SIR C.R.REDDY COLLEGE OF ENGINEERING, ELURU

DEPARTMENT OF INFORMATION TECHNOLOGY

LESSON PLAN

SUBJECT: **DATA STRUCTURES THROUGH C++ (R1621055)**

CLASS: II/IV B.Tech., 1ST SEMESTER, A.Y.2019-20

INSTRUCTOR: **SK.Meeravali**

SIR C R REDDY COLLEGE OF ENGINEERING
DEPT. OF INFORMATION TECHNOLOGY

Programme	:	B. Tech
Year	:	Second
Semester	:	First Semester
Academic Year	:	2019-20
Course	:	DATA STRUCTURES THROUGH C++ (R1621055).
Instructor	:	SK.Meeravali

COURSE CONTENTS

Coverage Of Course	Course Title	Course Title	Credits- 3c	Theory Paper
R1621055	DATA STRUCTURES THROUGH C++	R1621055	L-4	Max. Marks: 70 Duration: 3hrs

Course objectives:

1. To be familiar with basic techniques of object oriented principles and exception handling using C++.
2. To be familiar with the concepts like Inheritance, Polymorphism.
3. Solve problems using data structures such as linear lists, stacks, queues, hash tables.
4. Be familiar with advanced data structures such as balanced search trees, AVL Trees, and B-Trees.

Course Outcomes:

1. Understand the basic data structures for storage and retrieval of ordered or unordered data.
2. Understand basic data structures such as arrays, linked lists, stacks and queues.
3. Ability to implement algorithms for the creation, insertion, and deletion of each data structure.
4. Analyze and compare algorithms for efficiency using Big-O notation.
5. Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data.

ONLINE REFERENCES:

https://www.includehelp.com/data-structure-tutorial/
https://www.geeksforgeeks.org/data-structures/
https://www.studytonight.com/data-structures/
http://btechsmartclass.com/data_structures/introduction-to-algorithms.html
https://www.tutorialspoint.com/data_structures_algorithms/index.htm

Prerequisite: Organizing the data in computer system, basics about algorithms, flowcharts and C++ language.

Internal Assessment Details:

Internal Test 1& 2	:	15 Marks
Quiz	:	10 Marks
Assignment	:	5 Marks

Total: 30 Marks

II Year - I Semester	**L**	**T**	**P**	**C**
	4	**0**	**0**	**3**

DATA STRUCTURES THROUGH C++--R1621055

OBJECTIVES:

- To be familiar with basic techniques of object oriented principles and exception handling using C++
- To be familiar with the concepts like Inheritance, Polymorphism
- Solve problems using data structures such as linear lists, stacks, queues, hash tables
- Be familiar with advanced data structures such as balanced search trees, AVLTrees, and B Trees.

UNIT-I: ARRAYS

Abstract Data Types and the C++ Class, An Introduction to C++ Class- Data Abstraction and Encapsulation in C++- Declaring Class Objects and Invoking Member Functions- Special Class Operations- Miscellaneous Topics- ADTs and C++Classes, The Array as an Abstract Data Type, The Polynomial Abstract Data type- Polynomial Representation- Polynomial Addition. Spares Matrices, Introduction- Sparse Matrix Representation- Transposing a Matrix- Matrix Multiplication, Representation of Arrays.

UNIT-II: STACKS AND QUEUES

Templates in C++, Template Functions- Using Templates to Represent Container Classes, The Stack Abstract Data Type, The Queue Abstract Data Type, Sub typing and Inheritance in C++, Evaluation of Expressions, Expression- Postfix Notation- Infix to Postfix.

UNIT-III: LINKED LISTS

Single Linked List and Chains, Representing Chains in C++, Defining a Node in C++- Designing a Chain Class in C++- Pointer manipulation in C++- Chain Manipulation Operations, The Template Class Chain, Implementing Chains with Templates- Chain Iterators- Chain Operations- Reusing a Class, Circular Lists, Available Space Lists, Linked Stacks and Queues, Polynomials, Polynomial Representation- Adding Polynomials- Circular List Representation of Polynomials, Equivalence Classes, Sparse Matrices, Sparse Matrix Representation- Sparse Matrix Input- Deleting a Sparse Matrix, Doubly Linked Lists, Generalized Lists, Representation of Generalized Lists- Recursive Algorithms for Lists- Reference Counts, Shared and Recursive Lists

UNIT-IV: TREES

Introduction, Terminology, Representation of Trees, Binary Trees, The Abstract Data Type, Properties of Binary Tress, Binary Tree Representations, Binary Tree Traversal and Tree Iterators, Introduction, Inorder Traversal Preorder Traversal, Postorder Traversal, Thread Binary Trees, Threads, Inorder Traversal of a Threaded Binary Tree, Inserting a

Node into a Threaded Binary Tree, Heaps, Priority Queues, Definition of a Max Heap, Insertion into a Max Heap, Deletion from a Max Heap, Binary Search Trees, Definition, Searching a Binary Search Tree, Insertion into a Binary Search Tree, Deletion from a Binary Search Tree, Height of Binary Search Tree.

