FR. Conceicao Rodrigues College Of Engineering

Father Agnel Ashram, Bandstand, Bandra-west, Mumbai-50

S.E. (ECS) (semester III) (2020-2021)

Lesson Plan

Subject: Data Structures and Analysis (ECC304)

Credits-4

SYLLABUS

Sr. No.	Module	Detailed Content	CO Mapping
00	Prerequisite	C Programming Language	
01	Introduction to Data structures and Analysis	Introduction to Data Structures, Types of Data Structures Linear and Nonlinear, Operations on Data Structures, Concept of array, Static arrays vs Dynamic Arrays, structures. Introduction to Analysis of Algorithms, characteristics of algorithms, Time and Space complexities, Asymptotic notations.	CO1 CO2 CO4 CO5
02	Stack and Queues	Introduction, Basic Stack Operations, Representation of a Stack using Array, Applications of Stack Well form-ness of Parenthesis, Infix to Postfix Conversion and Postfix Evaluation. Queue, Operations on Queue, queue-Round Robin Algorithm	CO1 CO4 CO5
03	Linked list	Introduction, Representation of Linked List, Linked List v/s Array, Types of Linked List - Singly Linked List (SLL), Operations on Singly Linked List: Insertion, Deletion, reversal of SLL, Print SLL. Implementation of Stack and Queue using Singly Linked List. Introduction to Do Representation of a Queue using array, Circular Queue, concept of priority Queue, Applications of Qubly Linked List and Circular Linked List	CO1 CO4 CO5
04	Trees	Introduction, Tree Terminologies, Binary Tree, Types of Binary Tree, Representation of Binary Trees, Binary Tree Traversals, Binary Search Tree Operations on Binary Search Tree, Applications of Binary Tree Expression Tree, Huffman Encoding.	CO2 CO4 CO5

05	Graphs	Introduction, Graph Terminologies, Representation of graph (Adjacency matrix and adjacency list), Graph Traversals Depth First Search (DFS) and Breadth First Search (BFS), Application Topological Sorting.	CO2 CO4 CO5
06	Introduction to Sorting and Searching	Introduction to Searching: Linear search, Binary search, Sorting: Internal VS. External Sorting, Sorting Techniques: Bubble, Insertion, selection, Quick Sort, Merge Sort, Comparison of sorting Techniques based on their complexity. Hashing Techniques, Different Hash functions, Collision & Collision resolution techniques: Linear and Quadratic probing, Double hashing	CO3 CO4 CO5

Internal Assessment (IA):

Two tests must be conducted which should cover at least 80% of syllabus. The average marks of both the tests will be considered as the final IA marks.

CO-Statements:

CO1. Implement various linear data structures.

CO2. Implement various non linear data structures.

CO3. Select appropriate sorting and searching techniques for a given problem and use it.

CO4. Develop solutions for real world problems by selecting appropriate data structure and algorithms.

CO5. Analyse the complexity of the given algorithms.

CO-PO-PSO Mapping

Course Name	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2
CO1		2	2										1	
CO2		2	2										1	
CO3	1	2											1	
CO4		2	2										1	
CO5		2		1										

CO Assessment Tools

	Direct Methods							**Indirect Methods**
	Test1	**Assig1**	**Lab Work**	**Test2**	**Assig2**	**University Theory Result**	**University Theory Result**	Course Exit Survey
CO1	**20%**	**20%**	**10%**			**25%**	25%	100%
CO2			25%	25%		**25%**	25%	100%
CO3				25%	**25%**	**25%**	25%	100%
CO4	**25%**			25%		**25%**	25%	100%
CO5				25%	**25%**	**25%**	25%	100%

Curriculum Gap/

Content beyond syllabus

Lecture Plan:

No of classes available:	55	No of Classes taken:	56	
Sr. No.	Topic Planned	Planned Date	Actual Date	Delivery Mechanisms
1	Revision of arrays & introduction to data structures	10/07/2020	Friday 10/07/2020	Online mode using ppt
2	Introduction to Data Structures	13/07/2020	Monday 13/07/2020	Online mode using ppt
3	Implementation of stack	16/07/2020	Thursday 16/07/2020	Online mode using ppt and live coding
4	Applications of stack-1. Reversing string, 2. Paranthesis checking of expression	17/07/2020	Friday 17/07/2020	Online mode using ppt
5	Applications of stack continued... 2. Paranthesis checking of expression	20/07/2020	Monday 20/07/2020	Online mode using ppt
6	algorithm: Infix to postfix conversion using stack	23/07/2020	Thursday 23/07/2020	Online mode using ppt
7	Algorithm: Infix to postfix conversion using stack	24/07/2020	Friday 24/07/2020	Online mode using ppt

