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

Data Structures and Algorithms

Course Code:	CS-250	**Semester:**	Fall 2017
Credit Hours:	3+1	**Prerequisite Codes:**	CS110 Fundamentals of Computer Programming
Instructor:	Dr. Faisal Shafait	**Class:**	BSCS-6 (AB)
Office:	A 305	**Telephone:**	03330544462
Lecture Days:	Mon and Wed	**E-mail:**	faisal.shafait@seecs.edu.pk
Class Room:	IAEC Lecture Hall	**Consulting Hours:**	Will be arranged on requirement
Lab Engineer:	Nadeem Nawaz	**Lab Engineer Email:**	nadeem.nawaz@seecs.edu.pk
Knowledge Group:	Programming	**Updates on LMS:**	After lecture

Course Description:

Introduction to abstract data structures and their implementations: lists, stacks, queues, trees, and graphs. Concept in running time complexity, function growth, and recurrences. Sorting algorithms such as insertion sort, bubble sort, selection sort, merge sort, and quick sort. Introduction to trees, binary trees, binary search tree operations. Introduction to graphs, depth first search and breadth first search, shortest paths, and topological sort.

Course Objectives:

Learn about basic abstract data structures and implement them efficiently. Understand the importance of data structures in developing and implementing efficient algorithms. Introduce commonly used sorting algorithms and compare them based on computational efficiency and memory requirements. Discuss the use of trees and graphs to solve more complex problems. Be able to apply the correct algorithms and data structures for a given problem.

Course Learning Outcomes (CLOs):		
At the end of the course the students will be able to:	**PLO**	**BT Level***
1. Understand the fundamentals of data structures and algorithms	A	C-2
2. Apply Data Structures and Algorithms to solve complex engineering problems.	B	C-3
3. Select appropriate data structures and employ suitable algorithms for solution development.	C	C-3
4. Investigate and evaluate various algorithms based on accuracy, time complexity, and memory requirements.	J	C-6

* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain
Knowledge(C-1), Comprehension(C-2), Application(C-3), Analysis(C-4), Synthesis(C-5), Evaluation(C-6)
Perception(P-1), Set(P-2), Guided Response(P-3), Mechanism(P-4), Complete Overt Response(P-5), Adaption(P-6),
Organization(P-7), Receiving(A-1), Responding(A-2), Valuing(A-3), Organization(A-4), Internalizing(A-5)

Mapping of CLOs to Program Learning Outcomes

PLOs/CLOs	CLO1	CLO2	CLO3	CLO4
PLO A (Computing Knowledge)	X			
PLO B (Problem Analysis)		X		
PLO C (Design/Development of Solutions)			X	
PLO D (Individual & Team Work)				
PLO E (Ethics)				
PLO F (Communication)				
PLO G (Societal Impact)				
PLO H (Lifelong Learning)				
PLO I (Modern Tool Usage)				
PLO J (Investigation)				X
PLO K (Project Management)				

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

Assessments/CLOs	
Theory: 75%	
• Quizzes: 10%	
• Assignments: 10%	
• Project: 10%	
• OHT-1: 15%	
• OHT-2: 15%	
• End Semester Exam: 40%	
Practical: 25%	
• Labs Assignments: 100%	
Total : 100 %	
To be filled in at the end of the course.	

Books:

Text Book:
1. Adam Drozdek. Data Structures and Algorithms in C++, Fourth Edition

Reference Books:
1. Steven S Skiena. The Algorithm Design Manual, Second Edition
2. T. H. Cormen, Charles E. Leiserson, R. L. Rivest, Clifford S. Introduction to Algorithms, Third Edition
3. Mark A. Weiss, Data Structures and Algorithm Analysis in C++, Third Edition

Topics to be Covered:

1. Linked lists	2. Sorting algorithms
3. Stacks and Queues	4. Trees
5. Running time complexity	6. Introduction to graph theory
7. Recurrence	8. Introduction to greedy algorithms

Lecture Breakdown:

