connected component.	Remember	—	CO 5
11	Define bi-connected component.	Remember	—	CO 4
12	Compare connected and disconnected graphs.	Understand	Learner to recallthe concept of connected and disconnected graphsand Compare their properties.	CO 5
13	Which data structures are used for implementing the breadth first search and depth first search?	Remember	—	CO 5
14	List the binary tree traversal techniques.	Remember	—	CO 5
15	What is the use of design technique?	Remember	—	CO 1

16	Explain step count method and analyze the time complexity when two n×n matrices are added.	Remember	Recallstep count method and summarize time complexity of given matrix.	CO 1
17	What is meant by divide and conquer? Give the recurrence relation for divide and conquer.	Remember	—	CO 1
18	Explaincontrol abstraction of divide and conquer.	Remember	This would require learner to recall the concepts of divide and conquer and demonstrate control abstraction of divide and conquer.	CO 1
19	Find out any two drawbacks of binary search algorithm.	Remember	—	CO 1
20	Find out the drawbacks of Merge Sort algorithm..	Remember	—	CO 1
MODULE III				
GREEDY METHOD AND DYNAMIC PROGRAMMING				
PART A- PROBLEM SOLVING AND CRITICAL THINKING				
Q.No	**QUESTION**	**Taxonomy**	**How does this subsume the level**	**CO's**
1	Identifythe optimal solution for job sequencing with deadlines using greedy method. N=4, profits (p1,p2 ,p3,p4) = (100,10, 15,27), Deadlines (d1,d2, d3,d4) = (2,1,2,1)	Apply	Learner to recallthe concept of greedy method then understandjob sequencing with deadline problem and solve the given problem.	CO 4
2	Identifythe optimal solution for knapsack problem using greedy method N=3, M= 20, (p1,p2,p3)= (25,24,15), (w1,w2,w3) =(18,15,10)	Apply	Learner to recallthe concept of greedy method then understandknapsack problem andsolve the given problem.	CO 4

3	Construct minimum cost spanning tree using Prims algorithm 	Apply	Learner to recallthe concept of Prims algorithm understandminimum cost spanning treeand construct spanning tree for a given graph.	CO 6
4	Apply single source shortest path algorithm for the following graph 	Apply	Learner to recallgreedy method then understand single source shortest path method and identify the shortest path for a given graph.	CO 6
5	Construct Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm. 	Understand	Learner to recallthe concept of Kruskal's algorithm understandminimum cost spanning treemethod and construct spanning tree for a given graph	CO 4
6	Identify whether a given graph is connected or not using DFS method. 	Apply	Learner to recall the concept of connected components then understand DFS method and identifywhether a given graph is connected or not	CO 4

7	Construct Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.  	Apply	Learner to recallthe concept of prims's algorithm understandminimum cost spanning tree methodand construct spanning tree for a given graph	CO 4
8	Applyoptimal binary search tree algorithm and compute wij, cij, rij, 0¡=i¡=j¡=4,p1=1/10, p2=1/5, p3=1/10, p4=1/120, q0=1/5, q1=1/10, q2=1/5, q3=1/20,q4=1/20.	Apply	Learner to recall the concept of dynamic programming then understand optimal binary search tree algorithm and solve the given example.	CO 4
9	Construct optimal binary search tree for (a1, a2, a3, a4) = (do, if,int, while), p(1 : 4) = (3,3,1,1) q(0 : 4)= (2,3,1,1,1)	Apply	Learner to recall the concept of dynamic programming then understand optimal binary search tree algorithm and construct tree for the given data.	CO 4
10	Solve the solution for 0/1 knapsack problem using dynamic programming(p1,p2,p3, p4) = (11, 21, 31, 33), (w1, w2, w3, w4) = (2, 11, 22, 15), M=40, n=4	Apply	Learner to recall the concept of dynamic programming then understand 0/1 knapsack problem and solve given example problem.	CO 6
PART B-LONG ANSWER QUESTIONS				
1	Explain job sequencing with deadlines problem and write the algorithm.	Apply	Learner to recall the concept of greedy method then explain job sequencing with deadlines and write the algorithm.	CO 5
2	Explain single source shortest path problem with example using greedy method.	Apply	Learner to recall the concept of greedy method then explain single source shortest path problem with example	CO 5

