COURSE CONTENT

ADVANCED DATA STRUCTURES								
I Semester: CSE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
BCSD02	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	40	60	100
Contact Classes: 45	**Total Tutorials: Nil**	**Total Practical Classes: Nil**				**Total Classes: 45**		

I. COURSE OVERVIEW:

This course covers the overview of advanced data structures for storing and organizing the data. Topics covered include algorithm analysis, dictionaries, hashing, trees, and graphs. The course is to teach the students how to select and apply data structures that are appropriate for problems that they might encounter in problem-solving in engineering areas.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The performance trade-offs of different algorithms/implementations and asymptotic analysis of their running time and memory usage.
II. Choose appropriate data structures, understand the ADT/libraries, and use them to design algorithms for a specific problem
III. The knowledge of basic abstract data types (ADT) and associated algorithms to perform various operations on different types of data structures.
IV. The fundamentals of how to store, retrieve, and process the data efficiently and come up with an analysis of efficiency and proof of correctness.

III. COURSE OUTCOMES:

After successful completion of the course, students should be able to

CO1	Analyze the performance of the algorithms using mathematical tools such as asymptotic notations.	Understand
CO2	Implement ADTs, Lists, and their operations	Apply
CO 3	Construct hash tables by using hash functions and perform various operations on hash tables	Understand
CO 4	Organize data in the form of trees and graphs for retrieving information effectively	Apply
CO 5	Assess the efficiency of a binary search tree (BST), AVL trees, Red Black, and splay trees for searching	Understand
CO 6	Apply the concepts of text compression and pattern matching to solve problems effectively	Apply

IV. COURSE CONTENT:

MODULE-I: OVERVIEW OF DATA STRUCTURES (09)

Algorithm analysis: Algorithms; Performance analysis: Time complexity and space complexity, asymptotic notation: Big Oh, omega and theta notations, complexity analysis examples; Data structures: Linear and nonlinear data structures, ADT concept, linear list ADT, stack and queue ADTs, array and linked list representations; Circular queue: Insertion and deletion, de queue ADT, priority queue ADT, implementation using heaps, insertion into a max heap, deletion from a max heap, singly linked lists, doubly linked lists, circular linked list.

MODULE-II: DICTIONARIES, HASH TABLES (09)
Dictionaries: Linear list representation, operations insertion, deletion and searching, hash table representation, hash functions, collision resolution, separate chaining, open addressing, linear probing, quadratic probing, double hashing, rehashing, extendible hashing.

MODULE-III: TREES AND GRAPHS (09)
Trees: Ordinary and binary tree terminology, properties of binary trees, binary tree ADT, representations, recursive and nonrecursive traversals, threaded binary trees.
Graphs: Graphs terminology, graph ADT, representations, graph traversals; Search methods: DFS and BFS; Applications of Graphs: Minimum cost spanning tree using Kruskal's algorithm, Dijkstra's algorithm for single source shortest path problem.

MODULE-IV: SEARCH TREES I (09)
Binary search tree: Binary search tree ADT, insertion, deletion, and searching operations, finding the parent of a given node, attaining a reference to a node, finding the smallest and largest values in the binary search tree; Balanced search trees: AVL trees, definition, height of an AVL tree; Operations: Insertion, deletion, and searching.

MODULE-IV: SEARCH TREES II (09)
Red-Black and Splay Trees; B trees: Definition, operations, and applications; R trees: Nearest neighbor query, join and range queries; Comparison of search trees; Text compression: Huffman coding and decoding; Pattern matching: KMP algorithm.

V TEXT BOOKS:

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Universities Press Private Limited, India, 2nd Edition, 2008.
2. G.A.V.Pai, "Data Structures and Algorithms", Tata McGraw Hill, NewDelhi,1st Edition,2008
3. Reema Thareja, S. Rama Sree, Advanced Data Structures, Oxford University Press, 2018

VI. REFERENCE BOOKS:

1. D. Samanta, "Classic Data Structures", Prentice Hall of India Private Limited, 2nd Edition, 2003.
2. Aho, Hop craft, Ullman, "Design and Analysis of Computer Algorithms", Pearson Education India, 1st Edition, 1998.
3. Goodman, Hedetniemi, "Introduction to the Design and Analysis of Algorithms", Tata McGraw Hill, New Delhi, India, 1st Edition, 2002.
4. Adam Drozdek, "Data Structures and Algorithms in C++", Thomson Course Technology, 3rd Edition, 2005.
5. Introduction to Algorithms, Thomas H Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Third Edition, 2009, The MIT Press.
6. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, Pearson Education India. Fourth Edition, 2014,

VII. WEB REFERENCES:

1. http://www.tutorialspoint.com/data_structures_algorithms/data_structures_basics.htm
2. http://www.geeksforgeeks.org/b-tree-set-1-introduction-2/
3. http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html

VIII. E-TEXTBOOKS:

1. https://comsciers.files.wordpress.com/2015/12/horowitz- -of-computer-algorithms-2nd-edition.pdf