UNIT-V: GRAPHS

The Graph Abstract Data Type, Introduction, Definition, Graph Representation, Elementary Graph Operation, Depth First Search, Breadth First Search, Connected Components, Spanning Trees, Biconnected Components, Minimum Cost Spanning Trees, Kruskal S Algorithm, Prim s Algorithm Sollin' s Algorithm, Shortest Paths and Transitive Closure, Single Source/All Destination: Nonnegative Edge Cost, Single Source/All Destination: General Weights, All-Pairs Shortest Path, Transitive Closure.

UNIT-VI: SORTING

Insertion Sort, Quick Sort, Merge Sort Merging, Iterative Merge Sort, Recursive Merge Sort, Heap Sort.

OUTCOMES:

- Distinguish between procedures and object oriented programming.
- Apply advanced data structure strategies for exploring complex data structures.
- Compare and contrast various data structures and design techniques in the area of Performance.
- Implement data structure algorithms through C++.
- Incorporate data structures into the applications such as binary search trees, AVL and B Trees
- Implement all data structures like stacks, queues, trees, lists and graphs and compare their Performance and trade offs

TEXT BOOKS:

1. Data structures, Algorithms and Applications in C++, S.Sahni, University Press (India) Pvt.Ltd, 2nd edition, Universities Press, Pvt. Ltd.
2. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
3. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and .Mount, Wiley student edition, John Wiley and Sons.

REFERENCE BOOKS:

1. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Thomson
2. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
3. Problem solving with C++, The OOP, Fourth edition, W. Savitch, Pearson education.

SIR C R REDDY COLLEGE OF ENGINEERING

DEPT. OF INFORMATION TECHNOLOGY

COURSE SCHEDULE

The schedule for the whole course/subject is:

Unit No	Description Of The Chapter	Description of the Topics	Total no.of Periods (L+T)
1.	**Introduction To C++ And Data Structures**	Abstract Data Types and the C++ Class, Data Abstraction and Encapsulation in C++- Declaring Class Objects and Invoking Member Functions, The Array as an Abstract Data Type, The Polynomial Abstract Data type- Polynomial Representation- Polynomial Addition. Spares Matrices, Introduction- Sparse Matrix Representation- Transposing a Matrix- Matrix Multiplication, Representation of Arrays.	**5**
2.	**Stacks And Queues**	Templates in C++, Using Templates to Represent Container Classes, The Stack Abstract Data Type, The Queue Abstract Data Type, Sub typing and Inheritance in C++, Evaluation of Expressions, Expression- Postfix Notation- Infix to Postfix.	**10**
3.	**Linked Lists**	Single Linked List and Chains, Representing Chains in C++, Chain Manipulation Operations, Chain Operations- Reusing a Class, Circular Lists, Available Space Lists, Linked Stacks and Queues, Polynomials, Adding Polynomials- Circular List Representation of Polynomials, Equivalence Classes, Sparse Matrices, Sparse Matrix Representation- Sparse Matrix Input- Deleting a Sparse Matrix, Doubly Linked Lists, Generalized Lists, Representation of Generalized Lists- Recursive Algorithms for Lists- Reference Counts, Shared and Recursive Lists.	**15**
4.	**Trees**	Terminology, Representation of Trees, Binary Trees, Properties of Binary Tress, Binary Tree Representations, Binary Tree Traversal and Tree Iterators, Inorder Traversal Preorder Traversal, Postorder Traversal, Thread Binary Trees, Threads, Inorder Traversal of a Threaded Binary Tree, Inserting a Node into a Threaded Binary Tree, Heaps, Priority Queues, Definition of a Max Heap, Insertion into a Max Heap, Deletion from a Max Heap, Binary Search Trees, Definition, Searching a	**15**