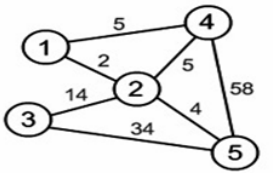
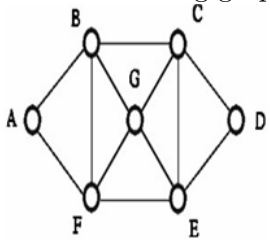
3	Explain the knapsack problem with suitable example.	Apply	Learner to recall the concept of greedy method then explain the knapsack problem with example	CO 5
4	Explain Prim's algorithm with example.	Apply	Learner to recall the concept of prim's algorithm then illustrate with an example	CO 5
5	Explainkruskal's algorithm with example	Understand	Learner to recall the concept of krushkal's algorithm then illustrate with an example	CO 5
6	Showthat Prim's method generates minimum-cost spanning tree.	Apply	Learner to recall the concept of prim's algorithm then Show thatPrim's method generates minimum-cost spanning tree	CO 5
7	Explain control abstraction of greedy method, how it is useful for real time problems.	Apply	Learner to recall the concept greedy method then Explain control abstraction of greedy method and how it is useful for real time problems	CO 5
8	DemonstrateBellman Ford algorithm to compute shortest path.	Apply	Learner to recallthe concept of dynamic programming then understand Bellman and Ford algorithm tocomputeshortest path.	CO 5
9	Explain the greedy method for generating the shortest paths	Apply	Learner to recall the concept of greedy method then Outlinethe shortest paths	CO 5
10	Explain the time complexities of Prim's and Kruskal's algorithms.	Apply	Learner to recall the concept of prim's and kruskal's algorithms then explain time complexities of both algorithms.	CO 5
11	Compare Prim's and Kruskal's algorithms.	Apply	Learner to recall the concept of prim's and kruskal's algorithms then compare both algorithms	CO 5

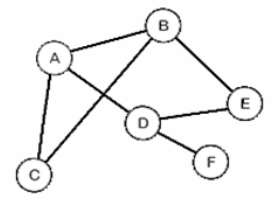
12	Make use of Prim's algorithm to find minimum cost spanning tree for a graph G(6,10) with vertices named as a,b,c,d,e,f and edges ab=3,bc=1, af=5,ae=6,ed=8, fe=2,fd=5, cd=6,cf=4 and bf=4 by showing results in each stages.	Apply	Learner to recall the concept of spanning tree then understand prim's algorithm and solve minimum cost spanning tree for a given graph.	CO 5
13	Make use of the control abstraction for subset paradigm using greedy method. Solve the job sequencing with deadline problem using greedy method for the given data N=7,P=3,5,20,18,1,6,30 are profits and D=1,3,4,3,5,1,2 are deadline respectively.	Apply	Learner to recallthe concept of greedy method then understandjob sequencing with deadline problem and solve the given problem.	CO 5
14	Indentifyminimum cost spanning tree for a graph G(6,10)with vertices named as a,b,c,d,e,f and edges ab=1,bc=3, af=9,ae=4,ed=6, fe=4,fd=5,cd=6,cf=4 and bf=4 using Kruskal's algorithm and showingresults in each stages.	Apply	Learner to recall the concept of spanning tree then understandprim's algorithm and solveminimum cost spanning tree for a given graph.	CO 5
15	Identify the shortest path from source a to all other vertices in the graph shown in below Fig. Using greedy method .Give the greedy criterion used. 	Apply	Learner to recallgreedy method then understand single sourse shortest path method and identify the shortest path for a given graph	CO 5

