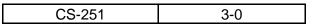
Design and Analysis of Algorithms



Course Description

This course offers a comprehensive exploration of the design and analysis of algorithms, equipping students with the skills and knowledge needed to tackle real-world computational challenges. The course focuses on various algorithmic problem-solving techniques, including dynamic programming, divide-and-conquer, greedy algorithms and more, which are applied to a wide range of practical problems. Students learn to communicate their algorithmic solutions effectively, ensuring that they can present the algorithms with correctness proof.

Textbook:

- 1. Thomas H. Cormen, Introduction to Algorithms (4th edition) 2022, MIT Press
- 2. Jon Kleinberg, Eva Tardos. Algorithm Design, 1st Edition. 2006. Pearson Education, Inc.

Reference Book:

- 1. Clifford A. Shaffer, Data Structures and Algorithm Analysis, Edition 3.2, 2012.
- 2. Steven S Skiena, The Algorithm Design Manual, 2nd Edition (2008).
- 3. M. A. Iqbal, Graph Theory & Algorithms, Google Books, 2012.

Prerequisites

CS-250 (Data Structures and Algorithms)

ASSESSMENT SYSTEM

Quizzes	10%
Assignments	10%
Semester Project	10%
Mid Semester Exam (MSE)	30%
End Semester Exam (ESE)	40%

Teaching Plan

	Торіс
Week 1	Role of Algorithms in Computing
	Growth of Functions
	Asymptotic Analysis
Week 2	Recurrence Relations
	 Solving recurrences using Substitution
	 Solving recurrences using Recursion Tree Method
	 Solving recurrences using Master Method
Week 3	Sorting Algorithms
	 Sorting Algorithms: Selection Sort, Bubble Sort, Insertion Sort
	Sorting Algorithms: Quick Sort
	Analysis of Sorting Algorithms: Randomized Quick Sort

Week 4	 Analysis of Sorting Algorithms: Merge Sort Analysis of Sorting Algorithms: Heap Sort Drigrity Queues (Heaps)
Week 5	 Priority Queues (Heaps) Analysis of Sorting Algorithms: Counting Sort Analysis of Sorting Algorithms: Radix Sort Analysis of Sorting Algorithms: Bucket Sort
Week 6	 Graph Algorithms Introduction to Graphs Breadth-First Search and Depth-First Search Introduction to Trees, MST's, Kruskal's Algorithm
Week 7	 Prim's algorithm Shortest Path Algorithms Dijkstra Shortest Path Algorithm
Week 8	 Bellman-Ford shortest path algorithm Floyd-Warshall all pair shortest path algorithm Johnson All pair shortest path algorithm
Week 9	Mid Semester Exam (MSE)
Week 10	 Greedy algorithms Huffman coding Activity selection problem Bin Packing Problem Money Counting Problem
Week 11	 Linear Programming System of linear inequalities Linear Programming Word Problems
Week 12	 Simplex Algorithm Standard and Slack form Practice Examples using Simplex method
Week 13	 String Matching Algorithms Naïve String-Matching Algorithm Rabin Karp String Matching Algorithm Knuth-Morris-Pratt String Matching Algorithm
Week 14	Memoization Dynamic programming • Knapsack algorithm
Week 15	Weighted Interval Scheduling ProblemAssembly Line Scheduling
Week 16	NP Completeness
Week 17	Project presentations
Week 18	End Semester Exam (ESE)