8	Program for infix to postfix	27/07/2020	Monday 27/07/2020	Online mode using ppt and live coding
9	Program to evaluate postfix expression	30/07/2020	Thursday 30/07/2020	Online mode using ppt and live coding
10	Queue using array	31/07/2020	Friday 31/07/2020	Online mode using ppt and live coding
11	Queue using array-Linear Queue and strated Circular queue	03/08/2020	Monday 03/08/2020	Online mode using ppt and live coding
12	Program for Circular queue using array	06/08/2020	Thursday 06/08/2020	Online mode using ppt and live coding
13	Priority Queue using array-method1	07/08/2020	Friday 07/08/2020	Online mode using ppt and live coding
14	Priority queue using array-method2, Linked List introduction, Array Vs Linked list	10/08/2020	Monday 10/08/2020	Online mode using ppt and live coding
15	Linked List: started coding for linked list	13/08/2020	Thursday 13/08/2020	Online mode using ppt and live coding
16	Linked List: create_linked_list, insert_front, insert_end functions	17/08/2020	Monday 17/08/2020	Online mode using ppt and live coding
17	Linked List: insert_front, insert_end functions, insert_at_pos	20/08/2020	Tuesday 18/08/2020	Online mode using ppt and live coding
18	Linked_List: remove_front, remove_end	21/08/2020	Wed 19/08/2020	Online mode using ppt and live coding
19	Linked List: remove front, remove_end, remove_pos	24/08/2020	Friday 21/08/2020	Online mode using ppt and live coding
20	Split and Traverse function of linked list	27/08/2020	Monday 24/08/2020	Online mode using ppt and live coding
23	Concatenate and Print lternate nodes in linked list	28/08/2020	Thursday 27/08/2020	Online mode using ppt and live coding
24	Codes of Split, Concatenate and print alternate nodes	31/08/2020	Monday 31/08/2020	Online mode using ppt and live coding
26	Copy function of linked list	03/09/2020	Thu 03/09/2020	Online mode using ppt and live coding

27	Reverse function of linked list	04/09/2020	Friday 04/09/2020	Online mode using ppt and live coding
28	Stack using linked list.	07/09/2020	Monday 07/09/2020	Online mode using ppt and live coding
29	Stack using linked list.	10/09/2020	Thursday 10/09/2020	Online mode using ppt and live coding
30	Revision of previous topics	11/09/2020	Friday 11/09/2020	Online mode using ppt and live coding
31	Queue using linked list and circular linked list	21/09/2020	Thursday 24/09/2020	Online mode using ppt and live coding
32	Started with Doubly Linked llst	24/09/2020	Friday 25/09/2020	Online mode using ppt and live coding
33	Operations on doubly linked list	25/09/2020	Monday 28/09/2020	Online mode using ppt and live coding
34	Applications of Linked List and Started with Tree Data Structure	28/09/2020	Thursday 01/10/2020	Online mode using ppt and live coding
35	Binary Tree traversal techniques	01/10/2020	Monday 05/10/2020	Online mode using ppt and live coding
36	Examples to solve on Binary tree traversal	5/10/2020	Thursday 08/10/2020	Online mode using ppt
37	Binary sreach tree	8/10/2020	Friday 09/10/2020	Online mode using ppt
38	Insert operation of BST	9/10/2020	Wednesday 14/10/2020	Online mode using ppt and live coding
39	Search and Delete operation on BST	12/10/2020	Thursday 15/10/2020	Online mode using ppt and live coding
40	Delete operation of BST	15/10/2020	Friday 16/10/2020	Online mode using ppt and live coding
41	Delete operation of BST, find_min, find _max functions and Applications of BST	16/10/2020	Monday 19/10/2020	Online mode using ppt and live coding
42	Application of Binary tree, Huffman encoding	19/10/2020	Thursday 22/10/2020	Online mode using ppt

43	Started with Graph data structure	22/10/2020	Friday 23/10/2020	Online mode using ppt
44	Adjecency Matrix, Adjecancy List and Graph Traversal algorithms	23/10/2020	Monday 26/10/2020	Online mode using ppt
45	Graph Traversal: DFS algorithm and implementation	26/10/2020	Thursday 29/10/2020	Online mode using ppt
46	DFS implementation completed, BFS algorithm	29/10/2020	Monday 02/11/2020	Online mode using ppt
47	BFS implementation, topological sort	02/11/2020	Thursday 05/11/2020	Online mode using ppt
48	Analysis of algorithms	05/11/2020	Friday 06/11/2020	Online mode using ppt
49	Asymptotic Notations	06/11/2020	Monday 09/11/2020	Online mode using ppt
50	Bubble Sort, Selection	09/11/2020	Tuesday 10/11/2020	Online mode using ppt
51	Insertion Sort	12/11/2020	Wednesday 11/11/2020	Online mode using ppt
52	Merge Sort	13/11/2020	Thursday 12/11/2020	Online mode using ppt
53	Linear Search and Binary Search	19/11/2020	Thursday 19/11/2020	Online mode using ppt
54	Quick Sort	20/11/2020	Friday 20/11/2020	Online mode using ppt
55	Quick sort implementation and started with Hashing	23/11/2020	Friday 20/11/2020	Online mode using ppt
56	Hashing in Data structures and UT2 QP format		Monday 23/11/2020	Online mode using ppt

Lab Plan: Data Structures Lab (ECC303)
Lab Outcomes: Students will be able to:

1. Students will be able to implement linear data structures & will be able to handle operations like insertion, deletion, searching and traversing on them.