16	Explain optimal binary search tree algorithm with example	Apply	Learner to recall the concept of dynamic programming then explainoptimal binary search tree algorithm with example	CO 5
17	Explain 0/1 knapsack problem with example.	Apply	Learner to recall the concept of dynamic programming then explain0/1 knapsack problem with example	CO 5
18	Explain all pairs shortest path problem with example	Apply	Learner to recall the concept of dynamic programming then explainall pairs shortest path problem with example	CO 5
19	Explain the travelling salesman problem and discuss how to solve it using dynamic programming?	Apply	Learner to recall the concept of dynamic programming then explain the travelling salesman problem.	CO 5
20	Explain matrix chain multiplication with example.	Understand	Learner to recall the concept of dynamic programming then explain matrix chain multiplication with example	CO 5
PART C - SHORT ANSWER QUESTIONS				
1	Define dynamic programming.	Apply	Define dynamic programming.	CO 5
2	Definethe principle of optimality.	Apply	Definethe principle of optimality.	CO 5
3	List the features of dynamic programming	Apply	List the features of dynamic programming	CO 5
4	Compare greedy method and dynamic programming	Apply	Compare greedy method and dynamic programming	CO 5
5	Find the formula for computing cost of binary search tree.	Understand	Find the formula for computing cost of binary search tree.	CO 5
6	Findthe number of possible binary search trees with 3 identifiers.	Apply	Findthe number of possible binary search trees with 3 identifiers.	CO 5

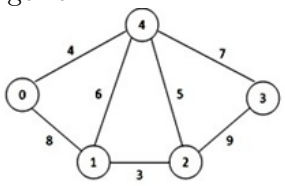
7	Findthe time complexity of travelling salesperson problem using dynamic programming.	Apply	Findthe time complexity of travelling salesperson problem using dynamic programming.	CO 5
8	List the applications of traveling sales person problem.	Apply	List the applications of traveling sales person problem.	CO 5
9	Define minimum cost spanning tree.	Apply	Define minimum cost spanning tree.	CO 5
10	Find the time complexity of all pairs shortest paths problem.	Apply	Find the time complexity of all pairs shortest paths problem.	CO 5
11	Explain an approach of dynamic programming.	Apply	Explain an approach of dynamic programming.	CO 5
12	Define 0/1 knapsack problem.	Apply	Define 0/1 knapsack problem.	CO 5
13	Explain advantages of travelling salesperson problem.	Apply	Explain advantages of travelling salesperson problem.	CO 5
14	Define matrix chain multiplication problem.	Apply	Define matrix chain multiplication problem.	CO 5
15	List the applications of dynamic programming.	Apply	List the applications of dynamic programming.	CO 5
16	Define greedy method.	Apply	Define dynamic programming.	CO 5
17	Define job sequencing with deadlines problem.	Apply	Definethe principle of optimality.	CO 5
18	Define minimum cost spanning tree.	Apply	List the features of dynamic programming	CO 5
19	Explain importance of prim's algorithm.	Apply	Compare greedy method and dynamic programming	CO 5
20	Explainimportance of kruskal's algorithm.	Understand	Find the formula for computing cost of binary search tree.	CO 5

MODULE IV				
BACKTRACKING AND BRANCH AND BOUND				
PART A-PROBLEM SOLVING AND CRITICAL THINKING				
Q.No	**QUESTION**	**Taxonomy**	**How does this subsume the level**	**CO's**
1	Build the state space tree generated by 4 queen's problem.	Apply	Learner to recall the concept of back tracking then understand4 queens problem and build the state space tree.	CO 5
2	Apply the backtracking algorithm to solve the following instance of the sum of subsets problem S=5,10,12,13,15,18 and d=30	Apply	Learner to recall the concept of back tracking then understand the sum of subsets problemand solve given example problem.	CO 5
3	Build the state space tree generated all possible 3-color,4-node graph	Apply	Learner to recall the concept of back tracking thenunderstandgraph coloring method and Build the state space tree for a given graph.	CO 5
4	Identify Hamiltonian cycle from the following graph	Apply	Learner to recall the concept of back tracking then understand Hamiltonian cycle method and Identify Hamiltonian cycle from the given graph.	CO 5
5	Solve the following instance of travelling sales person problem using Least Cost Branch and Bound $\begin{bmatrix} \infty & 12 & 5 & 7 \\ 11 & \infty & 13 & 6 \\ 4 & 9 & \infty & 18 \\ 10 & 3 & 2 & \infty \end{bmatrix}$	Understand	Learner to recall the concept of backtracking then understandLeast Cost Branch and Bound method and solvegiventravelling sales person problem.	CO 5