		Binary Search Tree, Insertion into a Binary Search Tree, Deletion from a Binary Search Tree, Height of Binary Search Tree.	
5.	**Graphs**	The Graph Abstract Data Type, Definition, Graph Representation, Elementary Graph Operation, Depth First Search, Breadth First Search, Connected Components, Spanning Trees, Biconnected Components, Minimum Cost Spanning Trees, Kruskal S Algorithm, Prim s Algorithm Sollin' s Algorithm, Shortest Paths and Transitive Closure, Single Source/All Destination: Nonnegative Edge Cost, Single Source/All Destination: General Weights, All-Pairs Shortest Path, Transitive Closure.	**15**
6.	**Sorting**	Insertion Sort, Quick Sort, Merge Sort Merging, Iterative Merge Sort, Recursive Merge Sort, Heap Sort.	**7**

Total no.of instructional periods available for the course : 67 periods

Total no.of estimated periods : 64 periods

Signature of the H.O.D

Signature of the Faculty

Date:

LECTUREPLAN

DEPARTMENT	INFORMATION TECHNOLOGTY
NAME OF LECTURER	D. SRINIVASA RAO

S. No	Topics To Be Covered	No. Of Lecture Hours	Teaching Method	Outcomes
1.	Data Structures Introduction and review of C++	1	BB	
2.	Introduction to abstract data types	1	BB	
3.	Data Abstraction and Encapsulation in C++	1	BB	
4.	The Array as an Abstract Data Type, The Polynomial Abstract Data type- Polynomial Representation- Polynomial Addition	1	BB	
5.	Introduction- Sparse Matrix Representation	1	BB	
6.	Transposing a Matrix- Matrix Multiplication, Representation of Arrays.	1	BB	
7.	Templates in C++, Using Templates to Represent Container Classes	1	BB	
8.	The Stack Abstract Data Type	1	BB	
9.	The Queue Abstract Data Type	1	BB	
10.	Sub typing and Inheritance in C++	1	BB	
11.	Evaluation of Expressions,	1	BB	
12.	Expression- Postfix Notation	1	BB	
13	Infix to Postfix.	1	BB	
14.	Examples on postfix expression evaluation	1	BB	
15.	Examples on infix to postfix expression evaluation	1	BB	
16.	Single Linked List and Chains, Representing Chains	1	BB	
17.	Representing Chains	1	BB	
18.	Chain Manipulation Operations	1	BB	
19.	Chain Operations- Reusing a Class	1	BB	
20.	Circular Lists	1	BB, LCD	

21.	Linked Stacks and Queues	1	BB	
22.	Circular List Representation of Polynomials	1	BB	
23.	Equivalence Classes	1	BB	
24.	Sparse Matrices, Sparse Matrix Representation	1	BB	
25.	Sparse Matrix Input- Deleting a Sparse Matrix	1	BB	
26.	Doubly Linked Lists	1	BB, LCD	
27.	Generalized Lists, Representation of Generalized Lists	1	BB, LCD	
28.	Recursive Algorithms for Lists- Reference Counts	1	BB	
29.	Shared and Recursive Lists.	1	BB	
30.	Terminology, Representation of Trees	1	BB	
31.	Binary Trees, Properties of Binary Tress	1	BB	
32.	Binary Tree Representations, Binary Tree Traversal	1	BB	
33.	Inorder Traversal Preorder Traversal with example	1	BB	
34.	Postorder Traversal with example	1	BB	
35.	Thread Binary Trees, Threads,	1	BB, LCD	
36.	Inorder Traversal of a Threaded Binary Tree	1	BB, LCD	
37.	Inserting a Node into a Threaded Binary Tree	1	BB	
38.	Examples on binary tree and threaded binary tree.	1	BB	
39.	Heaps and Priority Queues	1	BB	
40.	Definition of a Max Heap	1	BB	
41.	Insertion into a Max Heap and Deletion from a Max Heap,	1	BB	
42.	Binary Search Trees, Definition and Searching a Binary Search Tree,	1	BB	
43.	Insertion into a Binary Search Tree, Deletion from a Binary Search Tree	1	BB	
44	Height of Binary Search Tree.	1	BB	
45.	Graph ADT and Graph Representation	1	BB	
46.	Elementary Graph Operation, Depth First Search	1	BB	
47.	Breadth First Search	1	BB	

48	Connected Components	1	BB	
49.	Spanning Trees	1	BB	
50.	Biconnected Components, Minimum Cost Spanning Trees	1	BB	
51.	Biconnected Components, Minimum Cost Spanning Trees	1	BB	
52.	Kruskal S Algorithm	1	BB	
53.	Prim s Algorithm	1	BB	
54.	Sollin' s Algorithm	1	BB	
55.	Shortest Paths and Transitive Closure	1	BB	
56.	Single Source/All Destination: Nonnegative Edge Cost	1	BB	
57.	Single Source/All Destination: General Weights	1	BB	
58.	All-Pairs Shortest Path, Transitive Closure.	1	BB	
59.	Insertion Sort	1	BB	
60.	Quick Sort	1	BB	
61	Merge Sort Merging	1	BB	
62	Iterative Merge Sort	1	BB	
63	Recursive Merge Sort	1	BB	
64	Heap Sort	1	BB	
TOTAL			**64**	