2. Students will be able to implement nonlinear data structures & will be able to handle operations like insertion, deletion, searching and traversing on them.

3. Students will be able to choose appropriate data structure and apply it in various problem domains.

4. Students will be able to select appropriate searching techniques for given problems.

Lab Plan: Data Structures Lab

Sr. No.	Topic	Week No.	Lab Outcome
1.	Implement Stack ADT using array	Week 1	LO1
2.	Convert an Infix expression to Postfix expression using stack ADT	Week 1	LO1, LO3
3.	Evaluate Postfix Expression using Stack ADT	Week 2	LO1, LO3
4.	Implement Linear Queue ADT using array	Week 2	LO1, LO3
5.	Implement Circular Queue ADT using array	Week 3	LO1, LO3
6.	Implement Singly Linked List ADT	Week 3	LO1
7.	Implement Stack ADT using Linked List	Week 4	LO1, LO3
8.	Implement Linear Queue ADT using Linked List	Week 4	LO1, LO3
9.	Implement Binary Search Tree ADT using Linked List	Week 5	LO2, LO3
10.	Implement Depth First Search and Breadth First Search Graph Traversal technique	Week 5	LO2, LO3
11.	Implement searching algorithms -Linear search, Binary search	Week 6	LO4
12.	Implement sorting algorithms (any 2)- bubble, selection, insertion, merge, quick	Week 6	LO4

Assignment Plan:

Assig No.	**Date**	**Questions**	**CO/LO**
1	**03/09/2020**	1) Select and write appropriate data structure as applicable to following specified problem definitions. 1. To record the sequence of all the pages browsed in one session. 2. You have to store social network “feeds”. You do not know the size, and things may need to be dynamically added. 3. Process scheduling in kernel	**CO1, CO4**

		4. You need to store undo/redo operations in a word processor. 5. parser parsing an expression (i.e., parse). 6. To implement printer spooler so that jobs can be printed in the order of their arrival. 7. To implement back functionality in the internet browser 8. To store a set of fixed key words which are referenced very frequently. 9. To store the customer order information in a drive-in burger place. (Customers keep on coming and they have to get their correct food at the payment/food collection window.) 10. To store a set of programs which are to be given access to a hard disk according to their priority. 2) Convert ((5+4)*8/(3*4)) into postfix using stack. Show status of stack at every step. (Draw pictures of different stack to show the status of stack). 3) Use the postfix expression in the question no. 2 and evaluate the same using stack. Show status of stack diagrammatically at every step. 4) Explain the application of stack to check the validity of parenthesis in the expression with one example. Also write the pseudo code for the same. 5) Given a queue Q containing n elements, transfer these items on to a stack S (initially empty) such that front element of Q appears at the top of the stack. 6) Write functions to Print alternate nodes and remove duplicates in the linked list.	
2	**19/11/2020**	Q1: Trace the quick sort algorithm to sort the array [11, 25,9,3,5,0,20,28,7] into ascending order. Use the array implementation exactly as described in the class. Also list the calls to quicksort and partition in the order they occur. Assume that the last element is chosen as pivot. Q2: Algorithm Analysis Compare all sorting algorithms on their running time complexities for best, average and worst case in tabular form. Q3: Determine the running time analysis on quick sort, merge sort and insertion sort a. Do the runs on an already sorted input array. Order the algorithms according to how their execution time grows with n.	**CO3, CO5**

		b. Do all the runs on an input array sorted in reverse order. Order the algorithms according to how their execution time grows with n. c. Do the runs on an input array where all elements have the same value. Order the algorithms on how their execution time grows with n. d. Which algorithms have a running-time behaviour only dependent on the input size n, (i.e. which algorithms are not sensitive for the distribution of input elements?) Q4: Define asymptotic notations along with examples and List out the properties of asymptotic notations. Q5: Draw a hash table with open addressing and a size of 9. Use the hash function "k%9". Insert the keys: 5, 29, 20, 0, 27 and 18 into your table (in that order)	

Term Work:

At least 10 experiments and 2 assignments covering entire syllabus of Data Structures and Algorithms (ECC 304) should be set to have well predefined inference and conclusion. The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Practical and Oral exam will be based on the entire syllabus.

Total 25 Marks = (Experiments-15 mark + Attendance -5 mark + Assignments-05 mark)

Oral & Practical Exam: An Oral & Practical exam will be held based on the above syllabus.