INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad -500 043

INFORMATION TECHNOLOGY

TUTORIAL QUESTION BANK

Course Name	:	**DESIGN AND ANALYSIS OF ALGORITHMS**
Course Code	:	AIT001
Class	:	B.Tech III Semester
Branch	:	Informaiotn Technology
Academic Year	:	2018 – 2019
Course Faculty	:	Dr. K Rajendra Prasad, Professor and Head Dr. R Obula konda Reddy, Professor Dr.G.Ramu, Professor Dr.B.V. Rao, Professor Mr. Ch.Suresh Kumar Raju, Assistant Professor Ms.K.Radhika, Assistant Professor

COURSE OVERVIEW:

The primary objective of this course is to introduce the concept of algorithm as a precise mathematical concept, and study how to design algorithms, establish their correctness, study their efficiency and memory needs. The course consists of a strong mathematical component in addition to the design of various algorithms.

COURSE OBJECTIVES:

The course should enable the students to:

I	Calculate performance of algorithms with respect to time and space complexity.
II	Illustrate the graph traversals and tree traversals to solve the problems.
III	Demonstrate the concepts greedy method and dynamic programming for several applications like knapsack problem, job sequencing with deadlines, and optimal binary search tree, TSP.
IV	Illustrating the methods of backtracking and branch bound techniques to solve the problems like n-queens problem, graph coloring and TSP respectively.
V	Understand the concept of deterministic and non-deterministic algorithms.

COURSE LEARNING OUTCOMES:

Students, who complete the course, will have demonstrated the ability to do the following:

CAIT001.01	Use big O-notation formally to give asymptotic upper bounds on time and space complexity of algorithms
CAIT001.02	Explain the use of big-Omega, big-Theta, and little-o notations to describe the amount of work done by an algorithm.
CAIT001.03	Use recurrence relations to determine the time complexity of recursive algorithms.
CAIT001.04	Evaluateandcomparedifferentalgorithmsusingworst,average,andbest-caseanalysis
CAIT001.05	Solve elementary recurrence relations, e.g., using some form of a Master Theorem. Give examples that illustrate time-space trade-offs of algorithms.
CAIT001.06	Demonstrate the ability to evaluate algorithms, to select from a range of possible options, to provide justification for that selection, and explain an implementation of the algorithm in a particular context.

CAIT001.07	Describe and use major algorithmic techniques (brute-force, greedy, divide-and-conquer, dynamic programming, and graph explorations).
CAIT001.08	Use a divide-and-conquer algorithm to solve an appropriate problem.
CAIT001.09	Use a greedy approach to solve an appropriate problem and determine if the greedy rule chosen leads to an optimal solution.
CAIT001.10	Use dynamic programming to develop the recurrence relations and to solve an appropriate problem.
CAIT001.11	Use recursive backtracking to solve a problem such as navigating a maze.
CAIT001.12	Explain the major graph algorithms and their analysis and employ graphs to model application problems.
CAIT001.13	Determine appropriate algorithmic approaches to apply to a given problem.
CAIT001.14	Describe heuristic problem-solving methods.
CAIT001.15	Understand the mapping of real-world problems to algorithmic solutions.
CAIT001.16	Define the classes P and NP.
CAIT001.17	Explain the significance of NP-completeness.
CAIT001.18	Provide examples of NP-complete problems.
CAIT001.19	Explain the impact of NP-complete problems to different application domains.
CAIT001.20	Explain the difference between NP-complete and NP-hard.
CAIT001.21	Prove that a problem is NP-complete.
CAIT001.22	Use reduction techniques between problems.
CAIT001.23	Demonstrate the use of approximation algorithms for NP-hard problems.
CAIT001.24	Explain the Halting problem and other undecidable problems.
CAIT001.25	Possess the knowledge and skills for employability and to succeed in national and international level competitive examinations.

