

SEMESTER S5

DESIGN AND ANALYSIS OF ALGORITHMS

(Common to CS/CD/CM/AM/CB/CN/CU/CG)

Course Code	PCCST502	CIE Marks	40
Teaching Hours/Week (L: T:P: R)	3:1:0:0	ESE Marks	60
Credits	4	Exam Hours	2 Hrs 30 Min.
Prerequisites (if any)	PCCST303	Course Type	Theory

Course Objectives:

- To gain a foundational understanding of algorithms and their analysis.
- To develop problem-solving skills using various algorithm design paradigms like divide and conquer, dynamic programming, etc.
- To understand the concepts of tractable and intractable problems, and different complexity classes (P, NP, NP-hard, NP-complete).

SYLLABUS

Module No.	Syllabus Description	Contact Hours
1	Algorithms – Characteristics, Criteria for Analysing Algorithms; Time and Space Complexity - Best, Worst, and Average Case Complexities; Asymptotic Notations and their properties; Time and Space Complexity Calculation of simple algorithms; Analysis of Recursive Algorithms - Recurrence Equations, Solution of Recurrence Equations : Iteration Method, Recursion Tree Method, Substitution method and Master's Theorem (proof not expected); Balanced Search Trees - AVL Trees (Insertion and deletion operations with all rotations in detail, algorithms not expected)	11

2	Disjoint Sets - Disjoint set operations, Union and find algorithms, Analysis of union by rank with path compression, Connected components of a Graph; Graphs – Representations, Traversals : BFS, DFS and their analysis, Strongly Connected Components; Topological Sorting. Divide and Conquer Strategy – Control Abstraction, Merge Sort, Strassen’s Matrix Multiplication, Analysis.	11
3	Greedy Strategy - Control Abstraction, Fractional Knapsack; Minimum Cost Spanning Tree – Kruskal’s and Prim’s, Analysis; Shortest Path Problem – Dijkstra’s Algorithm, Analysis; Dynamic Programming - Control Abstraction, Optimality Principle, Matrix Chain Multiplication, Analysis; All Pairs Shortest Path Algorithm - Floyd-Warshall Algorithm, Analysis; Backtracking - Control Abstraction, N – Queens Problem, Algorithm.	11
4	Branch and Bound - Control Abstraction, Travelling Salesman Problem, Algorithm; Complexity - Tractable and Intractable Problems; Complexity Classes : P, NP, NP- Hard and NP-Complete Classes; NP Completeness proof - Clique Problem and Vertex Cover Problem; Approximation algorithms - Bin Packing; Randomized Algorithms - Definitions of Monte Carlo and Las Vegas algorithms; Randomized version of Quick Sort algorithm with analysis.	11

Course Assessment Method
(CIE: 40 marks, ESE: 60 marks)

Continuous Internal Evaluation Marks (CIE):

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written)	Total
5	15	10	10	40

End Semester Examination Marks (ESE)

In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions

Part A	Part B	Total
<ul style="list-style-type: none"> • 2 Questions from each module. • Total of 8 Questions, each carrying 3 marks <p>(8x3 =24 marks)</p>	<ul style="list-style-type: none"> • Each question carries 9 marks. • Two questions will be given from each module, out of which 1 question should be answered. • Each question can have a maximum of 3 subdivisions. <p>(4x9 = 36 marks)</p>	60

Course Outcomes (COs)

At the end of the course students should be able to:

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Analyze any given algorithm and express its time and space complexities in asymptotic notations.	K4
CO2	Solve the recurrence equations using Iteration, Recurrence Tree, Substitution and Master's Method to compute time complexity of algorithms.	K3
CO3	Illustrate the operations of advanced data structures like AVL trees and Disjoint sets.	K3
CO4	Illustrate the representation, traversal and different operations on Graphs.	K3
CO5	Demonstrate Divide-and-conquer, Greedy Strategy, Dynamic programming, Branch-and Bound and Backtracking algorithm design techniques.	K2
CO6	Classify a problem as computationally tractable or intractable, and discuss strategies to address intractability.	K4

Note: K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create

CO-PO Mapping Table (Mapping of Course Outcomes to Program Outcomes)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2								3
CO2	3	3	3									2
CO3	3	3	3									3
CO4	3	3	3									3
CO5	3	3	3	2								2
CO6	3	3	3	2								2

Note: 1: Slight (Low), 2: Moderate (Medium), 3: Substantial (High), -: No Correlation

Text Books				
Sl. No	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
1	Introduction to Algorithms	T.H.Cormen, C.E.Leiserson, R.L.Rivest, C. Stein,	Prentice-Hall India	4/e, 2018
2	Fundamentals of Computer Algorithms	Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran,	Orient Longman Universities Press	2/e, 2008
3	Computer Algorithms, Introduction to Design and Analysis	Sara Baase and Allen Van Gelder	Pearson Education	3/e, 2009

Reference Books				
Sl. No	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
1	Design and Analysis of Algorithms	Michael T. Goodrich Roberto Tamassia	Wiley	1/e, 2021
2	Algorithm Design	Jon Kleinberg, Eva Tardos	Pearson Education	1/e, 2005
3	Algorithms	Robert Sedgewick, Kevin Wayne	Pearson Education	4/e, 2011
4	Fundamentals of Algorithmics	Gilles Brassard, Paul Bratly	Pearson Education	1/e, 1996
5	The Algorithm Design Manual	Steven S. Skiena	Springer	2/e, 2008

Video Links (NPTEL, SWAYAM...)	
Module No.	Link ID
1	https://archive.nptel.ac.in/courses/106/106/106106131/
2	https://www.coursera.org/learn/dynamic-programming-greedy-algorithms
3	https://online.stanford.edu/courses/soe-ycsalgorithms1-algorithms-design-and-analysis-part-1
4	https://online.stanford.edu/courses/soe-ycs0001-algorithms-design-and-analysis-part-2