Course Title : ANALYSIS AND DESIGN OF ALGORITHMS LAB			
Course Code : P18CSL47	Semester : 4	L :T:P:H : 0:0:3:3	Credits: 1.5
Contact Period: Laboratory : 3 Hrs/week, Exam: 3 Hr		Weightage: CIE:50%, SEE:50%	

Divide and Conquer

1.Sort a given set of n integer elements using **Quick Sort** method and compute its time complexity. Run the program for varied values of n> 5000 and record the time taken to sort. The elements can be read from a file or can be generated using the random number generator.
2.Sort a given set of n integer elements using **Merge Sort** method and compute its time complexity. Run the program for varied values of n> 5000, and record the time taken to sort. The elements can be read from a file or can be generated using the random number generator.

Greedy Method

3. Implement the **0/1 Knapsack problem using**Greedy method
4.From a given vertex in a weighted connected graph, find shortest paths to other vertices using **Dijkstra's algorithm.**
5.Find Minimum Cost Spanning Tree of a given connected undirected graph using **Kruskal'salgorithm**. Use Union-Find algorithms in the program.
6. Find Minimum Cost Spanning Tree of a given connected undirected graph using**Prim's algorithm.**
7. Sort a given set of n integer elements using **Heap Sort** method and compute its time complexity. Run the program for varied values of n> 5000 and record the time taken to sort. The elements can be read from a file or can be generated using the random number generator.

Dynamic Programming

8. Writeprogram to Implement All-Pairs Shortest Paths problem using **Floyd's algorithm.**
9. Write program to implement **Warshalls algorithm.**
10. Implement the **0/1 Knapsack problem** usingDynamic Programming method.
11. Implement **TravellingSales Person** problem using Dynamic programming.

Backtracking

12Design and implement a C program to **find a subset** of a given set S = {Sl, S2,.....,Sn} of n positiveintegers 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}. Display a suitable message, if the givenproblem instance doesn't have a solution.
13 Design and implement a C program to find all **HamiltonianCycles** in a connected undirectedGraph G of n vertices using backtracking principle.
14. Design and implement**'n' Queens Problem**using Backtracking method.