TUTORIAL QUESTION BANK

UNIT – I			
PART – A (SHORT ANSWER QUESTIONS)			
S. No	**Question**	**Blooms Taxonomy Level**	**Course Learning Outcomes**
1	Define the term algorithm and state the criteria the algorithm should satisfy.	Remember	CAIT001.06
2	Define order of an algorithm and the need to analyze the algorithm.	Remember	CAIT001.01,
3	List asymptotic notations for big ‘Oh’, omega and theta?	Remember	CAIT001.01
4	What do you mean by probability analysis?	Remember	CAIT001.05
5	State the best case and worst case analysis for linear search	Understand	CAIT001.04
6	If $f(n)=5n^2 + 6n + 4$, then prove that $f(n)$ is $O(n^2)$	Understand	CAIT001.02
7	Give the recurrence equation for the worst case behavior of merge sort.	Remember	CAIT001.07, CAIT001.08
8	Compute the average case time complexity of quick sort	Remember	CAIT001.04
9	Define algorithm correctness	Remember	CAIT001.06
10	Describe best case, average case and worst case efficiency of an algorithm?	Understand	CAIT001.04
11	Explain the term amortized efficiency	Remember	CAIT001.05
12	Define order of growth	Understand	CAIT001.01
13	How do you measure the runtime of an algorithm?	Remember	CAIT001.02
14	Describe the role of space complexity and time complexity of a program.	Understand	CAIT001.05
15	What is the use of design technique?	Remember	CAIT001.06
16	Use step count method and analyze the time complexity when two n×n matrices are added	Understand	CAIT001.05

17	What is meant by divide and conquer? Give the recurrence relation for divide and conquer.	Remember	CAIT001.03
18	Define control abstraction of divide and conquer.	Understand	CAIT001.08
19	List out any two drawbacks of binary search algorithm.	Remember	CAIT001.08
20	List out the drawbacks of Merge Sort algorithm.	Remember	CAIT001.08
21	State the best, average and worst case complexities of binary search for successful and unsuccessful search	Understand	CAIT001.08

PART – B (LONG ANSWER QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Learning Outcomes
1	Discuss various the asymptotic notations used for best case average case and worst case analysis of algorithms.	Remember	CAIT001.04
2	Differentiate between priori analysis and posteriori analysis.	Understand	CAIT001.06
3	Discuss binary search algorithm and analyze its time complexity	Understand	CAIT001.08, CAIT001.03
4	Explain quick sort algorithm and simulate it for the following data: 20, 35, 10, 16, 54, 21, 25	Understand	CAIT001.07
5	Write and explain iterative binary search algorithm	Remember	CAIT001.08
6	Illustrate merge sort algorithm and discuss time complexity in both worst case and average cases.	Understand	CAIT001.08, CAIT001.03
7	Describe the advantage of Strassen's matrix multiplication when compared to normal matrix multiplication for the any two 16 x 16 matrices	Understand	CAIT001.05
8	Explain amortized analysis and discuss how amortized complexity and actual complexity related.	Understand	CAIT001.05
9	Discuss probabilistic analysis and randomized algorithms	Remember	CAIT001.06
10	Sort the list of numbers using merge sort: 78, 32, 42, 62, 98, 12, 34, 83	Understand	CAIT001.07
11	Devise an algorithm that sorts a collection of $n \geq 1$ elements of arbitrary type	Remember	CAIT001.07
12	Solve the recurrence relation using substitution method $T(n) = \{ T(1) \quad n=1$ $aT(n/b)+f(n) \quad n>1$,where a=5,b=4,and $f(n)=cn^2$	Understand	CAIT001.05
13	Describe the Pseudo code conventions for specifying algorithms of recursive and an iterative algorithm to compute n!	Remember	CAIT001.05

14	Determine the frequency counts for all statements in the following algorithm segment.
	i=1;
	while(i<=n) do
	{
	x=x+1;
	i=i+1;
	}
		Understand	CAIT001.05

15	What is stable sorting method? Is merge sort a stable sorting method? Justify	Remember	CAIT001.08

PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Learning Outcomes
1	Solve the following recurrence relation $T(n)=2\,T(n/2) + n$, and $T(1)=2$	Understand	CAIT001.05
2	Solve the following recurrence relation $T(n) = 7T(n/2)+cn^2$	Understand	CAIT001.05

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	Understand	CAIT001.05
4	Explain quicksort algorithm and simulate it for following data sequence: 3 5 9 7 1 4 6 8 2	Understand	CAIT001.07
5	Show the tracing steps of merge sort and quicksort and analyze the time complexity for the following data: 33, 44, 2, 10, 25, 79, 86, 47, 14, 36	Understand	CAIT001.08
6	Derive the average case time complexity of quick sort and merge sort methods	Remember	CAIT001.01
7	Use merge sort on following letters H, K, P,C,S,K,R,A,B,L	Understand	CAIT001.08
8	When Strassen's method outperforms the traditional matrix multiplication method. How many number of multiplication operations are required during multiplication of two matrices with size of 32 x 32 in Stressen's method.	Remember	CAIT001.05
9	Write and solve recurrence relation for Strassen's matrix multiplication	Understand	CAIT001.05
10	Solve the following recurrence relation T(n)=2 T(n/2) + 1, and T(1)=2	Understand	CAIT001.03

