Syllabus Template FF No. :654

CS2041:: Advanced Data Structures

Credits: 4..... **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Data Structures, C/C++ Programming Language.

Course Objectives:

1. To introduce the fundamental design, analysis, and implementation of advanced data structures.
2. To emphasize concepts of principles for good program design, especially the uses of Trees, Graphs, Priority Queues and its applications.
3. To make understand the Significance of Strings Data Structures in the computer field.
4. To construct and implement various aspects of data structures.
5. To make understand about writing algorithms and step by step approach in solvingproblems with the help of fundamental data structures.
6. Toassociate data structures in developing and implementing efficient algorithms.

Course Relevance:

The Advanced Data structures is one of the essential branches of data science which is used for storage, organization and management of data and information for efficient, easy accessibility and modification of data. They are the basic element for creating efficient and effective software design and algorithms. The knowledge of creating and designing a good data structure is vital for becoming a commendable programmer. Its scope is also increasing with the increase in new methodologies of working in information technology.

Section 1: Topics/Contents

Unit-I:Trees 06 Hours

Basic terminology, representation using array and linked lists. Tree Traversals: Recursive and Non recursive, Operations on binary tree. Binary Search trees (BST), Threaded Binary Tree.

Unit-II:Graphs 06 Hours

Terminology and representation using Adjacency Matrix and Adjacency Lists, Graph Traversals and Application: BFS and DFS, connected graph, Bipartite Graph, Detecting Cycle in graph. Minimum Spanning tree: Prims and Kruskal's Algorithm, Shortest Path Algorithms, Union Find.

Unit-III:Hashing02 Hours

Hashing techniques, Hash table, Hash functions. Collision handling and Collision resolution techniques.

Section2:Topics/Contents

Unit-IV:Advanced Trees and Applications. 06 Hours
AVL Tree, Red-Black Trees,van Emde Boas tree, Fusion tree, Dynamic Finger Search Trees, B Trees and B+ Trees, Splay trees.

Unit-V: Priority Queues and Applications. 04 Hours
Amortized Analysis, Double Ended Priority queues, Leftist Trees, Binomial Heaps, Fibonacci Heaps, skew heaps, pairing heaps.

Unit-VI: Data Structures for Strings 04 Hours
String Searching: preliminaries, the DAWG, the position Heaps,tries and compressed tries, Suffix Treesand suffix arrays,Dictionaries Allowing Errors in Queries.

List of Tutorials (13):

1. Threaded Binary tree and Stack less Traversals using TBT.
2. Expression Trees.
3. BFS and DFS Traversals.
4.Prims and Kruskal's Algorithm
5. Dijkstra's Single source Shortest Path Algorithm.
6.Collision Handling with and Without Chaining with and Without Replacement.
7. AVL Tree and different operations on it.
8. Operations on Red Black tree.
9. Operations on B tree and B+ tree.
10. Operations on Splay trees.
11. Implementation of Priority Queue Applications.
12. Implementation of Binomial, Fibonacci Heap.
13. Operations on Suffix Trees and Suffix array.

List of Practical's (Minimum Six):

1. Assignment based on BST operations (Create, Insert, Delete and Traversals).
2. Assignment based on various operations on Binary Tree (Mirror image, Height, Leaf node display, Level wise display etc.).
3. Assignment based on TBT.
4. Assignment based on Graph traversal: DFS and BFS.
5. Assignment based on MST using Prim's and Kruskal's Algorithm.
6. Assignment based on Dijkstra's Algorithm.
7. Assignment based on Hashing.
8. Assignment based on AVL Tree.
9. Assignment based on operations on RED-Black trees and van Emde Boas trees.
10. Assignment based on B Trees and B+ Trees.

11. Assignment based on Priority Queues Application.
12. Assignment based on tries.
13. Assignment based on Suffix Trees.

List of Course Project areas:
1. Job Scheduling.
2. String processing.
3. Dictionary and Search engines.
4. Modeling the real world problems using graphs and trees,
5. Applications of B trees and B+ trees in Database management system.
6. GIS.
7. Image processing.
8. Internet routing.
9. Computational biology.
10. Computational geometry.
11. Data Mining.
12. Shortest Path Applications. (Kirchhoff's Circuit, TSP with Scenario.)
13. Design Gaming Applications.

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Assessment scheme covers following aspects of Modified Blooms Taxonomy:

L2 Understanding, L3 Apply, L3 Design, L3 Implement, L4 Analyze and L5 Evaluate

Laboratory Work: Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 10 Marks
Theory: End Semester Examination: 60 Marks converted to 20 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks
Programming Practical: End Semester Examination: 100 Marks converted to 40 Marks

Text Books: ***(As per IEEE format)***

1. Sartaj Sahni, Dinesh P. Mehta; Handbook of Data Structures and Applications; 2nd edition, Chapman & Hall/CRCGupta, R., Internet & Web Technologies, Engineering Handbook, 2019

2. *Fundamentals of Data Structures in C", E. Horwitz, S. Sahani, Anderson-Freed, Second Edition, Universities Press.*
3. *Narasimha karumanchi, "Data Structures and Algorithm Made Easy", Fifth Edition, CareerMonk publication.*

Reference Books: ***(As per IEEE format)***

1. *T. Cormen, R.Rivest, C. Stein, C. Leiserson, "Introduction to Algorithms", Second Edition, PHI publication.*
2. *Peter Brass, Advanced Data Structures, First Edition, Cambridge University Press.*

MOOCs Links and additional reading material:

1.www.nptelvideos.in
2. www.geeksforgeeks.org
3. www.udemy.com

Course Outcomes:

The student will be able to –

1. To demonstrate the use of binary tree traversals and to perform various operations on Non-linear data structures.
2. To analyze the Graph data structure and to implement the applications of Graph data structures.
3. To design the appropriate data structure by applying various hashing Techniques.
4. Model the real world problem with the help of appropriate tree data structure.
5. Analyze the amortized time complexity by applying suitable priority queue data structure.
6. Comprehend and select the storage pattern for strings processing application.

CO-PO Map:

	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CS2202.1	2															
CS2202.2		3														
CS2202.3			3													
CS2202.4				2												
CS2202.5								2			1					
CS2202.6				2									1			
Average	**2.0**	**3.0**	**3.0**	**2.0**				**2.0**			**1.0**		**1.0**			

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L3, CO2– L3, CO3 – L2, CO4 – L4, CO5 – L4 and CO6 – L5

Future Course Mapping:

Design and Analysis of Algorithms, Operating Systems, Compiler Design, Systems Programming, Data Science and similar courses.

Job Mapping:

Advanced Data Structures is must necessary part of any core programming job. Without Data structures it is not possible to be good in Competitive coding. All Industries always look for a strong knowledge in Advanced Data structures. Without learning this course, one can't imagine a job in computer/IT related industries and research.