6	Build the state space tree generated by LCBB by the following knapsack problem n=5 , (p1,p2,p3,p4,p5) =(10,15,6, 8, 4), (w1,w2,w3,w4,w5)=(4,6,3,4,2) and m=12	Apply	Learner to recall the concept of Branch and Bound then understandLeast Cost Branch and Bound method and Build the state space tree for a given knapsack problem.	CO 5
7	Build the state space tree generated by FIFO knapsack for the instance N=4 , (P1, P2, P3, P4)= (10, 10, 12, 18) , (w1, w2,w3,w4) = (2, 4, 6, 9) , m=15	Apply	Learner to recall the concept of Branch and Bound then understandfirst in first out Branch and Bound method and Build the state space tree for a given knapsack problem.	CO 5
8	Solve the following instance of travelling sales person problem using Least Cost Branch Bound 	Apply	Learner to recall the concept of Branch and Bound then understandLeast Cost Branch Bound and solve the given travelling sales person problem.	CO 5
9	Identify Hamiltonian cycle from the following graph 	Apply	Learner to recall the concept of back tracking then understand Hamiltonian cycle method and Identify Hamiltonian cycle from the given graph.	CO 5

10	Apply the backtracking algorithm to find chromatic number for the following graph 	Apply	Learner to recall the concept of back tracking then understandgraph coloring method and identify chromatic number for the given graph.	CO 5
PART B- LONG ANSWER QUESTIONS				
1	Develop an algorithm for N-queens problem using backtracking.	Apply	Learner to recall the concept of backtrackingthen understand N-queens problem and develop an algorithm for it.	CO 7
2	Applybacktracking method and solve subset-sum problem and discuss the possible solution strategies.	Apply	Learner to recall the concept of backtrackingthen understand subset-sum problem and solvepossible solution strategies .	CO 7
3	Apply graph coloring technique and write an algorithm for m-coloring problem.	Apply	Learner to recall the concept of backtrackingthen understand graph coloring problem and solve m-coloring problem and write an algorithm for it.	CO 7
4	Explainan algorithm for Hamiltonian cycle with an example.	Apply	Learner to recall the concept of backtrackingthen explainHamiltonian cycle with an example.	CO 7
5	Explain properties of LC search.	Understand	Learner to recall the concept of branch and bound then explainproperties of LC search.	CO 7
6	Explaincontrol abstraction for LC Search.	Apply	Learner to recall the concept of branch and bound then explaincontrol abstraction for LC Search.	CO 7

7	Explain principle of FIFO branch and bound.	Apply	Learner to recall the concept of branch and bound then explainprinciple of FIFO branch and bound.	CO 7
8	Explain principle of LIFO branch and bound.	Apply	Learner to recall the concept of branch and bound then explainprinciple of LIFO branch and bound.	CO 7
9	Apply the method of reduction to solve travelling sales person problem using branch and bound.	Apply	Learner to recall the concept of branch and bound then understand method ofreduction solvetravelling sales person problem.	CO 7
10	Explain TSP using branch and bound method with example	Apply	Learner to recall the concept of branch and bound then understand method oftravelling sales person problem and solveitwith an example.	CO 7
11	Explain the basic principle of Backtracking and list the applications of Backtracking.	Apply	Learner to recall the concept of back tracking then demonstratethe basic principle and applications of backtracking.	CO 7
12	Explainbacktracking technique and solve the following instance for the subset problem s=(1,3,4,5) and d=11.	Apply	Learner to recall the concept of back tracking and understand sum of subset problem and solve given example problem.	CO 7
13	Construct the portion of the state space tree generated by LCBB for the knapsack instance: n=5,(p1,p2,p3,p4,p5)=(w1,w2,w3,w4,w5)=(4,4,5,8,9), and m=15.	Apply	Learner to recall the concept of branch and bound then understandLCBB technique 4,5,8,9)construct the state space tree for a given example.	CO 7
14	Explain an algorithm for 4-queens problem using backtracking	Apply	Learner to recall the concept of back tracking thendemonstrate4-queens problem.	CO 7

