**Design and Analysis of Algorithms**

**Unit I**

Asymptotic Bounds and Representation problems of Algorithms: Computational Tractability: Some Initial Attempts at Defining Efficiency, Worst-Case Running Times and Brute-Force Search, Polynomial Time as a Definition of Efficiency, Asymptotic Order of Growth: Properties of Asymptotic Growth Rates, Asymptotic Bounds for Some Common Functions, A Survey of Common Running Times: Linear Time, O(n log n) Time, Quadratic Time, Cubic Time, O(nk) Time, Beyond Polynomial Time, Sub linear Time.

Some Representative Problems, A First Problem: Stable Matching: The Problem, Designing the Algorithm, Analyzing the Algorithm, Extensions, Implementing the Stable Matching Algorithm, Using Lists and Arrays: Arrays and Lists, Five Representative Problems: Interval Scheduling, Weighted Interval Scheduling, Bipartite Matching, Independent Set, Competitive Facility Location.

**Unit II**

**Graphs & Divide and Conquer:** Graph Connectivity and Graph Traversal, Breadth-First Search: Exploring a Connected Component, Depth-First Search, Implementing Graph Traversal Using Queues and Stacks: Implementing Breadth-First Search, Implementing Depth-First Search, An Application of Breadth-First Search: The Problem, Designing the Algorithm, Directed Acyclic Graphs and Topological Ordering: The Problem, Designing and Analyzing the Algorithm, A First Recurrence: The Merge sort Algorithm: Unrolling the Merge sort Recurrence, Counting Inversions: The Problem, Designing and Analyzing the Algorithm.

**Unit III**

Greedy Algorithms: Interval Scheduling: The Greedy Algorithm Stays Ahead: Designing a Greedy Algorithm, Analyzing the Algorithm, Scheduling to Minimize Lateness: An Exchange Argument: The Problem, Designing the Algorithm