Course Syllabus: **Design and Analysis of Algorithms**

Institute for Advancing Intelligence (IAI), TCG CREST

Instructors: Dr. Laltu Sardar , Dr. Ritankar Mandal, Prof. Rana Barua
Course Webpage: https://laltu-sardar.github.io/courses/courses_algo_2023_24.html

Course Description

This course provides a comprehensive overview of advanced algorithms and data structures. It covers various methods and principles used in algorithm design and analysis, including efficiency, growth of functions, and different types of complexities. The course also delves into several data structures, their implementations, and applications. Advanced topics such as graph algorithms, NP-completeness, and probabilistic algorithms are included to prepare students for complex problem-solving in computer science.

Course Objectives

- Understand and apply various algorithm design paradigms.
- Analyze the efficiency and complexity of algorithms.
- Solve complex problems using advanced algorithmic strategies.

Topics to be Covered

Introduction

- Algorithms and Problem Instances
- Efficiency and Growth of Functions
- Asymptotic Notation
- Time Complexity: Worst, Best, Average Case
- Substitution Method and Recursion Tree Method
- Master's Theorem

Elementary Data Structures

- Arrays and Linked Lists
- Stacks and Queues
- Hash Tables
- Binary Search Trees
- Disjoint Set Data Structures

Searching and Sorting

- Linear and Binary Search
- Heap Sort and Quick Sort
- Sorting in Linear Time: Counting sort, Radix sort, Bucket sort

Divide and Conquer Paradigm

- Merge Sort
- Counting Inversions
- Closest Pair of Points

Greedy Algorithms

- Interval Scheduling and Variants
- Optimal Caching Problem
- Minimum Spanning Tree and Huffman Code
- Clustering and Fractional Knapsack Problems
- Dijkstra's Algorithm

Dynamic Programming

- Matrix Chain Multiplication
- Longest Common Subsequence
- Optimal Binary Search Tree
- Segmented Least Squares Problem
- 0/1-Knapsack and Subset Sum Problems
- Bellman-Ford Algorithm

Graph Algorithms

- Breadth-First and Depth-First Search
- Floyd-Warshall Algorithm
- Ford-Fulkerson Algorithm

String Matching Problems

- Introduction to string matching algorithms
- Naive string matching algorithm
- Knuth-Morris-Pratt (KMP) Algorithm
- Rabin-Karp Algorithm
- Suffix Trees and Arrays
- Applications of string matching in computational biology and text processing

Advanced Topics

1. **Introduction to Advanced Computational Complexity**
    - Overview of computational models
    - Complexity classes: P, NP, and NPC
    - Decidability and reducibility
2. **Circuit Satisfiability**
    - Boolean circuits and SAT problem
    - NP-completeness of Circuit SAT
    - SAT solvers and their applications
3. **Vertex Cover and Graph Coloring**
    - Problem definitions and applications
    - NP-completeness proofs
    - Exact and approximation algorithms for Vertex Cover
    - Graph coloring algorithms and applications
4. **Reductions and Hardness of Approximation**
    - Polynomial-time reductions
    - Techniques for proving NP-hardness
    - Hardness of approximation: examples and principles
5. **Approximation Algorithms**
    - Design principles for approximation algorithms
    - Approximation schemes: PTAS and FPTAS
    - Case studies: Knapsack problem, Traveling Salesman Problem, Set Cover
6. **Probabilistic Algorithms**
    - Fundamentals of probabilistic algorithms
    - Monte Carlo and Las Vegas algorithms
    - Applications of probabilistic algorithms in cryptography and network security

Textbook and Materials

Primary Textbook: ”Introduction to Algorithms” by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
Supplementary Textbooks:

- ”Algorithms” by Robert Sedgewick and Kevin Wayne
- ”Data Structures and Algorithm Analysis in C++” by Mark Allen Weiss
- ”Algorithm Design” by Jon Kleinberg and Éva Tardos
- ”The Algorithm Design Manual” by Steven S. Skiena