DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) & OBE Scheme]

SEMESTER – IV

Course Code:	**P21IS403**	**Credits:**	**03**
Teaching Hours/Week (L:T:P):	**3:0:0**	**CIE Marks:**	**50**
Total Number of Teaching Hours:	**40**	**SEE Marks:**	**50**

Prerequisites: Students should have knowledge of Programming language and Data structures.

Course Learning Objectives: This coursewill enable students to:

- Explain various computational problem-solving techniques.
- Apply appropriate method to solve a given problem.
- Describe various methods of algorithm analysis.

UNIT - I	**8 Hours**

Introduction: Algorithm, Fundamentals of Algorithmic problem solving, Important Problem Types, Fundamental Data Structures - Graphs, Fundamentals of the **Analysis of Algorithm Efficiency**: Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical analysis of Non-Recursive Algorithms with Examples [Max Element, Unique Elements]and Recursive Algorithms with Examples [Factorial, Tower of Hanoi].

Self-study component:	Additional Examples of Mathematical analysis of Non-Recursive& Recursive Algorithms.

UNIT - II	**8 Hours**

Brute Force and Exhaustive Search: Selection Sort, Brute-Force String Matching,Exhaustive Search [Traveling Salesman Problem and Knapsack Problem].**Decrease and Conquer**: Introduction, Insertion Sort, Depth First Search, Breadth FirstSearch, Topological Sorting, Algorithms for Generating Combinatorial Objects.

Self-study component:	Bubble Sort and Sequential Search.

UNIT - III	**8 Hours**

Divide and Conquer: Merge sort, Quick Sort, Multiplication of Large integers and Strassen' Matrix Multiplication.**Transform and Conquer**: Presorting, Balanced Search Trees, Heaps and Heap sort.

Self-study component:	Binary Tree Traversals and Related Properties.

UNIT - IV	**8 Hours**

Space and Time Tradeoffs: Sorting by counting, Input Enhancement in String Matching, Hashing. **Dynamic Programming**: Three Basic Examples, the Knapsack Problem, Warshall's and Floyd's Algorithms.

Self-study component:	B-Trees, Optimal Binary Search Trees.

UNIT - V	8 Hours
Greedy Technique: Kruskal’s Algorithm, Prim’s Algorithm, Dijikstra’s Algorithm. Limitations of Algorithm Power: P, NP and NP- Complete Problems. Coping with theLimitations of Algorithm Power: Backtracking: n-Queens Problem, Subset-Sum Problem, Branch and Bound: Knapsack Problem.	
Self-study component:	Lower Bound Arguments, Decision trees.
Course Outcomes: On completion of this course, students are able to:	
CO1	.Understand the basic concepts of various algorithmic techniques
CO2	Analyze the asymptotic performance of algorithms
CO3	Design solutions for the given problem using algorithmic technique.

Text Book(s):

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin, 3rd Edition, 2011. Pearson.

Reference Book(s):

1. Computer Algorithms/C++, Ellis Horowitz, SatrajSahni and Rajasekaran, 2nd Edition, 2014, Universities Press.
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.

Web and Video link(s):

1. Algorithms: Design and Analysis, Part 1 (Coursera) | MOOC List (mooc-list.com)
2. https://onlinecourses.nptel.ac.in/noc15 _cs02/preview

CO-PO Mapping

CO	Statement	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	**Understand** the basic concepts of various algorithmic techniques	3											
CO2	**Analyze** the asymptotic performance of algorithms	1	2										
CO3	**Design** solutions for the given problem using algorithmic technique.	1	2	2									