DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY

IV Semester: CSE / IT

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AITB07	Core	L	T	P	C	CIA	SEE	Total
		-	-	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes: 36		

OBJECTIVES:

The course should enable the students to:

I. Learn how to analyze a problem and design the solution for the problem.

II. Design and implement efficient python programming for a specified application.

III. Identify and apply the suitable algorithm for the given real world problem.

COURSE OUTCOMES:

The student will have the ability to:

CO 1: Implement Quick sort ,Merge sort and Warshall's algorithm.

CO 2: Implement Dynamic Programming algorithm for the 0/1 Knapsack problem and greedy algorithm for job sequencing with deadlines.

CO 3: Implement Dijkstra’s , Prim’s, Kruskal’s algorithm on spanning tree.

CO 4: Implement Tree Traversal and Graph Traversals techniques using BFS and DFS.

CO 5: Implement Floyd’s algorithm for the all pair’s shortest path problem and N-queens problem.

COURSE LEARNING OUTCOMES:

1. Understand the basic concepts of python.
2. Understand the different sorting techniques to organize the data in ascending or descending order using quick sort and merge sort.
3. Computing the transitive closure of a given directed graph using Warshall's algorithm.
4. Implementation of dynamic programming for knapsack problem.
5. Identify the shortest paths to other vertices using Dijkstra’s algorithm.
6. Analyze the concept of minimum cost spanning trees using Kruskal’s algorithm
7. Implementation of tree traversal algorithms for given graphs.
8. Understand graphs and graph traversal techniques like Depth first search and Breadth first search.
9. Understand and implement the sum of subsets problem.
10. Implement the travelling salesperson problem.
11. Analyze the concept of minimum cost spanning trees using Prim’s algorithm
12. Implementation of All-Pairs Shortest Paths Problem using Floyd's algorithm and N-Queens problem

LIST OF EXPERIMENTS

WEEK-1 QUICK SORT

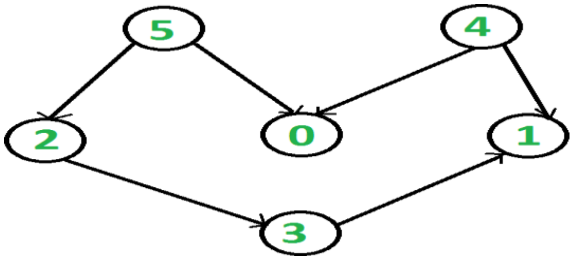
Sort a given set of elements using the quick sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the 1^{st} to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.

WEEK-2 MERGE SORT

Implement merge sort algorithm to sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.

WEEK-3 WARSHALL’S ALGORITHM

a) Obtain the Topological ordering of vertices in a given digraph.



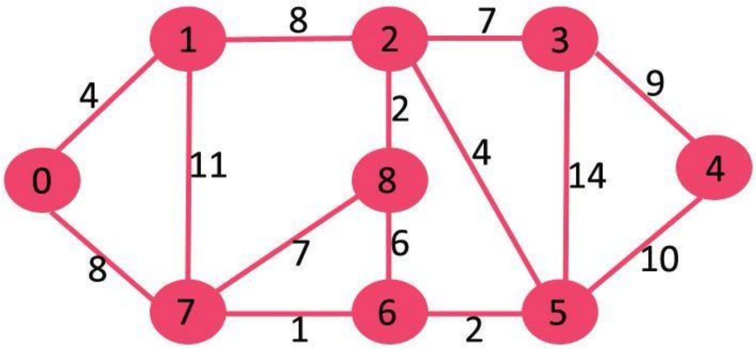
b) Compute the transitive closure of a given directed graph using Warshall's algorithm.

WEEK-4 KNAPSACK PROBLEM

Implement 0/1 Knapsack problem using Dynamic Programming.

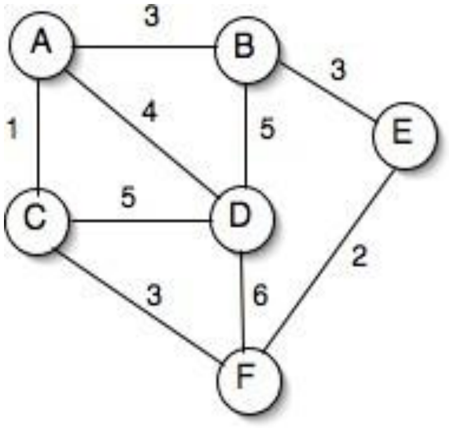
WEEK-5 SHORTEST PATHS ALGORITHM

From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra’s algorithm.



