**CS/IT313 - DESIGN & ANALYSIS OF ALGORITHMS**

| Lectures | : | 2 periods/week | Internal Marks | : | 30 |
| --- | --- | --- | --- | --- | --- |
| Tutorial: | : | 1 period/week | Semester End Examination Marks | : | 70 |
| Sem End Exam Duration |  | 3 Hours | Credits | : | 3 |

**Course Objectives:**

The main objectives of this course are to:

1. Impart knowledge on algorithm design strategies and performance analysis of algorithms.
2. Introduce pattern matching algorithms and NP-Completeness

**Course Outcomes**

After successful completion of the course, students will be able to:

1. Analyze the performance of algorithms based on time and space complexities.
2. Apply algorithm design strategies to solve the real world problems.
3. Use string matching algorithms to solve given problems.
4. Differentiate P and NP class problems.

**Course Content:**

| **UNIT I** | **10 Periods** |
| --- | --- |

**Introduction**- What is an Algorithm? Algorithm Specification, Performance Analysis, Randomized Algorithms – Identifying the repeated element, primality testing, advantages and disadvantages.

**Divide and Conquer:** General Method, Merge Sort, Quick sort, Divide and Conquer Run Time Recurrence Relations.

| **UNIT II** | **15 Periods** |
| --- | --- |

**Greedy Programming:** General Method, Knapsack problem, Job Sequencing with Dead Lines, Minimum Spanning Tree - Prim's and Kruskal's algorithms, Single Source Shortest-Paths-Dijkstra's.

**Dynamic Programming:** General Method, Multi Stage Graph, All Pairs Shortest Paths, Single Source Shortest Paths-general Weights, Optimal Binary Search Trees, 0/1 Knapsack, Traveling Salesman Problem.

| **UNIT III** | **13 Periods** |
| --- | --- |

**Back tracking:** General Method, 8-queen problem, Hamiltonian Cycles, 0/1 Knapsack.

**Branch and Bound:** Control Abstraction for LC Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem, Traveling Salesman Problem

| **UNIT IV** | **12 Periods** |
| --- | --- |

**String Matching** – The Naïve String Matching Algorithm, The Rabin-Karp Algorithm, String Matching with Finite Automata, The KMP Algorithm.

**NP-Completeness** - Polynomial Time, Polynomial Time verification, NP Completeness and reducibility, NP Complete Problems.

Approximation Algorithms - The Travelling Sales Person Problem.

**Learning Resources:**

**Text Book:**

1. E. Horowitz, S. Sahni and S.Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia Publication. (Unit I, II, III).
2. T. H. Cormen, Leiserson, Rivest and Stein, "Introduction of Computer Algorithm", PHI. (Unit IV).

**Reference Book(s):**

1. Sara Basse, A.V. Gelder, "Computer Algorithms", Addison Wesley.