**14P021/14F023/14L013/14T013/14E027/14N024 -DESIGN AND ANALYSIS OF ALGORITHMS**

**3 0 0 3**

**OBJECTIVES:**

* To learn the basic concepts of algorithm, its performance analysis and notations
* To study and analyze algorithms mathematically
* To demonstrate and analyze the known algorithms and data structures
* To apply important algorithmic design paradigms and methods of analysis
* To synthesize efficient algorithm in common engineering design
* To study about NP-completeness, Approximation and Randomized algorithms

**OUTCOMES:**

Learner should be able to

* Understand the basic notations of algorithm analysis and introduction to algorithms
* Analyze algorithms mathematical analysis of recursive and non-recursive algorithms
* Analyze the known algorithms and data structures efficiently
* Able to apply important problem solving paradigms to engineering problems.
* Analyze the NP-completeness problems efficiently and the benefits of using Approximation algorithms.

**MODULE I [9 HOURS]**

**Mathematical Background:** Summations-Sets-Relations-Functions-Graphs-Trees-Counting and Probability-Matrices.

**Algorithm Analysis:** Introduction to Algorithm - Notion of Algorithm – Euclids Algorithm - Reduction of Algorithm to Curve-Growth Rate Analysis-Asymptotic Analysis - Asymptotic Notations – Amortized analysis - Recurrence Equations - Analysis of Recursive and Non- Recursive Algorithms – General Method

**MODULE II [18 HOURS]**

**Divide And Conquer:** Brute Force: Selection Sort, Bubble Sort - Divide And Conquer: General Method – Binary Search – Min-Max Problem – Median Finding - Merge Sort - Quick Sort – Integer Multiplication - Strassen’s Matrix Multiplication-Convex Hull Problem

**Greedy Algorithms:** General Method – Fractional Knapsack – Job sequencing with deadlines - Huffman Coding - Minimum cost spanning trees, Single source shortest path problem.

**Transform and Conquer:** Binary Tree- BST – AVL Tree Operations – Heaps - Heap Sort-Horner’s Rule

**MOUDLE III [18 HOURS]**

**Dynamic Programming**-General Method - Principle of Optimality- Making Change Problem - Assembly Line scheduling - 0/1 Knapsack - Travelling Salesman Problem - Longest Common Subsequence – Optimal Search Tree- Matrix Chain Multiplication – A machine Scheduling Problem- Case-Studies.

**Back Tracking:** General Method – 8 Queens Problem – Sum of Subsets – Graph Colouring – Hamiltonian problems.

**Randomization: Introduction to randomized algorithms -** Random numbers, randomized Quick sort, Min cut problem

**NP-Completeness and Approximation**: The Class P and NP - Polynomial Time reduction - NP-completeness - NP-Hard problems – Hamiltonian cycle - Travelling Salesman problem- Approximation algorithms

**Total: 45**

**TEXT BOOK**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Author(s)** | **Title of Book** | **Publisher** | **Year of Publication** |
| 1. | Cormen T H,  Leiserson C E,  Rivest RL ,Stein C | Introduction to Algorithms | MIT Press, Third Edition | 2009 |
| 2. | Anany Levitin | Introduction to Design and Analysis of Algorithm | Pearson Education | 2011 |
| 3. | Jeff Edmonds | How to Think about Algorithms | Cambridge University Press | 2008 |

**REFERENCES:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Author(s)** | **Title of Book** | **Publisher** | **Year of Publication** |
| 2. | Sara Baase  Allen Van Gelder | Introduction to the Design and Analysis | Pearson Education | 1999 |
| 3. | Dasgupta | Algorithms | Tata McGraw Hill | 2006 |
| 4. | Kenneth A Berman Jerome L Paul | Algorithms | Cengage Learning | 2004 |

**WEB URLs:**

1. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
2. http://lcm.csa.iisc.ernet.in/dsa/dsa.html
3. http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2012/lecture-notes/
4. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
5. http://nptel.ac.in/course.php