# **Advanced Data Structures and Algorithms**

#### **Chapter 1: Role of Algorithms in Computing and Complexity Analysis**

1.1 Introduction to Algorithm and Analysis of Algorithm

1.2 Asymptotic Notations

1.3 Importance of Efficient Algorithms

1.4 Program Performance Measurement

1.5 Recurrences: The Substitution Method – The Recursion – Tree

Method

1.6 Data Structures and Algorithms

#### **Chapter 2: Hierarchical Data Structures**

2.1 Introduction to Hierarchical Data Structures

- 2.2 Binary Search Tree (BST)
- 2.3 Red-Black Trees
- 2.4 Binary Tree
- 2.5 Basic Operations on Binary Trees
- 2.6 Heap Data Structure
- 2.7 Disjoint Sets

## Chapter 3: Graphs

- 3.1 Introduction to Graphs
- 3.2 Representation of Graph
- 3.3 Graph Traversals
- 3.4 Topological Sort
- 3.5 Strongly Connected Components (SCC)
- 3.6 Minimum Spanning Trees (MST)
- 3.7 Single Source Shortest Path Algorithms
- 3.8 Dynamic Programming (DP)
- 3.9 Matrix Chain Multiplication Using Dynamic Programming

## **Chapter 4: Algorithm Design Techniques**

- 4.1 Dynamic Programming (DP)
- 4.2 Elements of Dynamic Programming

- 4.3 Matrix Chain Multiplication using Dynamic Programming
- 4.4 Longest Common Subsequence (LCS)
- 4.5 Greedy Algorithms
- 4.6 Activity-Selection Problem
- 4.7 Huffman Algorithm

## Chapter 5: NP Complete and NP Hard

- 5.1 Types of Problems: P & NP Classification
- 5.2 NP-Completeness
- 5.3 Polynomial Time Verification
- 5.4 NP-Completeness and Reducibility
- 5.5 NP-Completeness Proofs