GUJARAT TECHNOLOGICAL UNIVERSITY

Advanced Data Structures
SUBJECT CODE: 3710215

Type of course: Core

Prerequisite: UG level course in Data Structures

Rationale:

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks				Total Marks
L	T	P	C	Theory Marks		Practical Marks		
				ESE(E)	PA (M)	PA (V)	PA (I)	
3	0	2	4	70	30	30	20	150

Content:

Sr. No	Content	Total Hrs	% Weightage
1	**Dictionaries:** Definition, Dictionary Abstract Data Type, Implementation of Dictionaries. **Hashing:** Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing	7	15
2	**Skip Lists:** Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Deterministic Skip Lists	5	10
3	**Trees:** Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees	9	19
4	**Text Processing:** Sting Operations, Brute-Force Pattern Matching, The Boyer- Moore Algorithm, The Knuth-Morris-Pratt Algorithm, Standard Tries, Compressed Tries, Suffix Tries, The Huffman Coding Algorithm, The Longest Common Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem.	12	25
5	**Computational Geometry:** One Dimensional Range Searching, Two Dimensional Range Searching, Constructing a Priority Search Tree, Searching a Priority Search Tree, Priority Range Trees, Quadtrees, k-D Trees.	10	21
6	Recent Trands in Hashing, Trees, and various computational geometry methods for effeciently solving the new evolving problem	5	10

Reference Books:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 2nd Edition, Pearson, 2004
2. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley, 2002.

Course Outcome:

After learning the course the students should be able to:

- Understand the implementation of symbol table using hashing techniques
- Develop and analyze algorithms for red-black trees, B-trees and Splay trees
- Develop algorithms for text processing applications
- Identify suitable data structures and develop algorithms for computational geometry problems

List of Experiments:

(Note: At least 12 Practicals should be performed from the list.)

1. Write a program which creates Binary Search Tree. And also implement recursive and non-recursive tree traversing methods inorder, preorder and post-order for the BST.
2. Write a program to implement any two hashing methods. Use any one of the hashing method to implement Insert, Delete and Search operations for Hash Table Management.
3. Explain Dictionary as an Abstract Data Type. Implement Dictionary using suitable Data Structure.
4. Write a program which creates AVLTree. Implement Insert and Delete Operations in AVL Tree. Note that each time the tree must be balanced.
5. Implement Red-Black Tree.
6. Implement 2-3 Tree.
7. Implement B Tree.
8. Implement a program for String Matching using Boyer-Moore Algorithm on a text file content.
9. Implement a program for String Matching using Knuth-Morris-Pratt Algorithm on a text file content.
10. Implement Huffman-Coding Method. Show the result with suitable example.
11. Implement Longest Common Subsequence(LCS) Problem using Dynamic Programming Method. Show the DP table and also find the particular solution of given strings.
12. Implement One Dimensional and Two Dimensional Range Searching in any language.
13. Write a program which creates Priority Search Tree. Implement Insert and Search Operations in this Tree.
14. Write a program which creates Skip Lists. Implement Insert, Search and Update Operations in Skip-Lists.
15. Design a simple search engine to display the possible websites upon entering a search query. Use suitable data structure for storage and retrieval.
16. Prepare a Report/Presentation on Recent trends on Hashing/Trees/Computational Geometry to solve ay of recent evolving problem in real world.