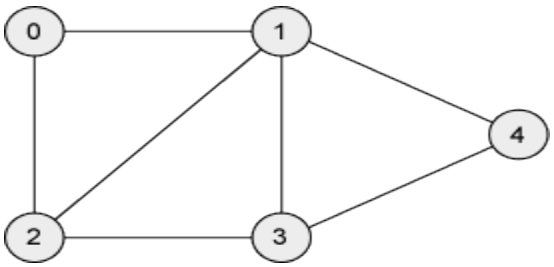
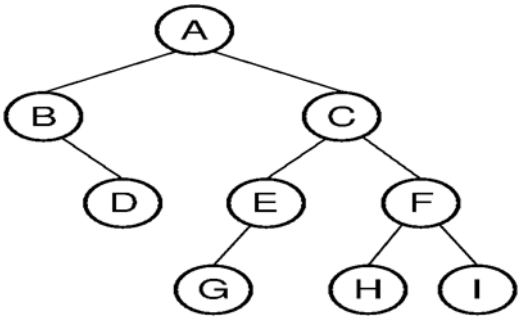
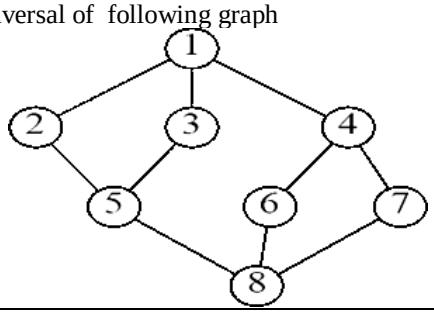
UNIT-II

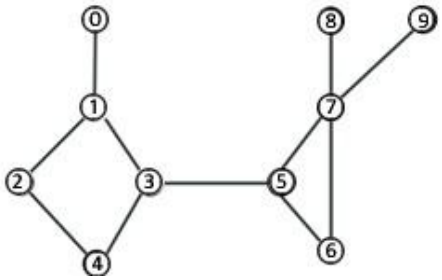
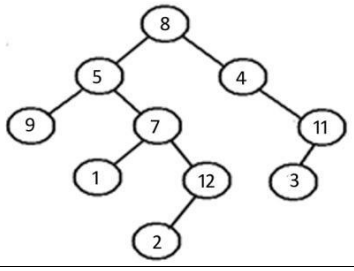
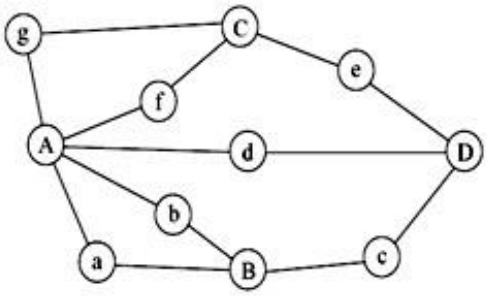
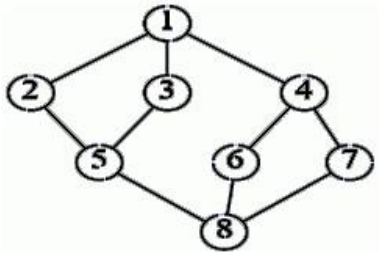
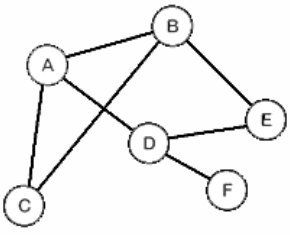
PART – A (SHORT ANSWER QUESTIONS)