15	Applybacktracking technique solve the following instance for the subset problem s=(6,5,3,7) and d=15.	Apply	Learner to recall the concept of back tracking then understand subset problemand solve the given example.	CO 7
16	Build the portion of state space tree generated by FIFOBB for the job sequencing with deadlines instance n=5 , (p1,p2,..,p5) =(6,3,4,8,5), (t1,t2,..t5) = (2,1,2,1,1) and (d1,d2,..,d5)=(3,1,4,2,4).What is the penalty corresponding to an optimal solution.	Apply	Learner to recall the concept of branch and bound then understandFIFOBB technique and construct the state space tree for a given example.	CO 7
17	Solve the solution for 0/1 knapsack problem using dynamic programming N=3 , m=6 profits (p1,p2,p3) = (1,2,5) weights (w1,w2,w3) = (2,3,4)	Apply	Learner to recall the concept of dynamic programming then understand 0/1 knapsack problem and solve given example problem.	CO 6
18	Choose shortest distances using all pairs shortest path algorithm 	Understand	Learner to recall the concept of dynamic programming then understandall pairs shortest path algorithm and chooseshortest distances for a given graph.	CO 4
19	Solve knapsack problem by Dynamic Programming method n=6, (p1, p2,...p6)= (w1,w2,...w6)= (100,50,20,10,7,3) and m=165.	Apply	Learner to recall the concept of dynamic programming then understands the knapsack problem and solves the given problem.	CO 4
20	Define greedy method.	Apply	Define dynamic programming.	CO 5
PART C- SHORT ANSWER QUESTIONS				
1	Definethe principle of Backtracking.	Apply	—	CO 7
2	Definecontrol abstraction for backtracking.	Apply	—	CO 7

3	List the applications of backtracking.	Apply	—	CO 7
4	Define a dead node.	Apply	—	CO 7
5	Define live node and dead node.	Understand	—	CO 7
6	Define state space tree.	Apply	—	CO 7
7	Define solution space.	Apply	—	CO 7
8	Define solution states and answer state.	Apply	—	CO 7
9	Define 8 – Queens problem.	Apply	—	CO 7
10	DefineSum of Subsets problem.	Apply	—	CO 7
11	Definegraph coloring.	Apply	—	CO 7
12	DefineHamilton cycle.	Apply	—	CO 7
13	Defineprinciple of branch and bound.	Apply	—	CO 7
14	Defineleast cost branch and bound.	Apply	—	CO 7
15	Definefirst in first out branch and bound.	Apply	—	CO 7
16	What is the efficiency of Warshall's algorithm?	Apply	—	CO 7
17	Explain how P and NP problems are related.	Apply	—	CO 8
18	CompareNP- hard and NP-complete problems	Apply	—	CO 8
19	What is the efficiency of Warshall's algorithm?	Apply	—	CO 7
20	Definethe principle of Backtracking.	Apply	—	CO 7
MODULE V				
NP-HARD AND NP-COMPLETE PROBLEM				
PART A-PROBLEM SOLVING AND CRITICAL THINKING				
Q.No	**QUESTION**	**Taxonomy**	**How does this subsume the level**	**CO's**
1	Show that satisfiability is at most three literals reduces to chromatic number.	Apply	Learner to recall the concept of NP-hard and NP-complete then Show that satisfiability is at most three literals reduces to chromatic number.	CO 7

2	Show that Hamiltonian cycle is in NP.	Apply	Learner to recall the concept of NP-hard and NP-complete then Show that Hamiltonian cycle is in NP.	CO 7
3	Showthat circuit-SAT is in NP.	Apply	Learner to recall the concept of NP-hard and NP-complete then Show thatcircuit-SAT is in NP.	CO 7
4	Explaintwo problems that have polynomial time algorithms justify your answer.	Apply	Learner to recallthe concept non-deterministic and deterministico algorithm then explain two problems that have polynomial time algorithms.	CO 7
5	Explain 3CNF satisfiability problem.	Understand	Learner to recallthe concept non-deterministic and deterministico algorithm then explain 3CNF satisfiability problem	CO 7
6	Explain P type problems with examples	Apply	Learner to recall the concept of NP-hard and NP-complete then Explain P type problems with examples.	CO 7
7	Explain in detail about Approximation Algorithms for NP hard problems.	Apply	Learner to recall the concept of NP-hard and NP-complete then Explainabout Approximation Algorithms.	CO 7
8	Show that satisfiability of Boolean formula in 3 conjective normal form is NP-complete.	Apply	Learner to recall the concept of NP- complete then Explainsatisfiability of Boolean formula in 3 conjective normal form	CO 7
9	Showthat Clique Decision problem is NP-Complete	Apply	Learner to recall the concept of NP-complete then show that Clique decision problem is NP-Complete.	CO 7