UNIT WISE QUESTIONS
(Short and essay)

Unit -1

Short answers

1. Differentiate between Information and data?
2. What is a data structure and what are types?
3. Explain how 3D arrays are represented?
4. What is a pointer?
5. What is ADT?
6. Applications of Data Structures?

Essay Questions

1. Write a short note on row major order column major order implementation?
2. Explain in detail all the types of data structures with examples?
3. Describe representation of numbers and characters in memory?
4. Explain the 3D array with an example?

UNIT- 2

Short Questions

1. What is a stack?
2. What are the primitive operations?
3. What are overflow and underflow errors?
4. What are applications of stacks?
5. What are the types of expressions?
6. What are the conditions for ADT?

Essay Questions

1. Explain the stack as abstract data type and implement the operations?
2. Write an algorithm for postfix evaluation?
3. Write an algorithm for infix to postfix conversion?
4. Write an algorithm for matching of nested parenthesis?

5. Write an algorithm for infix to prefix conversion?

Unit- 3

Short Questions

1. What is recursion?
2. What are the types in recursion?
3. Write code for recursive version of binary search?
4. Write about simulation and efficiency of recursion?

Essay Questions

1. Write an algorithm for Binary search and explain it with an example ?
2. Explain about the Towers Hanoi problem with example?
3. Explain about the simulation of factorial in detail?
4. What are the difference of iteration

Unit- 4

Short Questions

1. What are Queues? What are the types?
2. What are limitations of a queue?
3. What are limitations of a circular queue?
4. What are the applications of dequeue?
5. What are the applications of priority queue?
6. What is a linked list ? What are the types in it?
7. What are the advantages of LL over an array?
8. Write any three applications of CDLL?
9. What are the main differences of SLL and DLL?

Essay Questions

1. Explain the Queue as abstract data type and implement the operations of queue?
2. Explain the circular queue operations in detail?
3. Explain the dequeue operations in detail?
4. Write about the types of priority queue and implement the operations of it?
5. How stacks are implemented using linked list and how it differs from an array?

6. How Queues are implemented using linked list and how it differs from an array?

Unit -5

Short Questions

1. What is a tree? What are its applications?
2. What is a binary tree?
3. What is a binary search tree?
4. What are difference between complete binary tree almost complete binary tree?
5. What is the threaded binary tree and its applications?
6. What tree traversing methods?
7. How trees are represented?

Essay Questions

1. Explain Binary search tree construction process in detail?
2. Write about the threaded binary tree traversing method in detail?
3. Explain about tree traversing methods in detail with example?
4. Write an algorithm to insert an element and delete an element in a binary search tree?
5. Write an algorithm to search an element in a binary search tree?

Unit -6

Short Questions

1. What is Big O notation?
2. What is meant by efficiency of sorting?
3. What is best, average, and worst case analysis?
4. What is the efficiency of heap sort for all cases?

Essay Questions

1. Explain the quick sort with an example?
2. Explain the heap sort with an example?
3. Explain the radix sort with an example?
4. Explain the selection sort with an example?
5. Explain the Bubble sort with an example?

6. Explain the Binary tree sort with an example?
7. Explain the Merge sort with an example?
8. Explain the Address calculation sort with an example?
9. Explain the shell sort with an example?
10. Explain the Insertion sort with an example?

Unit -7

Short Questions

1. What is meant by searching?
2. What is the efficiency of different searching techniques?
3. What is meant by dictionary?
4. Explain the dictionary as an ADT?

Essay Questions

1. Explain linear searching technique with an example?
2. Explain binary searching technique with an example?
3. Explain Interpolation searching technique with an example?

Unit -8

Short Questions

1. What is a Graph?
2. What are the applications of the graph?
3. What is BFS and DFS?
4. What is meant by minimum spanning tree?
5. What is meant by transitive closure?

Essay Questions

1. Explain the graph traversing techniques with an example?
2. Explain the prims algorithm with an example?
3. Explain the Kruskals algorithm with an example?
4. Explain dijkstra's algorithm with an example?
5. Explain the single source shortest path algorithm with an example?