

Branch Name:	Computer Engineering / Information Technology/Information and Communication Technology
Program Code:	107
Course Name:	Design and Analysis of Algorithm
Course Code:	1ET1030502
Pre-requisite Course:	Fundamentals of Data and file Structures

Course Objectives:

1. To understand and apply the algorithm analysis techniques.
2. To critically analyze the efficiency of alternative algorithmic solutions for the same problem
3. To understand different algorithm design techniques.
4. To understand the limitations of Algorithmic power.

Teaching and Examination Scheme:

Teaching Scheme (Hours per week)				Evaluation Scheme (Marks)				
Lecture (L)	Tutorial (T)	Practical (P)	Credit	Theory (Marks)		Practical (Marks)		Total (Marks)
				University Assessment	Continuous Assessment	University Assessment	Continuous Assessment	
03	-	02	04	70	30	30	20	150

Subject Contents:

Sr. No	Topic	Total Hours	Weightage (%)
1	Analysis of Algorithm: Characteristic of algorithm, The efficient algorithm, Average, Best and worst case analysis, Asymptotic Notations, Analyzing control statement, Sorting Algorithms and analysis: Bubble sort, Selection sort, Insertion sort, Shell sort, Heap sort, Sorting in linear time : Bucket sort, Radix sort	7	15
2	Divide and Conquer Algorithm: Introduction, Recurrence and different methods to solve recurrence, Multiplying large Integers Problem, Problem Solving using divide and conquer algorithm -Binary Search, Sorting (Merge Sort, Quick Sort), Matrix Multiplication, Exponential.	8	20

3	Greedy Algorithm :General Characteristics of greedy algorithms, Problem solving using Activity selection problem, Elements of Greedy Strategy, Minimum Spanning trees (Kruskal's algorithm, Prim's algorithm), Graphs: Shortest paths, The Knapsack Problem, Job Scheduling Problem, Huffman code	8	20
4	Dynamic Programming: Introduction, The Principle of Optimality, Problem Solving using Dynamic Programming, Making Change Problem, Assembly Line Scheduling, Knapsack problem, All Points Shortest path, Matrix chain multiplication, Longest Common Subsequence.	7	15
5	Backtracking and Branch and Bound: Introduction, The Eight queens problem , Knapsack problem, Travelling Salesman problem, Min-max principle	4	10
6	String Matching: Introduction, The naive string, matching algorithm, The Rabin-Karp algorithm, String Matching with finite automata	3	10
7	Introduction to NP-Completeness: The class P and NP, Polynomial reduction, NP-Completeness Problem, NP-Hard Problems	3	10

Text Books:

1. Introduction to Algorithms. Thomas Cormen, Charles Leiserson, Ronald Rivest. PHI publication
2. Fundamentals of Algorithms. Gilles Brassard, Paul Bratley. PHI publication.

Reference Books:

1. Design Analysis and Algorithms by Hari Mohan Pandey
2. Introduction to Design and Analysis of Algorithms, Anany Levitin, Pearson.
3. Design and Analysis of Algorithms, Dave and Dave, Pearson.
4. Fundamentals of Algorithms.-E. Horowitz et al.

List of Open Source Software/learning website:

1. <http://nptel.ac.in/>

LAB/Practical

1. Implementation and Time analysis of sorting algorithms.
Bubble sort, Selection sort, Insertion sort, Merge sort and Quick sort
2. Implementation and Time analysis of linear and binary search algorithm.
3. Implementation and Time analysis of factorial program using iterative and recursive method
4. Implementation of a knapsack problem using dynamic programming
5. Implementation of chain matrix multiplication using dynamic programming.
6. Implementation of making change problem using dynamic programming
7. Implementation of a knapsack problem using greedy algorithm
8. Implement prim's algorithm
9. Implement kruskal's algorithm.
10. Implement LCS problem

Course Learning Outcomes (CLO): On completion of this course, the students will be able to:

CLO	Description	Bloom's Taxonomy Level
CLO1	Understand the basic concepts of algorithms and able to compare functions using asymptotic analysis and describe the relative merits of worst-, average-, and best-case analysis	1 Remembering 2 Understanding
CLO2	Familiar with a variety of sorting algorithms and their performance characteristics (eg, running time, stability, space usage) and be able to choose the best one under a variety of requirements.	2 Understanding
CLO3	Understand and identify the performance characteristics of fundamental algorithms and data structures and be able to trace their operations for problems such as sorting, searching, selection, operations on numbers, matrices, and graphs.	3 Applying 4 Analyze
CLO4	Be able to use the design techniques introduced i.e. dynamic programming, greedy algorithm etc. to design algorithms for more complex problems and analyze their performance.	3 Applying
CLO5	Manipulate the different concepts in string matching such as naive string, finite automata and basic knowledge of NP-Completeness.	2 Understanding 3 Applying

Mapping of CLOs with POs & PSOs

Course Learning Outcomes	Program Outcomes (POs)												Program Specific Outcomes(PSOs)			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CLO1	H	M	M	L	L	M			L				H	L		
CLO2		L	M	H	L	L					L	L	M	L		
CLO3	L	M	M	L	M	H		M		L			M	L		
CLO4	M	M	M	L	M	H		M	L				M	L		
CLO5	M	L	L	L	M	H		M		L			H	L		

H: High, M: Medium, L: Low