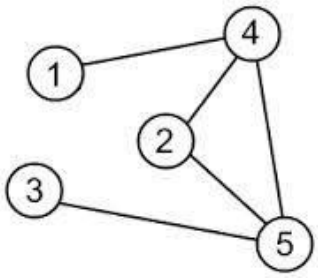
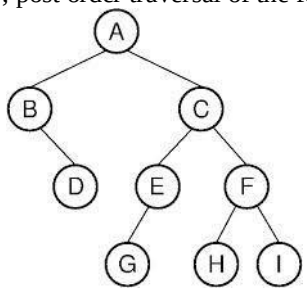
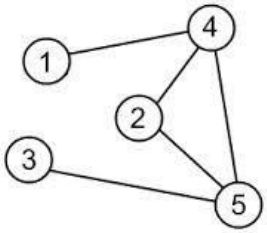
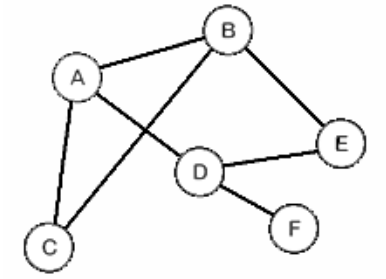
S. No	Question	Blooms Taxonomy Level	Course Learning Outcomes
1	Describe union operation on sets	Remember	CAIT001.14
2	Describe find operation on sets	Remember	CAIT001.14
3	Define spanning tree and minimal spanning tree	Remember	CAIT001.13
4	Write time complexities of depth first search for the inputs of adjacency list and adjacency matrix.	Understand	CAIT001.12
5	Write time complexities of breadth first search for the inputs of adjacency list and adjacency matrix	Understand	CAIT001.12
6	Differentiate breadth first search and depth first search	Understand	CAIT001.12
7	What do you mean by weighted union.	Remember	CAIT001.14
8	What is collapsing find?	Understand	CAIT001.13
9	Define an articulation point	Remember	CAIT001.12
10	Define connected component.	Remember	CAIT001.12
11	Define bi-connected component.	Remember	CAIT001.12
12	Differentiate connected and disconnected graphs	Understand	CAIT001.12
13	Which data structures are used for implementing the breadth first search and depth first search	Remember	CAIT001.12
14	List the binary tree traversal techniques.	Remember	CAIT001.12

PART – B (LONGANSWER QUESTIONS)

S .No	Question	Blooms Taxonomy Level	Course Learning Outcomes
1	Write and explain breadth first search algorithm with example.	Understand	CAIT001.12
2	Write and explain depth first search algorithm with example	Understand	CAIT001.12
3	Discuss iterative versions of binary tree traversal algorithms (inorder, preorder and post order).	Understand	CAIT001.13

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.	Remember	CAIT001.12
5	Describe BFS and spanning trees in detail.	Remember	CAIT001.12
6	Explain weighting rule for finding UNION of sets and collapsing rule	Remember	CAIT001.14
7	How to construct a binary tree from inorder and preorder traversals.	Remember	CAIT001.07 CAIT001.09
8	Discuss about DFS and spanning trees	Understand	CAIT001.12
9	Illustrate how to identify given graph is connected or not	Understand	CAIT001.12
10	Discuss the concept of biconnected component with an example	Understand	CAIT001.12
11	Write a program to print all the nodes reachable from a given starting node in a digraph using BFS method. 	Remember	CAIT001.12
12	Write a program to perform various tree traversal algorithms for a given tree. 	Remember	CAIT001.13
13	Construct binary tree from the following Inorder sequence: D B E A F C and Preorder sequence: A B D E C F	Understand	CAIT001.13
14	Illustrate the advantage of collapse find over simple find with example.	Remember	CAIT001.14
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	Understand	CAIT001.13
PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)			
1	Illustrate BFS traversal of following graph 	Understand	CAIT001.12

2	List the articulation points from the following graph 	Remember	CAIT001.14
3	Write inorder, pre order, post order traversal of the following tree 	Remember	CAIT001.13
4	Illustrate DFS and BFS traversals of following graph 	Understand	CAIT001.12
5	Illustrate DFS traversal of following graph 	Understand	CAIT001.12
6	Illustrate BFS traversal of the following graph 	Understand	CAIT001.14

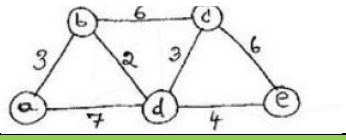
7	List the articulation points from the following graph 	Understand	CAIT001.07
8	Write inorder, preorder, post order traversal of the following tree 	Understand	CAIT001.12
9	Illustrate BFS and DFS traversals of following graph 	Understand	CAIT001.07
10	Illustrate DFS traversal of following graph 	Understand	CAIT001.13