Week No.	Topics
1	Introduction to data structures and algorithms
2	C++ Programming Review, Concept of Version Control
3	Concept of run time complexity, function growth
4	Linked lists
5	Containers: Arrays, Stack Applications, Queue Implementation
6	Linked Lists Run Time Analysis
7	**OHT-1**
8	Concept of Trees and Hash Tables
9	Concept of Graphs
10	Recursions + Merge sort + Linear Time Sorting
11	Quick sort + Radix Sort
12	Priority Queues
13	**OHT-2**
14	AVL Trees
15	Topological Sort + Minimum Spanning Tree
16	Shortest Path Algorithm
17	Greedy Algorithms
18	Dynamic Programming
19	**ESE**

Lab Experiments:

Lab 01:	Implementation of pointers and dynamic memory allocation
Lab 02:	Practice the concept of version control and test cases
Lab 03:	Implementation of linked list with its operations
Lab 04:	Use knowledge of linked lists to implement a small functionality.
Lab 05:	Implementation of stacks and queues in different problems
Lab 06:	Implementation of Hash tables
Lab 07:	Execution of a code to represent the social network
Lab 08:	Implementation of three Sorting Algorithms and compare them
Lab 09:	Concepts of Recursion, take a program using abstract stack and solve a popular puzzle
Lab 10:	Implementation of Merge sort and Radix Sort
Lab 11:	Implementation of quick sort.
Lab 12:	Applied Data Structures: ChatBots Lab Activity
Lab 13:	Implementation of binary heap (min heap)
Lab 14:	Applied Algorithms: Face Detection

Tools / Software Requirement:
• VC++ using MS Visual Studio

Grading Policy:	
Quiz Policy:	The quizzes will be unannounced and normally last for ten minutes. The question framed is to test the concepts involved in last few lectures. Number of quizzes that will be used for evaluation is at the instructor's discretion.
Assignment Policy:	In order to develop comprehensive understanding of the subject, assignments will be given. Late assignments will not be accepted / graded. All assignments will count towards the total (No 'best-of' policy). The students are advised to do the assignment themselves. Copying of assignments is highly discouraged and violations will be dealt with severely by referring any occurrences to the disciplinary committee. The questions in the assignment are meant to be challenging to give students confidence and extensive knowledge about the subject matter and enable them to prepare for the exams.
Lab Conduct:	The labs will be conducted for three hours every week. A lab handout will be given in advance for study and analysis The lab handouts will also be placed on LMS. The students are to submit their results by giving a lab report at the end of lab for evaluation. One lab report per group will be required. However, students will also be evaluated by oral viva during the lab.
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 in assignments and referral to the academic coordination office for disciplinary action.

Program Learning Outcomes (PLOs)

Program outcomes are the narrower statements that describe what students are expected to know and be able to do by the time of graduation. These relate to the knowledge, skills and attitude that the students acquire while progressing through the program. The program must demonstrate that by the time of graduation the students have attained a certain set of knowledge, skills and behavioral traits, at least to some acceptable minimum level. Specifically, it is to be demonstrated that the students have acquired the following graduate attributes:

(i) **Engineering Knowledge:** An ability to apply knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

(ii) **Problem Analysis:** An ability to identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

(iii) **Design/Development of Solutions:** An ability to design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

(iv) **Investigation:** An ability to investigate complex engineering problems in a methodical way including literature survey, design and conduct of experiments, analysis and interpretation of experimental data, and synthesis of information to derive valid conclusions.

(v) **Modern Tool Usage:** An ability to create, select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling, to complex engineering activities, with an understanding of the limitations.

(vi) **The Engineer and Society:** An ability to apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice and solution to complex engineering problems.

(vii) **Environment and Sustainability:** An ability to understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate knowledge of and need for sustainable development.

(viii) **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.

(ix) **Individual and Team Work:** An ability to work effectively, as an individual or in a team, on multifaceted and /or multidisciplinary settings.

(x) **Communication:** An ability to communicate effectively, orally as well as in writing, on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

(xi) **Project Management:** An ability to demonstrate management skills and apply engineering principles to one's own work, as a member and/or leader in a team, to manage projects in a multidisciplinary environment.

(xii) **Lifelong Learning:** An ability to recognize importance of, and pursue lifelong learning in the broader context of innovation and technological developments.