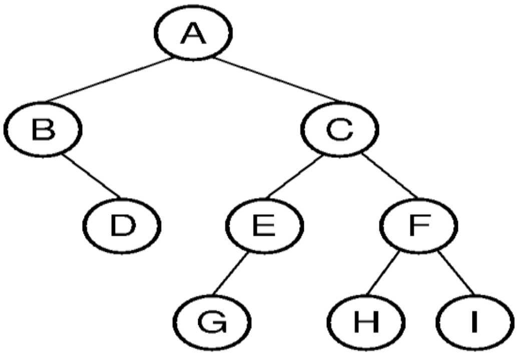
WEEK-6 **MINIMUM COST SPANNING TREE**

Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal’s algorithm.



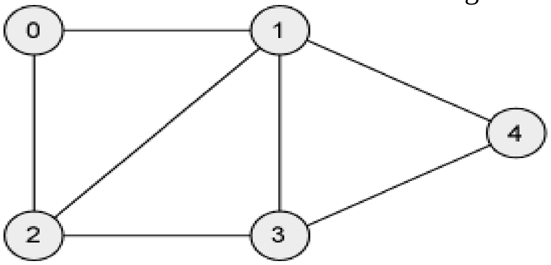
WEEK-7 **TREE TRAVESRSALS**

Perform various tree traversal algorithms for a given tree.

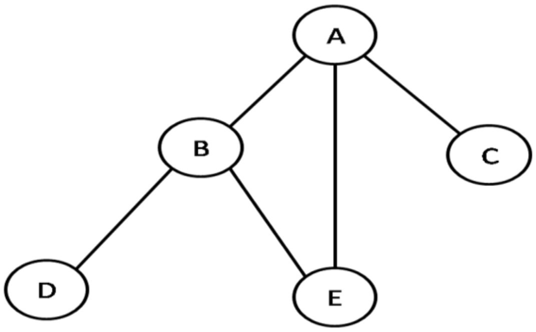


WEEK-8 GRAPH TRAVERSALS

a. Print all the nodes reachable from a given starting node in a digraph using BFS method.



b. Check whether a given graph is connected or not using DFS method.



WEEK-9 SUM OF SUB SETS PROBLEM

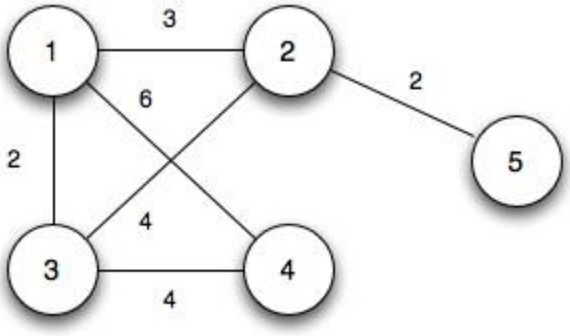
Find a subset of a given set S = {sl, s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions {1, 2, 6} and {1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.

WEEK-10 TRAVELLING SALES PERSON PROBLEM

Implement any scheme to find the optimal solution for the Traveling Sales Person problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.

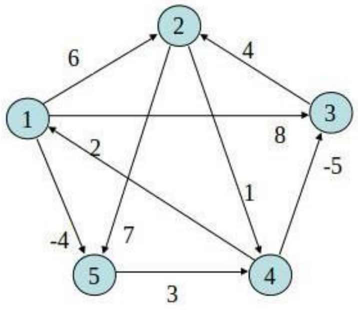
WEEK-11 MINIMUM COST SPANNING TREE

Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.



WEEK-12	**ALL PAIRS SHORTEST PATHS**

Implement All-Pairs Shortest Paths Problem using Floyd's algorithm.



	1	2	3	4	5
1	0	6	8	∞	-4
2	∞	0	∞	1	7
3	∞	4	0	∞	∞
4	2	∞	-5	0	∞
5	∞	∞	∞	3	0

WEEK-13	**N QUEENS PROBLEM**

Implement N Queen's problem using Back Tracking.

Reference Books:

1. Levitin A, “Introduction to the Design And Analysis of Algorithms”, Pearson Education, 2008.
2. Goodrich M.T.,R Tomassia, “Algorithm Design foundations Analysis and Internet Examples”, John Wileyn and Sons, 2006.
3. Base Sara, Allen Van Gelder ,“ Computer Algorithms Introduction to Design and Analysis”, Pearson, 3rd Edition, 1999.

Web References:

1. http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html
2. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
3. http://www.facweb.iitkgp.ernet.in/~sourav/daa.html