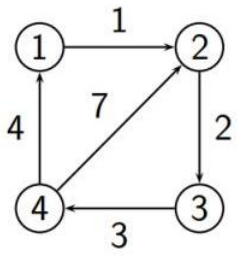
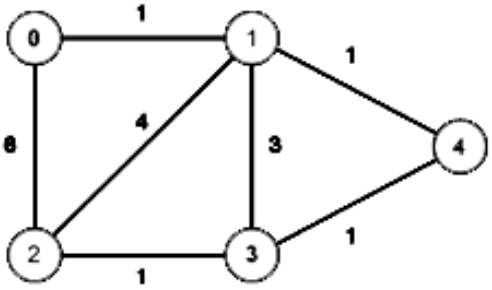
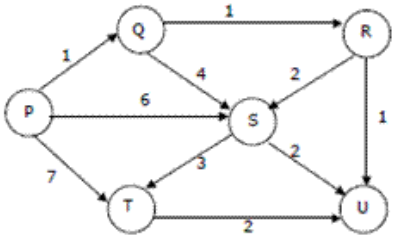
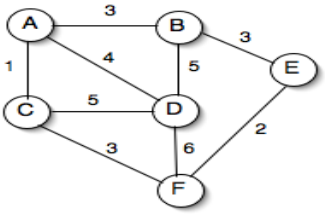
UNIT-III

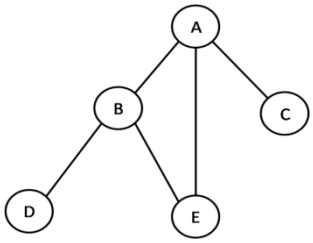
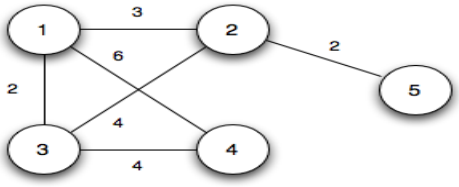
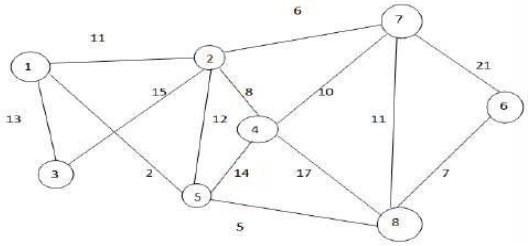
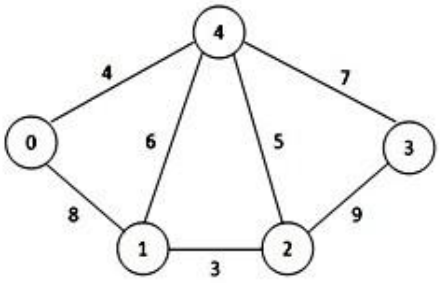
PART – A (SHORT ANSWER QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Learning Outcomes
1	Define greedy method	Remember	CAIT001.09
2	Define job sequencing with deadlines problem	Remember	CAIT001.09
3	Define minimum cost spanning tree	Remember	CAIT001.07
4	Write importance of prims algorithm	Understand	CAIT001.07
5	Write importance of kruskals algorithm	Understand	CAIT001.07

