**III Year – II SEMESTER**

| T | P | C |
|---|---|---|
| 4 | 0 | 3 |

# Design and Analysis of Algorithms

**Course Objectives:**

Upon completion of this course, students will be able to do the following:

- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations.

**Course Outcomes:**

Students who complete the course will have demonstrated the ability to do the following:

- Analyze worst-case running times of algorithms using asymptotic analysis.
- Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it.
- Describe the greedy paradigm and explain when an algorithmic design situation calls for it.
- Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate. Synthesize new graph algorithms and algorithms that employ graph computations as key components, and analyze them.
- Explain the different ways to analyze randomized algorithms (expected running time, probability of error). Recite algorithms that employ randomization. Explain the difference between a randomized algorithm and an algorithm with probabilistic inputs.
- Analyze randomized algorithms. Employ indicator random variables and linearity of expectation to perform the analyses. Recite analyses of algorithms that employ this method of analysis.

**Syllabus:**

**UNIT-I:**
Introduction: Algorithm, Psuedo 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.

**UNIT-II:**
Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort

**UNIT-III:**

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

**UNIT-IV:**
Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

**UNIT-V:**
Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

**UNIT-VI:**
Branch and Bound: General method, applications - Travelling sales person problem,0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution.

**TEXT BOOKS:**

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. Design and Analysis of Algorithms , S Sridhar, Oxford
3. Design and Analysis of Algorithms, Parag Himanshu Dave, Himansu BAlachandra Dave, 2ed,Pearson Education.

**REFERENCE BOOKS:**

1. Design and Analysis of algorithms, Aho, Ullman and Hopcroft,Pearson education.
2. Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA
3. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein,PHI Pvt. Ltd.
4. Algorithm Design, Foundation, Analysis and internet Examples, Michel T Goodrich, Roberto Tamassia, Wiley