CS1102: Data Structures

Course Title and Code Data Structures**:** CS1102	
Hours per Week	**L-T-P: 3-0-2**
Credits	**4**
Students who can take	B.Tech. Semester III (2020-2024) (CSE+EEE)

Course Objective: This course aims to develop understanding for Design, Analysis, and implementation of data structures and algorithms to solve computational problems using an object-oriented programming language. Topics includes introduction to algorithms and complexity analysis (time & space), Recursion, Linear Data Structures (Arrays, Queue, Stack, Linked list), Non-linear data structures (Trees, Graphs), Searching, Sorting, Indexing and Hashing.

Learning Outcome:

On successful completion of this course, the students should be able to:

CS1101.1. Write programs for performing basic operations like insertion, deletion, searching, sorting, merging, traversal etc. on various data structures like array, queue, stack, linked list, tree, graph.

CS1101.2. Use and design appropriate data structures for solving a variety of computational problem.

CS1101.3. Develop test cases for their programs and debug the code.

CS1101.4. Analyze the algorithms in terms of asymptotic time and space complexity.

CS1101.5. Implement and compare various searching and sorting algorithms

CS1101.6. Convert a recursive algorithm to non-recursive algorithm.

Prerequisites		**Programming Language**
Sr. No	**Specifications**	**Marks**
1	Attendance	Nil
2	Assignment	**20**
3	Class Participation	**10 (Hackerrank, code chef Medal Ranking Etc.)**
4	Quiz	**20 TCS ION LX**
5	Theory Exam-I	Nil
6	Theory Exam-II	Nil
7	Theory Exam-III	**25**
8	Report-I	Nil
9	Report-II	Nil
10	Report-III	Nil
11	Project-I	Nil
12	Project-II	Nil
13	Project-III	Nil
14	Lab Evaluation-I	**10 (Hacker Rank)**
15	Lab Evaluation-II	**15 (Hacker Rank)**
16	Course Portfolio	Nil
	Total (100)	**100**

Syllabus (Theory)

Unit I: Introduction to linear Data Structures: Types of Data Structures - Linear & Non-Linear Data Structures. Linear Structures: Arrays: Types, Operations and applications (searching sequential and binary, Sorting: bubble, Insertion, Selection, Quick and Merge sorting algorithms for different characteristics of input data. Complexity analysis, Comparison of sorting algorithms in term of complexity-time and space.

Unit II: Stacks and Queues: Operations and Applications, conversion of expression from one form to other form using stack (with & without parenthesis), Evaluation of expression in infix, postfix & prefix forms using stack, Queues: Operations and Applications, Circular Queues: Operations and Applications, De-queue and Priority queue, Recursion.

Unit III: Linear linked lists: Singly, doubly and circularly connected linear linked lists insertion, deletion at/ from beginning and any point in ordered or unordered lists, Application of linked list for polynomial operations, Comparison of arrays and linked lists as data structures. Implementation of stack, and queue, Algorithms for/of insertion, deletion of stack, and queue implemented using linked list data structure.

Unit IV: Trees: Trees definition, characteristics concept of child, sibling, parent child relationship etc., binary tree: different types of binary trees based on distribution of nodes, threaded binary tree and its application, insertion, deletion and traversal of binary trees, constructing binary tree from traversal results, BST tree: Concept of BST, insertion into and deletion from BST, Height balanced tree: AVL and its operations, Application of trees for representation of sets, Splay Tree and its operation.

Unit V: Graphs: Definition, Relation between tree & graph, directed and undirected graph, representation of graphs using adjacency matrix and list, Depth first and breadth first traversal of graphs, finding connected components and minimum spanning tree-Kruskal and Prims, Dijkstra Algorithm.

Indexing and Hashing: Hashing: The symbol table, Hashing Functions, Collision Resolution Techniques.

Syllabus (Lab):

All programs must be implemented and written in JAVA.