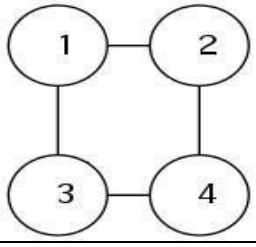
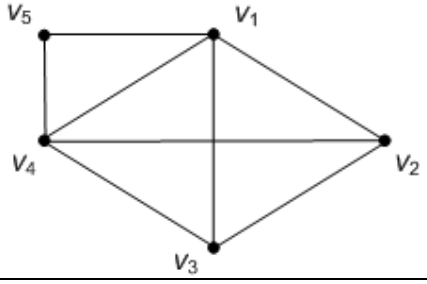
6	State single source shortest path problem	Understand	CAIT001.07
7	Define feasible solution.	Remember	CAIT001.09
8	Define optimal solution.	Remember	CAIT001.09
9	State the time complexities of prims and kruskals algorithms	Understand	CAIT001.10
10	List applications of subset paradigm.	Remember	CAIT001.10
11	Define knapsack problem.	Remember	CAIT001.09
12	Write time complexities of Prim's and Kruskal's algorithms.	Remember	CAIT001.09
13	Write high-level description of job sequencing algorithm.	Remember	CAIT001.09
14	Write the procedure of greedy method.	Remember	CAIT001.09
15	List the applications of greedy method.	Remember	CAIT001.09
1	Define dynamic programming.	Remember	CAIT001.10
2	State the principle of optimality	Understand	CAIT001.09
3	List the features of dynamic programming	Remember	CAIT001.09
4	Distinguish greedy method and dynamic programming	Understand	CAIT001.07
5	State the formula for computing cost of binary search tree.	Understand	CAIT001.07
6	Identify the number of possible binary search trees with 3 identifiers	Understand	CAIT001.07
7	State the time complexity of travelling salesperson problem using dynamic	Understand	CAIT001.10
8	List the applications of traveling sales person problem.	Remember	CAIT001.13
9	Define dominance rule.	Understand	CAIT001.10
10	State the time complexity of all pairs shortest paths problem.	Understand	CAIT001.10
11	Write an approach of dynamic programming.	Remember	CAIT001.09
12	Define 0/1 knapsack problem	Remember	CAIT001.10
13	Write advantages of travelling salesperson problem.	Understand	CAIT001.13
14	What is matrix chain multiplication problem.	Remember	CAIT001.13
15	List the applications of dynamic programming.	Remember	CAIT001.10
	PART – B (LONG ANSWER QUESTIONS)		
1	Describe job sequencing with deadlines problem and write the algorithm.	Remember	CAIT001.09
2	Explain single source shortest path problem with example using greedy method	Remember	CAIT001.10
3	Discuss the knapsack problem with suitable example	Remember	CAIT001.09
4	Write and explain an algorithm for Prim's algorithm.	Remember	CAIT001.07
5	Explain kruskals algorithm with example	Remember	CAIT001.07
6	Prove that Prim's method generates minimum-cost spanning tree.	Understand	CAIT001.07
7	Write control abstraction of greedy method and explain how it is useful for real time problems	Understand	CAIT001.07
8	Design Bellman and Ford algorithm to compute shortest path.	Understand	CAIT001.07
9	Discuss the greedy method for generating the shortest paths	Understand	CAIT001.07
10	Derive the time complexities of Prim's and Kruskal's algorithms.	Understand	CAIT001.07
11	Compare Prim's and Kruskal's algorithms.	Remember	CAIT001.07
12	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 using Prim's algorithm and showing results in each stages.	Understand	CAIT001.07

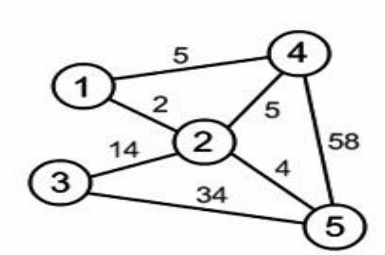
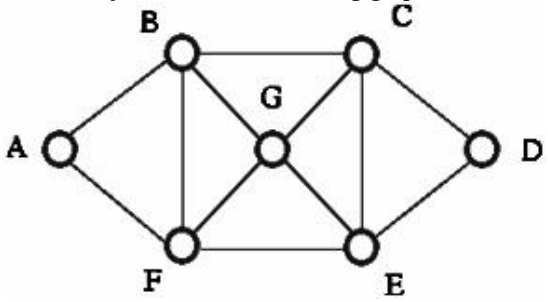
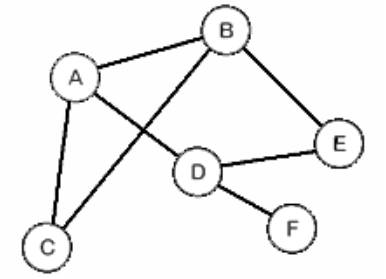
13	Give the control abstraction for subset paradigm using greedy method. Solve the job sequencing with deadline problem using greedy method for the given data N=7,P={3,5,20,18,1,6,30} are profits and D={1,3,4,3,5,1,2} are deadline respectively.	Remember	CAIT001.09
14	Find minimum cost spanning tree for 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 showing results in each stages.	Understand	CAIT001.07
15	Find 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. 	Understand	CAIT001.07
1	Explain optimal binary search tree algorithm with example	Understand	CAIT001.10
2	Explain 0/1 knapsack problem with example	Understand	CAIT001.09
3	Explain all pairs shortest path problem with example	Understand	CAIT001.10
4	Describe the travelling salesman problem and discuss how to solve it using dynamic programming?	Understand	CAIT001.10
5	Explain matrix chain multiplication with example	Understand	CAIT001.07
6	Explain single source shortest path problem with example using dynamic programming	Remember	CAIT001.10
7	Explain principle of optimality in Dynamic Programming	Remember	CAIT001.10
8	Derive the time complexity of optimal binary search tree	Understand	CAIT001.10
9	Write and Explain Informal knapsack algorithm by using dynamic Programming	Remember	CAIT001.09
10	Show the recursive steps of optimal binary search tree with an example	Understand	CAIT001.10
11	Draw all possible binary search trees with keys 10, 12, 20 and find optimal binary search tree.	Understand	CAIT001.10
12	Give the recurrence used to solve knapsack problem using dynamic programming and explain in brief the same. Solve the following Knapsack problem using dynamic programming. Capacity W=5 Item | 1 | 2 | 3 | 4 Weight | 2 | 1 | 3 | 2 Value | 12 | 10 | 20 | 15	Understand	CAIT001.09
13	Solve the 0/1 knapsack problem using dynamic programming. Given profits (60,100,120), weights (10, 20, 30) and capacity 50.	Understand	CAIT001.09
14	Compute the minimum number of multiplications required to multiply	Understand	CAIT001.07

