Course Title:	*Data Structures and Algorithms*
Course Code:	SEN-217
Credit Hours Theory:	Three (3)
Credit Hours Lab (If Applicable):	One (1)
Instructor Name with Qualification:	Adeel M. Syed, MS SE
Course Objectives:	A detailed study of Basic Structures commonly used in Data Processing, Implementation (in C++) and Applications of basic data structures, A Comparative study of different Sorting and Searching Techniques
Course Learning Outcomes:	On completion of the course, students should have: • Good understanding of the basic data structure • The knowledge to implement abstract data types • The ability to use an appropriate data structure for the solution of a problem
Contents (Catalog Description):	This course will focus on data structures and algorithms for manipulating them. Data structures for storing information in tables, lists, stacks, queues, trees and graphs will be covered. Basic algorithms for creating, manipulating and using these structures will also be discussed. Different types of searching and sorting techniques will also be introduced and will be compared. Students will carry out a number of programming assignments, which will emphasize various aspects of data organization and manipulation process.
Recommended Text Books:	A. M. Tenenbaum, **Data Structures using C and C++,** Prentice-Hall.
Reference Books:	• Nell Dale, **C++ Plus Data Structures**, Jones and Bartlet, Inc. • Sahni, **Data Structures, Algorithms and Applications**, McGrawHill. • Mark Allen Weiss, **Data Structures and Algorithm Analysis in C++**, Addison Wesley. • **Theory and Problems of Data Structures**, Schaum's Outline Series. • Frank M. Carrano, **Data Abstraction and Problem Solving with C++**, Addison Wesley.
Helping Web Sites:	▪ https://www.cs.auckland.ac.nz/~jmor159/PLDS210/ds_ToC.html ▪ https://people.mpi-inf.mpg.de/~mehlhorn/Toolbox.html
General Instructions for students:	Home Works and Assignments Attendance is mandatory. Every class is important. All deadlines are hard. Under normal circumstances late work will not be accepted. Students are required to take all the quizzes. No make-up quizzez will be given under normal circumstances. Any form of cheating on exams/assignments/quizzes is subject to serious penalty

Attendance

75% attendance is mandatory. Latecomers will be marked as absent.

Evaluation Criteria

Assignments/projects	20%
Quizzes	10%
Mid-Term	20%
Final	50%

Quizzes Schedule

Quiz # 1	Week # 4
Quiz # 2	Week # 6
Quiz # 3	Week # 9
Quiz # 4	Week # 11

Assignments Schedule

Assignment	**Delivery date**	**Submission Date**
Assignment # 1	Week # 2	Week # 4
Assignment # 2	Week # 5	Week # 6
Assignment # 3	Week # 9	Week # 10
Assignment # 4	Week #11	Week #12

Sixteen Week Lesson Plan

Week 1

1st Lecture

Introduction and Overview, Elementary Data Organization

2nd Lecture

Overview of Data Structures, Basic Data Structure Operations

3rd Lecture

Abstract Data Types (ADTs)

Week 2

1st Lecture

Stacks: Definition, Basic Operations, Stack ADT and Applications

2nd Lecture

Application of Stacks: Checking the Validity of Expressions

3rd Lecture

Representing Stacks in C++

Week 3

1st Lecture

Application of Stacks: Infix, Postfix and Prefix Expressions, Algorithm to Evaluate a Postfix Expression

	2nd Lecture Application of Stacks: Algorithm to Convert an Infix Expression into Postfix 3rd Lecture Recursion **Week 4** 1st Lecture Queues: Definition and Basic Operations and ADT 2nd Lecture Applications of Queues, Quiz 1 3rd Lecture Representing Queues in C++ **Week 5** 1st Lecture Priority Queues, Quiz 1 return and discussion. 2nd Lecture Implementation of Priority Queues 3rd Lecture De-Queues **Week 6** 1st Lecture Linked Lists: Definition, Basic Operations and ADT, Quiz 2 2nd Lecture Linked implementation of Stacks and Queues 3rd Lecture Representing Linked Lists in C++ **Week 7** 1st Lecture Circular Linked Lists, Quiz 2 return and discussion. 2nd Lecture Doubly Linked Lists 3rd Lecture Addition of long integers using Linked List **Week 8** 1st Lecture Trees: Definitions and Basic Terminology 2nd Lecture Binary Tree Operations, Heaps 3rd Lecture Representing Binary Trees in C++

	Week 9 1st Lecture Application of Binary Trees: The Huffman Algorithm 2nd Lecture Trees and their Representation in C++, 3rd Lecture Application of Trees: Game Trees, Quiz 3 **Week 10** 1st Lecture Graphs: Definition and Basic Operations, Quiz 3 return and discussion. 2nd Lecture Representing Graphs in C++ 3rd Lecture Graph Search and Traversal Techniques **Week 11** 1st Lecture Application of Graphs: Minimum Cost Spanning Trees, Quiz 4 2nd Lecture Application of Graphs: Dijkstra's Shortest Path Algorithm 3rd Lecture Dijkstra's Shortest Path Algorithm **Week 12** 1st Lecture Sorting Techniques: General Background Exchange Sorts: Bubble Sort, 2nd Lecture Quick Sort, Quiz 4 return and discussion. 3rd Lecture Selection Sorts **Week 13** 1st Lecture Tree Sorts 2nd Lecture Insertion Sorts 3rd Lecture Merge and Radix Sorts **Week 14** 1st Lecture

Searching Techniques: General Background, Sequential Search

2nd Lecture

Indexed Sequential Search

3rd Lecture

Binary Search, Tree Search

Week 15

1st Lecture

Hashing: Basic Concepts

2nd Lecture

Hashing Function

3rd Lecture

Resolving Hash Clashes

Week 16

1st Lecture, 2nd Lecture, 3rd Lecture

Revision

Course Learning Outcomes mapping to Program Learning Objectives

CONTRIBUTION OF COURSE LEARNING OUTCOMES TO PROGRAMME LEARNING OUTCOMES													
SOFTWARE ENGINEERING		Data Structures and Algorithms											
No.	Program Outcomes	Course Learning Outcomes											
		1	2	3	4	5	6	7	8	9	10	11	12
1	Engineering Knowledge				√								
2	Problem analysis							√					
3	Design/Development of solutions							√					
4	Investigation												
5	Modern tool usage			√									
6	Engineer and society												
7	Environment and sustainability												
8	Ethics												
9	Individual and Team work				√								
10	Communication												
11	Project Management		√										
12	Lifelong learning				√								