INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal - 500 043, Hyderabad, Telangana

COURSE CONTENT

DESIGN AND ANALYSIS OF ALGORITHMS								
IV Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / IT								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
ACSD13	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	40	60	100
Contact Classes: 48	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 48**		
Prerequisite: Programming for Problem Solving, Data Structures								

I. COURSE OVERVIEW:

Design and analysis of algorithms is the process of finding the computational complexity of algorithms. It helps to design and analyze the logic on how the algorithm will work before developing the actual code for a program. It focuses on introduction to algorithm, asymptotic complexity; sorting and searching using divide and conquer, greedy method, dynamic programming, backtracking, branch and bound. NP-hard and NP-complete problems. The applications of algorithm design are used for information storage, retrieval, transportation through networks, and presentation to users.

II. COURSES OBJECTIVES:

The students will try to learn

I. Mathematical approach for Analysis of Algorithms.
II. Methods and techniques for analyzing the correctness and resource requirements of algorithms.
III. Different paradigms of algorithm design including recursive algorithms, divide-and-conquer algorithms, dynamic programming, greedy algorithms, Backtracking, Branch and Bound and graph algorithms.
IV. Strategies for solving problems not solvable in polynomial time.

III. COURSE OUTCOMES:

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

CO1 Find the (worst case, randomized, amortized) running time and space complexity of given algorithms using techniques such as recurrences and properties of probability.
CO2 Apply divide and conquer algorithms for solving sorting, searching and matrix multiplication.
CO3 Make Use of appropriate tree traversal techniques for finding shortest path.
CO4 Compare Identify suitable problem solving techniques for a given problem and finding optimized solutions using Greedy and Dynamic Programming techniques
CO5 Apply greedy algorithm Utilize backtracking and branch and bound techniques to deal with traceable and in-traceable problems.
CO6 Apply Describe the classes P, NP, NP-Hard, and NP- complete for solving deterministic and non-deterministic problems.

IV. COURSE CONTENT:

MODULE – I: INTRODUCTION (10)

Algorithm: Pseudo code for expressing algorithms; Performance analysis: Space complexity, time complexity; Asymptotic notations: Big O notation, omega notation, theta notation and little o notation, amortized complexity; Divide and Conquer: General method, binary search, quick sort, merge sort, Stassen's matrix multiplication.

MODULE – II: SEARCHING AND TRAVERSAL TECHNIQUES (09)

Disjoint set operations, union and find algorithms; Efficient non recursive binary tree traversal algorithms, spanning trees; Graph traversals: Breadth first search, depth first search, connected components,

biconnected components.

MODULE – III: GREEDY METHOD AND DYNAMIC PROGRAMMING (10)

Greedy method: The general method, job sequencing with deadlines, knapsack problem, minimum cost spanning trees,single source shortest paths.

Dynamic programming: The general method, matrix chain multiplication optimal binary search trees, 0/1 knapsack problem, single source shortest paths, all pairs shortest paths problem, the travelling salesperson problem.

MODULE – IV: BACKTRACKING AND BRANCH AND BOUND (09)

Backtracking: The general method, the 8 queens problem, sum of subsets problem, graph coloring, Hamiltonian cycles; Branch and bound: The general method, 0/1 knapsack problem, least cost branch and bound solution, first in first out branch and bound solution, travelling salesperson problem.

MODULE – V: NP-HARD AND NP-COMPLETE PROBLEMS (09)

Basic concepts: Non-deterministic algorithms, the classes NP - Hard and NP, NP Hard problems, clique decisionproblem, chromatic number decision problem, Cook's theorem.

V. TEXT BOOKS:

1. Ellis Horowitz, Satraj Sahni, Sanguthevar Rajasekharan, "Fundamentals of Computer Algorithms", Universities Press, 2nd edition, 2015.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D, "The Design And Analysis Of Computer Algorithms", Pearson India, 1st edition, 2013..

VI. REFERENCE BOOKS:

1. Levitin A, "Introduction to the Design and Analysis of Algorithms", Pearson Education, 3rd edition, 2012.
2. Goodrich, M. T. R Tamassia, "Algorithm Design Foundations Analysis and Internet Examples", John Wileyn and Sons, 1st edition, 2001.
3. Base Sara Allen Vangelder, "Computer Algorithms Introduction to Design and Analysis", Pearson, 3rd edition, 1999.

VII. ELECTRONICS RESOURCES:

1. http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html
2. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
3. http://www.facweb.iitkgp.ernet.in/~sourav/daa.html

VIII. MATERIALS ONLINE

1. Course template
2. Tutorial question bank
3. Tech-talk topics
4. Open-ended experiments
5. Definitions and terminology
6. Assignments
7. Model question paper – I
8. Model question paper – II
9. Lecture notes
10. PowerPoint presentation
11. E-Learning Readiness Videos (ELRV)