15	Calculate shortest distances using all pairs shortest path algorithm 	Understand	CAIT001.12
PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)			
1	Compute the 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)	Understand	CAIT001.09
2	Compute the optimal solution for knapsack problem using greedy method N=3, M= 20, (p1,p2,p3)= (25,24,15), (w1,w2,w3) =(18,15,10)	Understand	CAIT001.09
3	Construct minimum cost spanning tree using Prims algorithm 	Understand	CAIT001.12
4	Apply single source shortest path algorithm for the following graph 	Understand	CAIT001.10
5	Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm. 	Understand	CAIT001.12

6	Check whether a given graph is connected or not using DFS method. 	Understand	CAIT001.07
7	Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm. 	Understand	CAIT001.10
8	Obtain the optimal solution when n=5, (p1, p2,...)=(20,15,10,5,1) and (d1,d2,...)=(2,2,1,3,3).	Understand	CAIT001.09
9	Define spanning tree. Compute a minimum cost spanning tree for the graph of figure using prim's algorithm 	Understand	CAIT001.12
1	Use optimal 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.	Understand	CAIT001.10
2	Construct optimal binary search for (a1, a2, a3, a4) = (do, if,int, while), p(1 : 4) = (3,3,1,1) q(0 : 4)= (2,3,1,1,1)	Understand	CAIT001.10
3	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	Understand	CAIT001.09
4	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)	Understand	CAIT001.09
5	Calculate shortest distances using all pairs shortest path algorithm 	Understand	CAIT001.12

6	Find the shortest tour of traveling sales person for the following cost matrix using dynamic Programming $\begin{bmatrix} \infty & 12 & 5 & 7 \\ 11 & \infty & 13 & 6 \\ 4 & 9 & \infty & 18 \\ 10 & 3 & 2 & \infty \end{bmatrix}$	Understand	CAIT001.12
7	Compute the minimum number of multiplications required to multiply chain of matrices A1,A2,A3,A4 of order 10x5, 5x10, 10x5, 5x20	Understand	CAIT001.07
8	Obtain the solution to knapsack problem by Dynamic Programming method n=6, (p1, p2,...p6)=(w1,w2,...w6)=(100,50,20,10,7,3) and m=165.	Understand	CAIT001.09
	UNIT-IV		
	PART – A (SHORT ANSWER QUESTIONS)		
1	State the principle of Backtracking	Understand	CAIT001.11
2	Write control abstraction for backtracking	Understand	CAIT001.11
3	List the applications of backtracking?	Remember	CAIT001.07
4	Define a dead node	Remember	CAIT001.07
5	Differentiate live node and dead node	Understand	CAIT001.07
6	Define state space tree	Remember	CAIT001.15
7	Define solution space	Remember	CAIT001.15
8	Define solution states and answer state?	Remember	CAIT001.13
9	State 8 – Queens problem	Understand	CAIT001.13
10	State Sum of Subsets problem	Understand	CAIT001.13
11	Define E-node	Remember	CAIT001.07
12	Define D-search	Remember	CAIT001.14
	PART – B (LONG ANSWER QUESTIONS)		
1	Write an algorithm for N-queens problem using backtracking	Understand	CAIT001.11
2	Explain subset-sum problem and discuss the possible solution strategies using backtracking.	Remember	CAIT001.11
3	Describe graph coloring problem and write an algorithm for m-coloring problem	Understand	CAIT001.13
4	Write an algorithm for Hamiltonian cycle with an example	Understand	CAIT001.15
5	Explain properties of LC search	Remember	CAIT001.14
6	Describe control abstraction for LC Search	Remember	CAIT001.14
7	Explain principle of FIFO branch and bound	Remember	CAIT001.14
8	Explain principle of LIFO branch and bound	Remember	CAIT001.14
9	Explain the method of reduction to solve travelling sales person problem using branch and bound	Remember	CAIT001.13
10	Explain TSP using branch and bound method with example	Understand	CAIT001.13
11	Explain the basic principle of Backtracking and list the applications of Backtracking.	Understand	CAIT001.11