10	Show that Chromatic number decision problem is NP-Complete.	Apply	Learner to recall the concept of NP-complete then show that Chromatic number decision problem is NP-Complete.	CO 7
PART B- LONG ANSWER QUESTIONS				
1	Explain and prove Cook's theorem	Apply	Learner to recallthe concept of non-deterministic algorithm then demonstrate the cook's theorem.	CO 8
2	Compare deterministic and non-deterministic algorithms	Apply	Learner to recallthe concept non-deterministic and deterministico algorithm then compare both the algorithms.	CO 8
3	Explainnon deterministic algorithm for sorting and searching.	Apply	Learner to recallthe concept of non-deterministic algorithm then demonstrate non deterministic algorithm for sorting and searching.	CO 8
4	Explainnon deterministic algorithm for sorting non-deterministic knapsack algorithm.	Apply	Learner to recallthe concept of non-deterministic algorithm then demonstrate non-deterministic knapsack algorithm.	CO 8
5	Explain how P and NP problems are related.	Understand	Learner to recall and relatehow P and NP problems are related.	CO 8
6	CompareNP- hard and NP-complete problems	Apply	Learner to recallthe concepts of NP-hard and NP-complete then compare both classes.	CO 8
7	Explain clique decision problem with an example	Apply	Learner to recallthe concepts of NP-hard and NP-complete then explain clique decision problem with an example	CO 8
8	Explain chromatic number decision problem and clique decision problem	Apply	Learner to recallthe concepts of NP-hard and NP-complete then explain clique decision and chromatic number decision problem.	CO 8

9	Explain the strategy to prove that a problem is NP-hard.	Apply	Learner to recallthe concepts of NP-hard and NP-complete then explain the strategy to prove that a problem is NP-hard.	CO 8
10	Explain intractable problems with examples	Apply	Learner to recallthe concepts of NP-hard and NP-complete then explain intractable problems with examples.	CO 8
11	Explain and prove Cook's theorem	Apply	—	CO 8
12	Compare deterministic and non-deterministic algorithms	Apply	—	CO 8
13	Explainnon deterministic algorithm for sorting and searching.	Apply	—	CO 8
14	Explainnon deterministic algorithm for sorting non-deterministic knapsack algorithm.	Apply	—	CO 8
15	Explain how P and NP problems are related.	Apply	—	CO 8
16	CompareNP- hard and NP-complete problems	Apply	—	CO 8
17	What is the efficiency of Warshall's algorithm?	Apply	—	CO 7
18	Definethe principle of Backtracking.	Apply	—	CO 7
19	Definecontrol abstraction for backtracking.	Apply	—	CO 7
20	List the applications of backtracking.	Apply	—	CO 7
PART C- SHORT ANSWER QUESTIONS				
1	Define class P	Apply	—	CO 8
2	Compare NP-hard and NP-completeness	Apply	—	CO 8
3	Define NP- hard problem	Apply	—	CO 8
4	Define NP-complete problem	Apply	—	CO 8
5	Define Deterministic problem?	Understand	—	CO 8

6	Define Non-deterministic problem	Apply	—	CO 8
7	Define a decision problem?	Apply	—	CO 8
8	Define Optimization problem	Apply	—	CO 8
9	Defineclique problem?	Apply	—	CO 8
10	Define Halting problem	Apply	—	CO 8
11	Define vertex cover problem.	Apply	—	CO 8
12	Definedeterministic algorithm.	Apply	—	CO 8
13	Definechromatic number.	Apply	—	CO 8
14	Definecook's theorem.	Apply	—	CO 8
15	Definechromatic number.	Apply	—	CO 8
16	CompareNP- hard and NP-complete problems	Apply	—	CO 8
17	DefineHamilton cycle.	Apply	—	CO 7
18	Defineprinciple of branch and bound.	Apply	—	CO 7
19	Defineleast cost branch and bound.	Apply	—	CO 7
20	Definefirst in first out branch and bound.	Apply	—	CO 7

Course Coordinator: **HOD CSE**
Ms. C Bhagyashree, Assistant professor