DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2016 -2017)

SEMESTER – IV

Subject Code: 15CS43 **IA Marks 20**

Number of Lecture Hours/Week: 04 **Exam Marks 80**

Total Number of Lecture Hours :50 **Exam Hours 03**

CREDITS – 04

Course objectives: This course will enable students to

- Explain various computational problem solving techniques.
- Apply appropriate method to solve a given problem.
- Describe various methods of algorithm analysis.

Module 1 10 Hours

Introduction: What is an Algorithm? (T2:1.1), Algorithm Specification (T2:1.2), Analysis Framework (T1:2.1), Performance Analysis: Space complexity, Time complexity (T2:1.3). Asymptotic Notations: Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ), and Little-oh notation (o), Mathematical analysis of Non-Recursiveand recursive Algorithms with Examples (T1:2.2, 2.3, 2.4). Important Problem Types: Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. Fundamental Data Structures: Stacks, Queues, Graphs, Trees, Sets and Dictionaries.(T1:1.3,1.4)

Module 2 10 Hours

Divide and Conquer: General method, Binary search, Recurrence equation for divideand conquer, Finding the maximum and minimum (T2:3.1, 3.3, 3.4), Merge sort, Quicksort (T1:4.1,

4.2), Strassen's matrix multiplication (T2:3.8), Advantages and Disadvantages of divide and conquer. Decrease and Conquer Approach Topological Sort. (T1:5.3)

Module 3 **10 Hours**

Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines (T2:4.1, 4.3, 4.5). Minimum cost spanning trees: Dijkstra's Algorithm (T1:9.3). Optimal Tree problem: Huffman Trees and Codes (T1:9.4). Transform and Conquer Approach: Heaps and Heap Sort (T1:6.4).

Module 4 **10 Hours**

Dynamic Programming: General method with Examples, Multistage Graphs (T2:5.1, 5.2). Transitive Closure: Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem ((T1:8.2, 8.3, 8.4), Bellman-Ford Algorithm (T2:5.4), Travelling Sales Person problem (T2:5.9), Reliability design (T2:5.8).

Module 5 **10 Hours**

Backtracking: General method (T2:7.1), N-Queens problem (T1:12.1), Sum of subsetsproblem (T1:12.1), Graph coloring (T2:7.4), Hamiltonian cycles (T2:7.5). Branch andBound: Assignment Problem, Travelling Sales Person problem (T1:12.2), 0/1Knapsack problem (T2:8.2, T1:12.2): LC Branch and Bound solution (T2:8.2), FIFOBranch and Bound solution (T2:8.2). NP-Complete and NP-Hard problems: Basicconcepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes(T2:11.1).

Course Outcomes: After studying this course, students will be able to

• Describe computational solution to well known problems like searching, sorting etc.

• Estimate the computational complexity of different algorithms.

• Devise an algorithm using appropriate design strategies for problem solving.

Graduate Attributes

• Engineering Knowledge

• Problem Analysis

• Design/Development of Solutions

• Conduct Investigations of Complex Problems

• Life-Long Learning

Question paper pattern:

The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

T1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.

T2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest,

Clifford Stein, 3rd Edition, PHI

2. Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education)