

<b>COURSE TITLE</b>	<b>DESIGN AND ANALYSIS OF ALGORITHMS</b>
<b>COURSE CODE</b>	<b>01AI0506</b>
<b>COURSE CREDITS</b>	<b>5</b>

**Objective:**

- 1 Student will understand about algorithms and develop efficient and effective solutions to computational problems. This course focuses on studying various algorithmic techniques, their design principles, and their analysis to understand their performance characteristics and capabilities.
- 2 -

**Course Outcomes:** After completion of this course, student will be able to:

- 1 To learn about the process of problem solving with algorithms for common problems.
- 2 To analyse the algorithms for time/space complexity and learn to write algorithms for a given problem using different design paradigms.
- 3 To understand computational complexity of problems.
- 4 Design optimal solution by applying various methods like Dynamic Programming and Greedy Method.
- 5 To understand the concepts behind NP Completeness, Approximation algorithms and randomized algorithms.

**Pre-requisite of course:**Data Structure and proficiency in programming language, knowledge of Mathematical functions like logarithms, graphs etc.

**Teaching and Examination Scheme**

<b>Theory Hours</b>	<b>Tutorial Hours</b>	<b>Practical Hours</b>	<b>ESE</b>	<b>IA</b>	<b>CSE</b>	<b>Viva</b>	<b>Term Work</b>
4	0	2	50	30	20	25	25

<b>Contents : Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
1	<b>Introduction</b> The Role of Algorithms in Computing, Algorithm, Designing Algorithms – Analysing Algorithms - Time and space complexity - Asymptotic Notations and its properties, The efficiency of algorithm - Best case, Worst case and average case analysis.	6
2	<b>ALGORITHM DESIGN TECHNIQUES</b> Recurrence relation: Introduction, methods to solve recurrence, Sorting and searching: Bubble sort, Selection sort, Insertion sort, Shell sort, Heap sort, Binary Search, Linear search Divide and Conquer: Finding maximum and minimum, Merge sort, Quick sort	9

<b>Contents : Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
3	<b>DYNAMIC PROGRAMMING</b> Introduction,, Elements of Dynamic Programming, Problem Solving using Dynamic Programming – Making Change Problem, Assembly Line Scheduling, Knapsack problem, Matrix chain multiplication, Longest Common Subsequence	9
4	<b>GREEDY METHOD AND GRAPH ALGORITHMS</b> General Characteristics of greedy algorithms, Elements of greedy strategy, Problem solving using - Activity selection problem, Fractional Knapsack Problem, Job Scheduling Problem, Huffman Code, Representations of graphs - Graph traversal: DFS – BFS - applications – Connectivity, Minimum spanning tree: Kruskal’s and Prim’s algorithms, Shortest path Algorithms: Dijkstra’s algorithm - Floyd-Warshall algorithm	10
5	<b>BACKTRACKING AND BRANCH &amp; BOUND</b> Backtracking: n-Queens problem , Hamiltonian Circuit Problem , Subset Sum Problem , Graph colouring problem, Travelling Salesman Problem, Branch and Bound: Assignment problem, Knapsack Problem	9
6	<b>NP-COMPLETE AND APPROXIMATION ALGORITHM</b> Tractable and intractable problems: Polynomial time algorithms , Venn diagram representation , NP algorithms , NP hardness and NP-completeness, Bin Packing problem, Problem reduction: TSP – 3- CNF problem., Approximation Algorithms: TSP , Randomized Algorithms: concept and application, primality testing - randomized quick sort, Finding kth smallest number	9
<b>Total Hours</b>		<b>52</b>

#### Suggested List of Experiments:

<b>Contents : Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
1	<b>Practical 1</b> Implementation and Time analysis of sorting algorithms. Bubble sort, Selection sort, Insertion sort, Merge sort and Quicksort	2
2	<b>Practical 2</b> Implementation and Time analysis of linear and binary search algorithm.	2
3	<b>Practical 3</b> Implementation of max-heap sort algorithm	2
4	<b>Practical 4</b> Implementation and Time analysis of factorial program using iterative and recursive method	2
5	<b>Practical 5</b> Implementation of a knapsack problem using dynamic programming.	2

### Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
6	<b>Practical 6</b> Implementation of chain matrix multiplication using dynamic programming.	2
7	<b>Practical 7</b> Implementation of making a change problem using dynamic programming	2
8	<b>Practical 8</b> Implementation of Graph and Searching (DFS and BFS).	2
9	<b>Practical 9</b> Implement prim's algorithm.	2
10	<b>Practical 10</b> Implement Kruskal's algorithm.	2
11	<b>Practical 11</b> Write a program for Floyd-Warshal algorithm.	2
12	<b>Practical 12</b> Write a program for travelling salesman problem.	2
<b>Total Hours</b>		<b>24</b>

### Textbook :

- 1 Introduction to Algorithms , Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein , PHI , -

### References:

- 1 Introduction to Algorithms, Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, PHI, -
- 2 Fundamental of Algorithms by Gills Brassard, Fundamental of Algorithms by Gills Brassard, Paul Bratley, PHI, -
- 3 Introduction to Design and Analysis of Algorithms, Introduction to Design and Analysis of Algorithms, Anany Levitin, Pearson, -
- 4 Foundations of Algorithms, Foundations of Algorithms, Shailesh R Sathe, Penram, -
- 5 Design and Analysis of Algorithms, Design and Analysis of Algorithms, Dave and Dave, Pearson, -

### Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
Remember / Knowledge	Understand	Apply	Analyze	Evaluate	Higher order Thinking / Creative
0.00	20.00	40.00	20.00	20.00	0.00

**Instructional Method:**

- 1 The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.
- 2 The internal evaluation will be done on the basis of continuous evaluation of students in the laboratory and class-room.
- 3 Practical examination will be conducted at the end of semester for evaluation of performance of students in laboratory.
- 4 Students will use supplementary resources such as online videos, NPTEL videos, e-courses, Virtual Laboratory.

**Supplementary Resources:**

- 1 <http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html>
- 2 <http://nptel.ac.in/courses/106101060/>
- 3 <http://www.comp.nus.edu.sg/~cs5234/Links/Course-Links.htm>
- 4 <https://www.coursera.org/learn/algorithm-design-analysis>
- 5 <http://apps.topcoder.com/wiki/di...>
- 6 <http://www.geeksforgeeks.org>,
- 7 <http://www.algolist.net>
- 8 <http://www.cprogramming.com>