2/4 B.Tech SECOND SEMESTER

IT4T3 DESIGN & ANALYSIS OF ALGORITHMS Credits: 4
(Common to CSE/IT/ECM)

Lecture: 4 periods/week	**Internal assessment: 30 marks**
Tutorial: 1 period /week	**Semester end examination: 70 marks**

Objectives:

- To explain the paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.
- To discuss the development of efficient algorithms for simple computational tasks and reasoning and their correctness.
- To familiarize the time order analysis for an algorithm, the space needs for the implementation of an algorithm and prove the correctness of an algorithm.
- To provide the complexity measures, different range of behaviors of algorithms and the notion of tractable and intractable problems.

Outcomes:

Student will be able to:

- Understand the principles of good algorithm design.
- Analyze the algorithms and estimate their worst-case and average-case behavior.
- Gain familiarity with different Brute Force techniques, Divide and Conquer and decrease and conquer techniques.
- Know a variety of greedy algorithms; know the basic ingredients of a greedy algorithm, and how to approach arguing the correctness of such algorithms
- Get familiar with dynamic-programming algorithms and apply them to test for optimality.
- Understand Backtracking, Branch and Bound technique, Know some standard NP-Complete problems and know the basics of an NP-hardness

Syllabus:

UNIT-I
INTRODUCTION:
Notion of Algorithm, Fundamentals of Algorithmic Problem Solving-Understanding the problem, deciding on appropriate data structures, Algorithm Design techniques, Methods of specifying an algorithm, proving an algorithm’s correctness, Analyzing and coding an Algorithm. Fundamentals of the Analysis of Algorithm Efficiency Analysis framework and Asymptotic Notations and Basic Efficiency Classes

UNIT-II
BRUTE FORCE AND EXHAUSTIVE SEARCH:
Selection sort, Bubble sort, Sequential search, Brute-Force String Matching. Exhaustive search- Travelling salesman problem, knapsack problem and Assignment problem.

UNIT-III

DIVIDE-AND-CONQUER:

Mergesort, Quicksort, Binary Search, Binary Tree Traversals and Related Properties, Multiplication of large integers, Strassen's Matrix Multiplication.

UNIT-IV

DECREASE-AND-CONQUER & TRANSFORM-AND-CONQUER:

DECREASE-AND-CONQUER Insertion Sort, Topological Sorting, Decrease-by-Constant-Factor Algorithms fake-coin problem, Josephus problem. TRANSFORM-AND-CONQUER Presorting, Heaps and heap sort, Horner's rule.

UNIT-V

GREEDY TECHNIQUE:

Prim's Algorithm, Kruskal's Algorithm Disjoint Subsets and Union-Find Algorithms, Dijkstra's Algorithm, Huffman trees.

UNIT-VI

DYNAMIC PROGRAMMING:

The Knapsack Problem and Memory Functions, Optimal Binary Search Trees, Warshall's and Floyd's Algorithms.

UNIT-VII

LIMITATIONS OF ALGORITHM POWER:

Decision Trees Decision Trees for Sorting Algorithms and Decision Trees for Searching Sorted Array. P, NP, and NP-complete Problems.

UNIT-VIII

COPING WITH THE LIMITATIONS OF ALGORITHM POWER :

Backtracking n-queens problem, Hamiltonian Circuit problem, Subset-sum problem. Branch-and-Bound Assignment Problem, Knapsack Problem and Travelling Salesman problem.

Textbook:

Introduction to The Design & Analysis of Algorithms, Anany Levitin, 2nd Edition, Pearson Education, 2007.

Reference book:

1. **Introduction to Algorithms**, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 2ndEdition, PHI, 2006.