School of Computer Engineering

Kalinga Institute of Industrial Technology (KIIT)
Deemed to be University
Bhubaneswar-751024

Lesson Plan and Activity Calendar

Design and Analysis of Algorithms – CS30001 (L-T-P-Cr: 3-0-0-3)

Semester: 5th
Discipline: B.Tech. (CSE), **Section:** CSE-07
Session: Autumn 2024

Prerequisites: Data Structures (CS21001)

Instructor:

Name :
Chamber :
Email :

Class Hours:

Day	Time	Class Room

:----------

Lesson Plan:

Unit	Unit Name	Topics to be covered	No. of lectures	Lecture serial nos.
1	Introduction	● Concepts in algorithm analysis & design - motivation ● Complexity of an algorithm (Space and time Complexity), Analysis of time complexity of Insertion Sort by step count method ● Growth of functions, Asymptotic Notations (Big Oh, Omega, Theta)) ● Solving recurrences: Iterative method Substitution method, Recurrence Tree method ● Solving recurrences: Master theorem, Change of variable	8	1-8
		Tutorials / Activity		

2a	**Divide and Conquer Approach**	● Structure of Divide-and-Conquer algorithm design technique ● Analysis of divide-and-conquer run time recurrence relations of ■ Finding Max/Min ■ Binary Search ■ Merge Sort ■ Quick Sort **Tutorials / Activity**	**3**	**9-11**
2b	**Greedy Approach**	● Revision of Heap: Build Heap/ Insertion/Deletion, Complexity ● Overview of Greedy design paradigm and Solving as well as analyzing the following problems using Greedy method: ■ Fractional knapsack problem ■ Job sequencing with deadlines ■ Huffman method of Optimal Coding ■ Finding Minimum spanning trees for a Graph: Kruskal's Method ■ Finding Minimum spanning trees for a Graph: Prim's Method ■ Finding Single Pair Shortest Path in a graph: Dijkstra's Method **Tutorials / Activity**	**7**	**12-18**
3	**Dynamic Programming Approach**	● Overview of Dynamic Programming paradigm, Difference between Dynamic Programming and Divide & Conquer/Greedy Methods ● Solving the following problems using Dynamic Programming method: ■ 0/1 Knapsack problem ■ Matrix Chain Multiplication ■ Longest Common Subsequence ● Introduction to Multistage Graph problem: : Problem statement, DP formulation, Discussion on Complexity ● All Pair Shortest Paths in a Graph - Floyd Warshall Algorithm ● Optimal Binary Search Tree (OBST): Problem statement, DP formulation, Discussion on Complexity ● Travelling Salesman Problem (TSP): Problem statement, DP formulation, Discussion on Complexity **Tutorials / Activity**	**8**	**19-26**

4	**Amortized Analysis and Randomized Algorithms**	• Basic Idea of Amortized Analysis of algorithms: Notions of Aggregate Analysis Method, Accounting Method, Potential Method • Randomized Algorithms: Basic Idea of Las Vegas and Monte Carlo algorithms • Randomized Quick Sort: Basic Idea • Minimum Cut in a graph: Basic Idea **Tutorials / Activity**	**6**	**27-32**
6.	**Complexity Classes and Approximation Algorithms**	• Concepts of Complexity Classes: P, NP, NP-Hard and NP-Complete • Reducibility of problems • Complexity Classes for the following selected problems: ▪ 3-CNF Satisfiability Problem ▪ Travelling Salesman Problem ▪ Maximal Clique Problem ▪ Hamiltonian cycle Problem ▪ Vertex Cover Problem • Introductory idea of Approximation algorithms • Approximation algorithms for following selective problems: ▪ Travelling Salesman Problem ▪ Vertex Cover Problem **Tutorials / Activity**	**8**	**32-40**

Day-wise Lesson Handouts:

Week	Lecture No.	Topics
Week - 1	1	Concepts in algorithm, difference between Algorithm and Program, characteristics of algorithms. Algorithm design and their Performance Analysis: Time and Space Complexity, motivation.
	2	Pseudo code Conventions, Analysis of Insertion Sort by step count method (Incremental Approach), Best-case, Worst-case and Average-case Analysis.
	3	Growth of functions, Asymptotic Notations (θ, O, Ω)
Week - 2	4	Solving recurrences using Iterative method
	5	Solving recurrences using Substitution method
	6	Solving recurrences using Recursion Tree method
Week – 3	7	Solving recurrences using Master's Theorem
	8	Solving recurrences using Change of Variables, Discussion and Examples of Limitations of Master Theorem
	9	Divide-and-Conquer Approach - Binary Search and its complexity analysis, Finding Minimum/Maximum
Week - 4	10	Divide-and-Conquer Approach - Merge Sort and its complexity analysis
	11	Divide-and-Conquer Approach - Quick Sort and its complexity analysis (Worst-case, Best-case, Role of Pivot Choice and Balanced Partitioning)
	12	Revision of Max and Min Heap: Discussion on Complexity of Heap Insertion, Heap Deletion and Heap Building

