

COURSE TITLE	DESIGN AND ANALYSIS OF ALGORITHM
COURSE CODE	01CT0512
COURSE CREDITS	4

Objective:

- 1 Obtaining efficient algorithms is very important in modern computer engineering as the world wants applications to be time and space and energy efficient. This course enables to understand and analyse efficient algorithms for various applications.
- 2 Obtaining efficient algorithms is very important in modern computer engineering as the world wants applications to be time and space and energy efficient. This course enables to understand and analyze efficient algorithms for various applications.
- 3 Obtaining efficient algorithms is very important in modern computer engineering as the world wants applications to be time and space and energy efficient. This course enables to understand and analyze efficient algorithms for various applications.

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

- 1 Compare various algorithm design techniques for developing algorithms by evaluating the asymptotic complexities and time-space trade-off.
- 2 Develop different algorithms using various methods like dynamic and Greedy methods.
- 3 Select appropriate pattern matching algorithm to develop model for substring, subsequence, DNA matching, etc.
- 4 Distinguish between Polynomial, Non-Polynomial complete and Hard problems
- 5 Evaluate various graph algorithms for sparse and dense network structures

Pre-requisite of course: Proficiency in any programming language, Data Structures, Discrete Mathematics.

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
3	0	2	50	30	20	25	25

Contents : Unit	Topics	Contact Hours
1	Introduction to Design and Analysis of Algorithms What is an algorithm, Mathematics for Algorithmic Sets, Functions and Relations, Vectors and Matrices, Linear Inequalities and Linear Equations	2

Contents : Unit	Topics	Contact Hours
2	Asymptotic Analysis of Algorithm The efficient algorithm,, Average, Best- and worst-case analysis,, Amortized analysis, Asymptotic Notations (Big Oh, Big Theta, Big Omega),, Master Method, Sorting Algorithms and their analysis,, Sorting in linear time: Bucket sort, Radix sort and Counting sort	6
3	Divide and Conquer Introduction, Recurrence and different methods to solve recurrence, multiplying large Integers Problem,, Problem Solving using divide and conquer algorithm - Binary Search, Max-Min problem,, Sorting (Merge Sort, Quick Sort), Matrix Multiplication, Exponential	6
4	Dynamic Programming Introduction, Elements of Dynamic Programming, The Principle of Optimality, Problem Solving using Dynamic Programming – Calculating the Binomial Coefficient,, Making Change Problem,, Assembly Line-Scheduling, Knapsack problem,, Matrix chain multiplication, Longest Common Subsequence	6
5	Greedy Algorithm General Characteristics of greedy algorithms, Elements of greedy strategy, Problem solving using - Activity selection problem, , Fractional Knapsack Problem, Job Scheduling Problem.	4
6	Graph Algorithms Representation of Undirected & Directed Graph, Traversing Graphs, Depth First Search, Breath First Search, Topological sort, Strongly Connected components, Single pair shortest path and Minimum Spanning trees (Kruskal's algorithm, Prim's algorithm) using greedy approach, All Points Shortest path using Dynamic Programming	6
7	Backtracking and Branch and Bound Introduction, The Eight queens' problem, Knapsack problem,, Travelling Salesman problem, Minmax principle	4
8	String Matching Introduction, The naive string-matching algorithm, The Rabin-Karp algorithm,, String Matching with finite automata	4
9	Introduction to NP-Completeness The class P and NP, Polynomial reduction,, 2-CNF Satisfiability, 3-CNF Satisfiability, , NP- Completeness Problem, NP-Hard Problems, Travelling Salesman problem, Hamiltonian problem	4
Total Hours		42

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	Linear Sorting Algorithms Implementation and Time analysis of sorting algorithms, Insertion sort, Bubble sort, Selection sort,	4

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
2	Divide and Conquer Algorithms Merge sort, Quicksort, Binary Search, Implementation of max-heap sort algorithm	6
3	Method comparison Implementation and Time analysis of factorial program using iterative and recursive method	2
4	Dynamic Programming Implementation of a knapsack problem using dynamic programming, Implementation of chain matrix multiplication using dynamic programming, Implementation of making a change problem using dynamic programming	6
5	Greedy Algorithm Implementation of a knapsack problem using greedy algorithm, Implementation of a fractional knapsack problem using greedy algorithm	4
6	Graph Algorithm Implementation of Graph and Searching (DFS and BFS), Prim and Kruskal	4
7	String Algorithms To implement following string-matching algorithms and analyze time complexities: a. Naïve b. Rabin Karp c. Knuth Morris Pratt, LCS	4
8	Graph Algorithm Floyd-Warshal, Dijkstra, travelling salesman, Hamiltonian cycle	4
9	Othr algorithms Strassen's Matrix Multiplication. Huffman coding	2
Total Hours		36

Textbook :

- 1 Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, The MIT Press, 2012
- 2 Fundamental of Algorithmics, Gills Brassard and Paul Bratley, Prentice-Hall International, 2011

References:

- 1 Introduction to Design and Analysis of Algorithms, Introduction to Design and Analysis of Algorithms, Anany Levitin, Pearson, 2021
- 2 Design and Analysis of Algorithms, Design and Analysis of Algorithms, Dave and Dave, Pearson, 2007

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
10.00	10.00	40.00	20.00	10.00	10.00

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://interactivepython.org/runestone/static/pythonds/index.html>
- 2 <http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html>
- 3 <http://nptel.ac.in/courses/106101060/>
- 4 <http://www.codeskulptor.org/docs.html><http://www.geeksforgeeks.org>
- 5 <https://www.coursera.org/learn/algorithm-design-analysis>