12	Using backtracking technique solve the following instance for the subset problem s=(1,3,4,5) and d=11.	Understand	CAIT001.13
13	Draw 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.	Understand	CAIT001.13
14	Explain an algorithm for 4-queens problem using backtracking	Understand	CAIT001.11
15	Using backtracking technique solve the following instance for the subset problem s=(6,5,3,7) and d=15.	Understand	CAIT001.13
PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)			
1	Sketch the state space tree degenerated by 4 queens problem	Understand	CAIT001.13
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	Understand	CAIT001.13
3	Sketch the state space tree generated all possible 3-color,4-node graph 	Understand	CAIT001.13
4	Identify Hamiltonian cycle from the following graph 	Understand	CAIT001.25
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	CAIT001.13
6	Draw the portion of 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	Understand	CAIT001.13
7	Draw the portion of 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	Understand	CAIT001.13

8	Solve the following instance of travelling sales person problem using Least Cost Branch Bound 	Understand	CAIT001.10
9	Identify Hamiltonian cycle from the following graph 	Understand	CAIT001.13
10	Apply the backtracking algorithm to color the following graph 	Understand	CAIT001.25

UNIT-V

PART – A (SHORT ANSWER QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Learning Outcomes
1	Define class P	Remember	CAIT001.16
2	Compare NP-hard and NP-completeness	Understand	CAIT001.17
3	Define NP- hard problem	Remember	CAIT001.16
4	Define NP-complete problem	Remember	CAIT001.16
5	Define Deterministic problem?	Remember	CAIT001.16
6	Define Non-deterministic problem	Remember	CAIT001.16
7	Define a decision problem?	Remember	CAIT001.24
8	Explain Optimization problem	Remember	CAIT001.24
9	Explain Maxclique problem?	Remember	CAIT001.18
10	Define Halting problem	Remember	CAIT001.24

11	Define vertex cover problem.	Remember	CAIT001.18
PART – B (LONG ANSWER QUESTIONS)			
1	State and prove Cook's theorem	Understand	CAIT001.18
2	Explain deterministic and non-deterministic algorithms	Remember	CAIT001.20
3	Write non deterministic algorithm for sorting and searching	Understand	CAIT001.18
4	Write a non-deterministic knapsack algorithm	Understand	CAIT001.19
5	Explain how P and NP problems are related	Remember	CAIT001.20
6	Distinguish NP- hard and NP-complete problems	Understand	CAIT001.20
7	Explain decision problem with an example	Understand	CAIT001.24
8	Explain chromatic number decision problem and clique decision problem	Remember	CAIT001.23
9	Explain the strategy to prove that a problem is NP-hard	Remember	CAIT001.21
10	Explain intractable problems with examples	Remember	CAIT001.18
PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)			
1	Show that satisfiability is at most three literals reduces to chromatic number	Understand	CAIT001.23
2	Prove Hamiltonian cycle is in NP	Understand	CAIT001.21
3	Prove circuit-SAT is in NP	Understand	CAIT001.21
4	List two problems that have polynomial time algorithms justify your answer	Understand	CAIT001.23
5	Explain 3CNF satisfiability problem	Remember	CAIT001.19
6	Explain P type problems with examples	Remember	CAIT001.23

Prepared By:
Dr. K Rajendra Prasad, Professor
HOD, CSE

HOD, IT