CSE 212: Data Structures Sessional

COURSE INFORMATION				
Course Code: **Course Title:**	**CSE 212** **Data Structures Sessional**	**Lecture Contact Hours:** **Credit Hours:**		**14** **1.5**

PREREQUISITE

Course Code: CSE 122, CSE 211
Course Title: Structures Programming Sessional, Data Structures (Theory)

CURRICULUM STRUCTURE

Outcome-Based Education (OBE)

RATIONALE

This Data Structures sessional course is designed to provide a clear concept on the implementation of the essential parts of the data structures and algorithms related to computer science. This course begins with the implementation of some commonly used data structures including arrays, linked lists, stack, queues, trees, graphs and then covers various relevant important topics related to this course.

OBJECTIVE

1.To develop a general understanding of basic data structures and algorithms.
2. Develop hands-on skills in implementing core data structures.
3. Strengthen problem-solving abilities using algorithmic techniques.
4. Foster teamwork and technical communication.

LEARNING OUTCOMES & GENERIC SKILLS

Course Outcome (CO) of the Course		Bloom's Taxonomy	CP	CA	KP	Assessment Method
CO1	Identify advantages and disadvantages of specific algorithms and data structures.	P1		1	1	Hw, Cw, Q

CO2	Select basic data structures and algorithms for autonomous realization of simple programs or program parts.	P3		1	1	Hw, Cw, Q
CO3	Initiate practical knowledge to determine and demonstrate bugs in programs.	P5		1	1	Hw, Cw, Q
CO4	Formulate new solutions for problems or improve existing code using learned algorithms and data structures.	P6		1	1	Hw, Cw, Q, ASG

(CP- Complex Problems, CA-Complex Activities, KP-Knowledge Profile, T – Test; PR – Project; Q – Quiz; ASG – Assignment; Pr – Presentation; R - Report; LT – Lab Test)

COURSE CONTENT

1. Internal data representation, Abstract data types.
2. Algorithm performance and elementary asymptotic analysis (Introduction to Big-O notation).
3. Elementary data structures: array, linked list, stack, queue, tree and tree traversal, graphs and graph representation, heap, binary search tree.
4. Sorting algorithms, Searching: linear search and binary search;
5. Advanced data Structures: balanced binary search trees, skip list, advanced heaps; hashing.

SKILL MAPPING

Course Outcome (CO) of the Course		Program Outcome (PO)											
		1	2	3	4	5	6	7	8	9	10	11	12
1	Demonstrate advantages and disadvantages of specific algorithms and data structures.	H											
2	Select basic data structures and algorithms for autonomous realization of simple programs or program parts.	H											

3	Initiate practical knowledge to determine and demonstrate bugs in programs.		M										
4	Formulate new solutions for problems or improve existing code using learned algorithms and data structures.		H										

(H – High, M- Medium, L-low)

JUSTIFICATION FOR CO-PO MAPPING

Mapping	Level	Justification
CO1-PO1	High	Increase breadth and depth of knowledge by demonstrating advantages and disadvantages of specific algorithms and data structures
CO2-PO1	High	Increase breadth and depth of knowledge by selecting basic data structures and algorithms for autonomous realization of simple programs or program parts.
CO3-PO2	Medium	Analyse and formulate different methods of analysis to determine and demonstrate bugs in programs
CO4-PO2	High	Analyse and formulate different methods of analysis to formulate new solutions for problems or improve existing code using learned algorithms and data structure

TEACHING LEARNING STRATEGY

Teaching and Learning Activities	Engagement (Hours)
Face-to-Face Learning	-
Lecture	42
Practical / Tutorial / Studio	-
Viva	-

Self-Directed Learning	-
Report	-
Revision	-
Assessment Preparations	
Formal Assessment Continuous Assessment	10
Total	52

COURSE SCHEDULE

Week	Topics
1	Introduction and Overview
2	Control Structures, Complexity of Algorithms etc.
3	String Processing (Storing Strings, String Operations, Word Processing etc.)
4	String Processing (Storing Strings, String Operations, Word Processing etc.)
5	Arrays (Linear, Traversing, Inserting etc.)
6	Arrays (Deleting, Sorting, Searching etc.)
7	Multidimensional Arrays, Pointers, Records, etc.
8	Linked Lists (Representation, Traversing, Searching etc.)
9	Linked Lists (Memory allocation, Insertion, Deletion, etc.)
10	Stacks (Array representation of Stacks, Polish Notation, Quicksort etc.)
11	Recursion, Towers of Hanoi, Queues, Deques, Priority Queues, etc.
12	Binary Trees (Traversing, Searching, Inserting, deleting etc.), General Trees.
13	Graphs (Shortest Path, Traversing, Sorting etc.)
14	Quiz, Lab Viva etc.

ASSESSMENT STRATEGY

Components	Grading	CO	Bloom's Taxonomy
Continuous Evaluation in every class	50% (4/5% marks in each class)	CO1-CO4	P1
Report	10%	CO1-CO4	P3
Viva	20%	CO1-CO4	P6
Quiz	20%	CO1-CO4	P5
Total Marks	100%		
(CO = Course Outcome, C = Cognitive Domain, P = Psychomotor Domain, A = Affective Domain)			

REFERENCE BOOKS

1. Introduction to Algorithms (CLRS), Latest edition.
2. Data Structures and Algorithm Analysis in C++ 2014