Advanced Data Structures & Algorithms

Code: CS33113CS	Course Type: Core	Semester: I
Evolution Scheme (TA-MSE-ESE): 20-30-100	Total Marks: 150	L-T-P (Credits): 2-0-2 (3)

Pre-Requisites: Data Structures, Algorithm Design and Analysis and Programming

Course Objective: To provide the foundations of the practical implementation and usage of Algorithms and Data Structures.

Course Outcomes: At the end of the course, the student will be able to

CO-1	Detect and apply the algorithmic structures in many different fields of engineering
CO-2	Comprehend and select algorithm design approaches in a problem specific manner
CO-3	Effectively combine fundamental data structures and algorithmic techniques in building a complete algorithmic solution to a given problem
CO-4	Apply data structures to improve the running time of basic algorithmic questions involving data represented as graphs, lists and other data structures.
CO-5	Have the knowledge for state of the art development in the field of algorithms.

Course Articulation Matrix:

PO	PEO-1	PEO-2	PEO-3	PEO-4	PEO-5	PEO-6
CO-1	3	3	2	1	3	2
CO-2	2	3	3	2	2	3
CO-3	2	3	2	1	2	1
CO-4	2	2	2	2	3	2
CO-5	2	3	3	1	3	3

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Unit-1: Importance and need of good data structures and algorithms, Growth of a Function , Heaps, M-way Search Tree, AVL Trees, Red-Black Trees, Splay Trees, B trees, B+ Trees, Data Structures for Disjoint Sets, Augmented Data Structures, Self-Adjusting Data Structures, Temporal data structures, efficiency of algorithms; notion of time and space complexity; performance measures for data structures.

Unit-2: Probabilistic Analysis with example, Amortized Analysis with example, Competitive Analysis with example, Internal and External Sorting algorithms like external merge sort, distribution

sorts, Leftist Heaps, Skew Heaps, Binomial Heaps, Lazy binomial Heaps, Fibonacci Heaps

Unit-3: Representation, Type of Graphs, Paths and Circuits: Euler Graphs, Hamiltonian Paths & Circuits; Cut-sets, Connectivity and Separability, Planar Graphs, Isomorphism, Graph Coloring, Covering and Partitioning, Topological sort, Max flow: Ford Fulkerson algorithm, max flow – min cut, Dynamic Graphs, Few Algorithms for Dynamic Graphs, Union-Find Algorithms.

Unit-4: Approximation Algorithms: Need of approximation algorithms: Introduction to P, NP, NP-Hard and NP-Complete; Deterministic, non-Deterministic Polynomial time algorithms; Las Vegas and Monte Carlo algorithms, Quad trees, K-D trees, TV-trees, Segment trees, Static and Dynamic, Range sum query, Range update query, Range minimum query, Lazy propagation, Geometric algorithms, line segment intersection, Map overlay detection, Voronoi diagram.

Learning Resources:

Text and Reference Books:

1. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms,"Prentice Hall India, 2nd Edition, 2001. ISBN 978-0-262-53305-82.
2. E. Horowitz, S. Sahni, and S. Rajasekaran, "Computer Algorithms," 2 ndEdition, Galgotia Publications, 2007. ISBN 0-7167-8316-9
3. Aho, Hopcroft, and Ullmann, "Data Structures & Algorithms," AddisonWesley, 1983. ISBN13: 9780201000238
4. Algorithm Design , Eva Tardos and Kleinberg, Pearson, 2006, ISBN-13 :978-0321295354
5. Sedgewick & Wayne, Algorithms, Addison-Wesley Professional (March 19, 2011), 4th Edition.
6. Rajeev Motwani and Prabhakar Raghavan, Randomized Algorithms, 2nd Edition,Cambridge University press, Cambridge, MA, 1995.
7. M. J. Quinn, Designing Efficient Algorithms for Parallel Computers, McGraw Hill HigherEducation, 1987, ISBN: 978-0070510715.