CSC-222: Data Structures (Theory)

General Information

Course Number	**CSC-222**
Credit Hours	3+1 (Theory Credit Hour = 3, Lab Credit Hours = 1)
Prerequisite	Programming Fundamentals
Course Coordinator	Not Specified

Course Objectives

The objective of this course is to make students familiar with the concepts of the way data is stored inside computer and its manipulation using different algorithms. Students will learn different data structures such as array, stack, queue, link list, trees, graphs, sorting algorithm etc. Since Programming fundamentals is the pre-requisite of this course, therefore, this course would be using c++ language to implement all the data structures.

Catalog Description

CSC-222

Course Content

Week No	*1. Topics*	Activities / Assignments

1	• Introduction to the course
	• Elementary data structures
	• Arrays
	  ○ Review of single-dimension arrays
	  ○ Concept and implementation of 2D arrays
	  ▪ Manipulating matrices using arrays
	  ○ Concepts and implementation of 3D arrays
	  ○ Basic concepts of Multi-dimensional arrays
	• What is data structure?
	  ○ Need of data structures
		Cognitive Approach:
		
		-Basic Concepts

2	• Structures
	• Pointers and Dynamic Memory
	  ○ Pointer variables
	  ○ Assignment operator with pointers
	  ○ Dynamic variables and new Operator
	• Dynamic array and objects
	• Concepts of static and dynamic memory
	• Data Structure operations
		-Basic Concept
		-Q & A Session

3	• Linked lists
	  ○ Arrays vs. Linked list
	• Types of linked list
	  ○ Singly linked list
	  ○ Circular singly linked list
	  ○ Doubly linked list
	  ○ Circular doubly linked list
		Affective and Psychomotor Approach:
		
		-Quiz
		-Lab Practice
		-Creating Algorithm

4	• Applying operations on linked lists
	  ○ Traversing a linked list
	  ○ Inserting new node	

	○ Searching a node ○ Removing a node	
5	• Introduction to Queues • The Queue data structure • Application of queues • Array Representation of Queue ○ Algorithm for Addition of an Element to the Queue ○ Algorithm for Deletion of an Element to the Queue	-Quiz -Assignment -Lab Exercise
	First Mid Term Examination	
7-8	• The FIFO structure • Queue operations • Extended queue operations • The priority queues • The LIFO structure • Introduction to the stack data structure • Applications of stack • Stack operations • Stack specifications ○ List and arrays ○ Stacks ○ Reversing a list ○ Information hiding • Stack implementation ○ Using arrays ○ Using linked list • Methods of stack ○ Push ○ Pop	-Assignment -Lab Exercise -Quiz
9	• What is algorithm? • Properties of an algorithm • Designing the algorithms ○ Simple algorithm design with daily life examples ○ Algorithms for mathematical formulas • Complexity of algorithm ○ Time complexity ○ Space complexity • Analysis of algorithms • Big O Notation ○ Best-case analysis ○ Worst-case analysis ○ Average-case analysis	-Class Discussion and Q&A Session. -Assignment
10	• Simple sorting ○ Understanding why sorting is important ○ Bubble sort ○ Selection sort ○ Insertion sort	-Quiz -Class and lab Exercise
11	• Quicksort • Efficiency of Quicksort	- Assignment - Lab Exercise
	Second Mid Term Examination	
13	• Concept of Binary Trees • Why use binary trees • Tree terminology • Concept of Binary Search trees and how they work • Finding a node in a binary search tree • Inserting a node • Traversing the tree in In order, Pre and Post	

	order • Applications of tree traversing in sorting	
14-15	• Deleting a node in a Binary Tree with all three cases • Efficiency of Binary Trees • Handling duplicate nodes in BST • Applications of BST • The Huffman code: Encoding and Decoding • Coding a complete message • The AVL trees Overview	-Assignment -Quiz -Class and Lab Exercise
16	• Graphs o Introduction o Searches (DFS & BFS) • Revision	- Assignment / Project - Q & A Session

Text Book

1. Data structures using c++, 2nd edition, D.S. Malik

Reference Material

1. C++ plus Data Structures, 3rd edition, Nell Dale.

Course Learning Outcomes

	Course Learning Outcomes (CLO)
1	Demonstrate the knowledge of linear and non-linear data structures such as array, list, queue, stack, trees and graphs.
2	Implement algorithms for the efficient representation and manipulation of data.
3	Apply suitable data structures in different scenarios.

CLO-SO Map

	SO IDs										
CLO ID	**GA1**	**GA2**	**GA3**	**GA4**	**GA5**	**GA6**	**GA7**	**GA8**	**GA9**	**GA10**	**GA11**

CLO 1	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CLO 2	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CLO 3	**0**	**1**	**1**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Approvals

Prepared By	Muhammad Faiz
Approved By	Not Specified
Last Update	10/08/2020