National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science (SEECS)
Department of Electrical Engineering

DATA STRUCTURES AND ALGORITHMS			
Course Code:	CS250	**Semester:**	Fall 2015
Credit Hours:	3+1	**Prerequisite Codes:**	CS113 Introduction to Programming
Instructor:	Dr. Arsalan Ahmad	**Discipline:**	BESE5-B
Office:	Room#B-208, IAEC Building	**Telephone:**	
Lecture Days:	Tuesday, Thursday	**E-mail:**	arsalan.ahmad@seecs.edu.pk
Class Room:	As per timetable	**Consulting Hours:**	Tuesday 1500-1550; also, through appointment via e-mail
Knowledge Group:	Programming	**Updates on LMS:**	End of Week

Course Description:

The first part is related to basic data structures, and then we will consider sorting (both elementary and classic algorithms) and searching (trees, binary search trees, balanced search trees and heap sort). In the second part, we will discuss advanced topics related to graph theory, elementary graph algorithms and shortest path algorithms, as well as some applications using different classic algorithms.

Course Learning Outcomes (CLOs):		
At the end of the course the students will be able to:	**PLO**	**BT Level***
1. Choose appropriate data structures and their implementations for a given problem.	1	C-1, C-2
2. Employ the correct class of algorithms for any given situation.	2,3	C-2, C-3
3. Analyze a complex problem, and design and implement solutions for it.	2,3	C-4
4. Compare various algorithms based on accuracy, time complexity, and memory requirements.	4	C-4
* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain		

Mapping of CLOs to Program Learning Outcomes

PLOs/CLOs	CLO1	CLO2	CLO3	CLO4
PLO 1 (Engineering Knowledge)	√	√		
PLO 2 (Problem Analysis)		√	√	
PLO 3 (Design/Development of Solutions)			√	
PLO 4 (Investigation)				√
PLO 5 (Modern tool usage)				
PLO 6 (The Engineer and Society)				
PLO 7 (Environment and Sustainability)				

PLO 8 (Ethics)				
PLO 9 (Individual and Team Work)				
PLO 10 (Communication)				
PLO 11 (Project Management)				
PLO 12 (Lifelong Learning)				

Mapping of CLOs to Assessment Modules and Weightages (In accordance with NUST statutes)

To be filled in at the end of the course.

Assessments/CLOs	**CLO1**	**CLO2**	**CLO3**	**CLO4**
Quizzes: 7.5%				
Assignments: 7.5%				
OHT-1: 15%				
OHT-2: 15%				
Labs:25% (Lab deliverables 15 marks, Project: 10 marks)				
End Semester Exam:30%				
Total : 100 %				

Books:

Text Book: Adam Drozdek. Data Structures and Algorithms in C++, 3/e (2005)

Reference Books:

1) T. H. Cormen, Charles E. Leiserson, R. L. Rivest, Clifford S. Introduction to Algorithms 3/e (2009)
2) Mark A. Weiss. Data Structures and Algorithm Analysis in C++, 3/e (2006)
3) Steven S Skiena. The Algorithm Design Manual, 2/e (2008)
4) Relevant resources/ references will be highlighted (wherever relevant) during the lecture and at the end of week, slides will be uploaded to LMS.

Main Topics to be Covered:

Introduction to Data Structures and Algorithms	Priority Queues
Review C++: Pointers	Binary Heaps
Review C++: Dynamic Memory Allocation and Classes	Hash Tables
Algorithm Analysis	Chained Hash Tables
Design of Algorithms	Graph Theory
Running Time	Topological Sort
Big-O Notation	Spanning Trees
Abstract Lists	Minimum Spanning Trees: Prim's Algorithm

Abstract Stack	Minimum Spanning Trees: Kruskal's Algorithm
Stack Application	Shortest Path: Dijkstra's Algorithm
Abstract Queue	Trees
Containers: Queue Application	Binary Trees
Recursion	Binary Search Trees
Sorting	AVL Trees

Lecture Breakdown:

Week No.	Topics
1	Lecture-1-3: Introduction to Algorithms and data structures, Array, linked list, singly linked list
2	Lecture-4-6:Stacks and queues (linked list implementation)
3	Lecture-7-9:Doubly linked list, circular linked list
4	Lecture-10-12:Concept in running time complexity, function growth
5	OHT-1
6	Lecture-13-16:Sorting algorithms
7	Lecture-17-19:Sorting algorithms
8	Lecture-20-23:Introduction to trees
9	Lecture-24-26:Binary search tree operations
10	Lecture-27-29:AVL trees, priority queues
11	OHT-2
12	Lecture-30-33:Binary heaps, hash tables
13	Lecture-34-36:Introduction to graphs
14	Lecture-37-39:Graph implementation and search operations
15	Lecture-40-43:Topological sort, spanning trees
16	Special Topics / Presentations

17	Revision
18	Final Exam

Lab Experiments:	
Lab 01:	Understand pointers and dynamic memory
Lab 02:	Implement singly-linked lists using pointers
Lab 03:	Solve a practical problem using linked lists
Lab04:	Implement doubly linked lists
Lab 05:	Implement stacks and queues
Lab 06:	Implement sorting algorithms
Lab 07:	Implement more sorting algorithms
Lab 08:	Use recursion to solve a problem
Lab 09:	Implementation of trees
Lab 10:	Implement binary search tree
Lab 11:	Implement binary heap
Lab 12:	Implement graphs
Lab 13:	Project Evaluation

Tools / Software Requirement:

- VC++ using MS Visual Studio
- gcc using Cygwin

Grading Policy:	
Quiz Policy:	The quizzes may be announced or unannounced and will normally last for ten minutes. The questions would test the concepts involved in last few lectures. Number of quizzes that will be used for evaluation is at the instructor's discretion. Grading for quizzes will be on a fixed scale of 0 to 10. All quizzes will count towards the total (No 'best-of' policy).
Assignment Policy:	In order to develop comprehensive understanding of the subject, assignments will be given. Late assignments (by up to 2 days) will be accepted but penalized as per the following formula: Less than a day late: 15% penalty More than 1 day late but less than 2 days late: 30% penalty More than 2 days late: ***not accepted.*** All assignments will count towards the total (No 'best-of' policy). The students are expected to submit assignments that are their own work. Students may collaborate by discussing general

	concepts and approach but not the specific answer/ technique that is asked for. In no circumstances should a student present someone else's work as their own work. Copying of assignments will be dealt with severely by awarding ZERO marks for the whole evaluation component and referring any occurrences to the disciplinary committee.
Plagiarism:	SEECS maintains a zero tolerance policy towards plagiarism. While collaboration in this course is highly encouraged, you must ensure that you do not claim other people's work/ ideas as your own. Plagiarism occurs when the words, ideas, assertions, theories, figures, images, programming codes of others are presented as your own work. You must cite and acknowledge all sources of information in your assignments. Failing to comply with the SEECS plagiarism policy will lead to strict penalties including zero marks and referral for disciplinary action.