Data Structures Lab:

1. Write a program to search an element in the Array using Linear Search.
2. Write a program to implement Binary Search in an Array.
3. Write a program to insert an element in the given Array.
4. Write a program to delete an element in the given Array.
5. Write a program to merge two Arrays into single Array.
6. Write a program to merge two sorted Arrays into one sorted Array.
7. Write a program to search an element in the Array using Iterative and Recursive Binary Search.
8. Write a menu driven program to implement QUEUE using Arrays that performs following operations
   (a) INSERT (b) DELETE (c) TRAVERSAL (d) PEEP (e) ISFULL (f) ISEMPTY
9. Write a menu driven program to implement Circular Queue using Arrays that performs following operations.
   (a) INSERT (b) DELETE (c) DISPLAY (d) PEEP (e) ISFULL (f) ISEMPTY
   Write a menu driven program to implement a program for Stack that performs following operations using Array.
   (a)PUSH (b) POP (c) PEEP (d) DISPLAY (e) ISFULL (f) ISEMPTY
10. Write a program to convert infix notation to postfix notation using Stack.
11. Write a program to convert infix notation to prefix notation using Stack.
12. Write a program to evaluate given postfix notation using Stack.
13. Write a menu driven program to implement following operations on the singly Linked List.
    a) Insert a node at the front of the Linked List.
    b) Insert a node at the end of the Linked List.
    c) Insert a node such that Linked List is in ascending order. (according to info. Field)
    d) Delete a first node of the Linked List.
    e) Delete a node before specified position.
    f) Delete a node after specified position.
    g) Traversal of Linked List
14. Write a menu driven program to implement Stack using Linked List.
15. Write a menu driven program to implement Queue using Linked List.
16. Write a program to implement following operations on the doubly Linked List.
    a) Insert a node at the front of the Linked List.
    b) Insert a node at the end of the Linked List.
    c) Delete a last node of the Linked List.
    d) Delete a node before specified position.
    e) Traversal of Linked List
17. Write a program to implement following operations on the circular Linked List.
    a) Insert a node at the end of the Linked List.
    b) Insert a node before specified position.

c) Delete a first node of the Linked List.
d) Delete a node after specified position.
e) Traversal of Linked List

18. Write a program which create Binary Tree.
19. Write a program to implement recursive and non-recursive Binary Tree traversing methods in-order, pre-order and post-order traversal.
20. Write a menu driven program to implement Binary Search Tree and its Traversal.
21. Write a menu driven program to implement AVL Tree and its Traversal.
22. Write a program to implement Breadth First Search in a given Graph.
23. Write a program to implement Depth First Search in a given Graph.
24. Write a program to check whether the given Graph is cyclic or not.
25. Write a program to implement Kruskal's Algorithm for the given Graph.
26. Write a program to implement Prim's Algorithm for the given Graph.
27. Write a program to implement Dijkstra's Algorithm for the given Graph.
28. Write a program to implement Bubble Sort, Selection sort, Insertion Sort in an Array.
29. Write a program to implement Merge Sort in an Array.
30. Write a program to implement Quick Sort in an Array.

Text Books:

T1. Sahni, Sartaj. Data structures, algorithms, and applications in Java. Universities Press, 2005.

T2. Goodrich, Michael T., Roberto Tamassia, and Michael H. Goldwasser. Data structures and algorithms in Java. John Wiley & Sons, 2014.
T3. Data Structures and Algorithms in Java -- Robert Lafore second edition Sams Publication, 2003

Reference Books:

R1. Introduction to Algorithms, by Cormen, Leiserson, Rivest, and Stein.
R2. Alfred V. Aho, Jeffrey D. Ullman, John E. Hop croft, Data Structures and Algorithms. Pearson Education, 2012

Recommended MooC :
Data Structure and Algorithms NPTEL

https://nptel.ac.in/courses/106/102/106102064/
https://nptel.ac.in/courses/106/106/106106127/

Coursera
https://www.coursera.org/specializations/data-structures-algorithms

GeekforGeeks
https://www.geeksforgeeks.org/data-structures/