Week - 5	13	Overview of Greedy paradigm, Elements of greedy strategy: Recursive, Iterative greedy algorithm. Knapsack Problem, Difference between Fractional Knapsack and 0/1 Knapsack, Greedy strategy for solving Fractional Knapsack Problem
	14	Job sequencing with deadlines - A greedy problem formulation and solving
	15	Problem of Optimal coding, Huffman Tree and Huffman method for Optimal coding
Week - 6	16	Minimum-Cost Spanning Tree (MST) for a weighted graph, Kruskal's algorithm for finding MST
	17	Prim's Method for finding MST
	18	Single-Source Shortest Path problem for weighted graphs, Dijkstra's Method
Week - 7	19	Overview of Dynamic Programming paradigm, Divide and Conquer vs Dynamic Programming, Greedy vs Dynamic Programming, Elements of dynamic programming, Tabulation vs Memoization approach
	20	Dynamic Programming approach for solving 0/1 Knapsack problem
	21	Matrix Chain Multiplication (MCM) problem, Dynamic Programming approach for solving MCM problem
Week - 8	22	Longest Common Subsequence (LCS) problem, Dynamic Programming approach for solving LCS problem
	23	Multistage Graph problem: Problem Statement, DP Formulation, Discussion
	24	All Pair Shortest Path problem for weighted graph: Floyd Warshall algorithm
Week - 9	25	Optimal Binary Search Tree (OBST): Statement, DP Formulation, Discussion
	26	Travelling Salesman Problem (TSP): Statement, DP Formulation, Discussion
	27	Basic Idea of Amortized Analysis of algorithms, Introductory Idea of Aggregate Analysis Method of Amortized Analysis
Week - 10	28	Introductory Idea of Accounting Method of Amortized Analysis
	29	Introductory Idea of Potential Method of Amortized Analysis
	30	Basic Idea of Las Vegas and Monte Carlo algorithms
Week - 11	31	Randomized Quick Sort: Basic Idea and discussion on average case analysis
	32	Minimum Cut in a graph: Concept and Introductory Idea
	33	Tractable vs Intractable problems, Decision vs Optimization Problems, Deterministic vs Nondeterministic Algorithms. Reduction, Polynomial Reduction and Equivalence of Problems.
Week - 12	34	Basic Concepts of Complexity Classes (P, NP, NP hard, NP Complete) and their hierarchy
	35	3CNF-SAT: Problem Statement and Discussion on Complexity Class
	36	Travelling Salesman Problem
Week - 13	37	Maximal Clique Problem
	38	Hamiltonian cycle Problem, Vertex Cover Problem
	39	Introductory Idea to Approximation Algorithms
	40	Approximation Algorithms for Vertex Cover and TSP

Activity Calendar - Autumn 2023

Activity No.	Type of Activity	Probable Date	Marks (Weightage)	CO
1	**ACTIVITY-1** (Class Test/ Subjective Test)	22.07.24 – 26.07.24	5	
2	**ACTIVITY-2** (Class Test/Quiz Test)	05.08.24 – 09.08.24	5	
3	**ACTIVITY-3** (Class Test/ Surprise Test)	02.09.24 – 06.09.24	5	
Mid Semester Examination [17.09.2024 – 21.09.2024]				
4	**ACTIVITY-4** (Home Assignment)	30.09.24 – 04.10.24	5	
5	**ACTIVITY-5** (Class Test/Home Assignment)	21.10.24 – 25.10.24	5	
6	**ACTIVITY-6** (Class Test/Quiz Test)	04.11.24 – 08.11.24	5	
End Semester Examination [16.11.2024 – 26.11.2024]				

Course Outcome: Upon completion of this course, the students will be able to:

CO1:	Analyze the time and space complexity for any algorithm
CO2:	Compare and contrast different algorithm design techniques
CO3:	Apply the algorithm design techniques in solving real world problems
CO4:	Perform amortize analysis for any algorithm
CO5:	Modify existing algorithms to apply in common engineering design situations
CO6:	Use NP class of problems to propose approximation algorithms

Text books:

- Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Universities Press.
- Thomas H. Coreman, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", PHI.

Reference books:

- Jon Kleinberg, Eva Tardos, "Algorithm Design", Pearson.
- Michael T. Goodrich, Roberto Tamassia, "Algorithm Design: Foundations, Analysis, and Internet Examples", Wiley India.

Grading Policy:

Pedagogy: Lecture, Assignments, Quiz, Debate, Short Projects, etc.

Evaluation Methodology: Internal: 50 (20- Midterm Exam & 30 Activity), End Term: 50

Distribution of Marks:

SL No.	Evaluation Component	Evaluation Marks	Course Lecture No.		Mode
			From	To	
1	Mid-Semester Examination	20	1	21	Closed Book
2	Activity based Teaching and Learning	30	NA	NA	Open Book, Closed Book and Presentation, Short quiz
3	End-Semester Examination	50	1	40	Closed Book

Note

- ***Tentative Mid-Semester Syllabus would be up to Matrix Chain Multiplication (MCM) problem under the unit of Dynamic Programming approach as per the Lesson Plan***
- ***Modifications to the above-mentioned structure (Lesson Plan / Examination Process / Any other modifications) may take place as per the Teacher's discretion adhering to the University Guidelines.***

==========XXXXXXXX==========