

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE
Department of Information Science and Engineering

SEMESTER – IV

DESIGN AND ANALYSIS OF ALGORITHMS

Credit: 4

Course Code	22ISE144	CIA Marks	50
Teaching Hours/Week (L: T: P: J)	3:0:2:0	SEA Marks	50
Total Number of Lecture Hours	50	Exam Hours	03

Final List of Programs for CIA and SEE

Module-1: Introduction to Design and Analysis of Algorithms

Practical:

1. Implement the concept of Linear Search
2. There are 5 books in the shelf, find the number of ways to select 3 books from 5 books on the shelf using the NCR with recursion.
3. Find the next three terms of the sequence 15, 23, 38, 61, ... Fibonacci series of the given number using recursion

**CO1)
Analyze**

Module-2: Brute Force, Divide and conquer

Practical:

4. Implement the selection sort algorithm (Brute Force Technique).
5. Write a program to search a key in a given set of elements using Binary search method and find the time required to find the key.
6. Sort a given set of elements using Quick Sort method and determine the time required sort the elements. Plot a graph of number of elements versus time taken. Specify the time efficiency class of this algorithm

**(CO2)
Apply**

Module-3: Greedy Method

Practical:

7. Implement Prim's algorithm and Find Minimum Cost Spanning Tree of a given connected undirected graph.
8. Implement Kruskal's algorithm and Find Minimum Cost Spanning Tree of a given connected undirected graph.
9. Implement Dijkstra's algorithm find shortest paths to other vertices from a given vertex in a weighted connected graph.

**(CO3)
Apply**

Module-4: Dynamic Programming	
Practical: 10. Implement all-pairs shortest paths problem using Floyd's algorithm. 11. Implementation of Bellman Ford Algorithm using a directed graph. 12. Implementation of Travelling Salesperson Problem using dynamic programming.	(CO4) Apply
Module-5: Backtracking	
Practical: 13. Implementation of N Queen Problem using Backtracking technique. 14. Implementation of SUM-SUBSET Problem 15. Implementation of Knapsack problem using Branch and Bound method	(CO5) Analyze