Design and Analysis of Algorithms

B.Tech – IV Semester (20IT404)

Lectures	:	3 Hours / Week	Tutorial	:	0	Practical	:	0
CIE Marks	:	30	SEE Marks	:	70	Credits	:	3

Prerequisites:

None

Course Objectives:

The purpose of this course is to acquaint the student with an overview of the theoretical foundations of computer science from the perspective of formal languages.

COB 1: Understand about designing and effectiveness of an algorithm, and divide and conquer method.

COB 2: Understand the optimal solution finding with the greedy and dynamic programming method

COB 3: Easy know the major graph algorithms and their analyses, and backtracking information.

COB 4: Get the ability to branch with bound value and NP problems.

Course Outcomes:

After the course the students are expected to be able to

CO 1: Explains Algorithm design and efficiency and master theorem

CO 2: Solve divide and conquer and greedy problems.

CO 3: Design the algorithms like dynamic and graph type tasks.

CO 4: Recognize the solutions for back tacking and branch and bound and also NP problems.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO11	PO12
CO 1	3	2	3	2	3	-	2	-	-	2	2	3
CO 2	2	2	2	2	2	-	2	-	-	2	2	2
CO 3	3	3	3	3	3	-	2	-	-	2	2	3
CO 4	2	2	1	2	2	-	2	-	-	2	2	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO 1	PSO 2	PSO 3
CO 1	3	3	1
CO 2	2	3	1
CO 3	2	3	2
CO 4	2	3	2

UNIT - I **(12 Hours)**

Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh-notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized analysis

Master Theorem: Introduction, Generic Form- Case1, Case2, Case3, Inadmissible equations, Application to common algorithms

UNIT - II **(12 Hours)**

Divide and conquer : General method , applications - Quick sort, Merge sort, Strassen's matrix multiplication.

Greedy method : General method, applications-Job sequencing with deadlines, Fractional knapsack problem, Minimum cost spanning trees - Prims, Kruskal, Single source shortest path problem - Dijkstra.

UNIT - III **(12 Hours)**

Dynamic Programming: General method, applications - 0/1 knapsack problem, Travelling salesperson problem, Longest common sequence algorithm, Multistage graphs using Forward & Backward approach, Reliability design.

Graph Searching and Traversal: Graph traversals - Depth first, Breadth first, Bio Connected Components, Strongly Connected Components.

UNIT - IV **(12 Hours)**

Back tracking: General method, applications-n-queen problem, sum of subsets problem.

Branch and Bound: General method, applications - 0/1 knapsack problem- LC Branch and Bound solution.

NP-Hard and NP-Complete problems: Basic concepts, non deterministic algorithms, NP-Hard and NP Complete classes, Cook's theorem.

TEXT BOOKS:

1. S. Sahni E. Horowitz and S.Rajsekran. *Fundamentals of Computer Algorithms*. Orient Longman, 2 edition, 2018. ISBN 978-8-173-71612-6

REFERENCES:

1. Michael Soltys-kulinicz. *Introduction to the Analysis Of Algorithms*. World Scientific, 3 edition, 2018. ISBN 978-9813235908

2. Anany Levitin. *Introduction to the Design and Analysis of Algorithms*. Pearson, 3 edition, 2017. ISBN 978-9332585485