THE UNIVERSITY OF LAHORE

Department of Computer Science & IT

Course Outline

CS-09204 Data Structures & Algorithms

Degree Program	BSCS
Credit Hours	(3+1) Credit(s)
Co-requisite (s)	None
Pre-requisite(s)	*CS 2203 Object Oriented Programming*
Post Requisite of	*CS-3444 Design and Analysis of Algorithms*
Weekly tuition pattern	2 sessions (90 min each session)
Course Mentor	Dr. Mehtab Afzal
Course Instructor	

COURSE DESCRIPTION:

Data structures are essential building blocks for designing efficient algorithms. Thus, they play a central role in computer science and are important in many other areas of other fields. It's rarely the language that matters in a program, it's the data structures and how they relate. And range of elegant Algorithms is possible only if we have appropriate Data Structures for the Job.

The purpose of this course is to provide the students with solid foundations in the basic concepts of programming: data structures and algorithms. The main objective of the course is to teach the students how to select and design data structures and algorithms that are appropriate for problems that they might encounter. This course is also about showing the correctness of algorithms and studying their computational complexities.

COURSE OBJECTIVES:

1. To understand the design of fundamental data structures as well as algorithms that operate on them.
2. Analysis of Wide Range of Data Structures available
3. Comparison of Data Structures Based on their Analysis
4. Implementing new Data Structures for the real time & industrial needs.
5. To provide rigorous 'hands-on' experience with implementing different data structures in a programming language

COURSE OUTCOMES:

By the end of this course, Students will understand:

1. Purpose and mathematical background of algorithm analysis and be able to apply this to determine the run time and memory usage of algorithms;
2. Analysis of data structures will lead us to know about the advantages and disadvantages of different data structures available.
3. The comparison of different data structures and algorithms will help in selection of efficient data structures and algorithms for our problems.
4. Prepare the students for more advanced material that students will encounter in later courses.
5. Implement well-known data structures such as dynamic arrays, linked lists, stacks, queues, trees and graphs in C++.

COURSE STRUCTURE:

1. Presentation by lecturer
2. Class mini tasks/assignments
3. Lab tasks

COURSE DURATION:

1. The class of 1.5 hours will be held twice a week.
2. However Extra Sessions may be scheduled if needed.
3. 1 lab of 3 hours will be held once a week.

COURSE STYLE:

The course will be delivered both in a classroom and as well as in a Lab.
In Class Implementation,
Live Coding Demo,
Power point slides (if needed),

Class Discussion,
Notes.
Exams can be held open-book/open-notes, as announced by the instructor.
Extra credit can be awarded to students based on extra achievements, e.g. participation in programming/project competitions, solving online programming competition problems, etc. on the instructor's discretion.

LANGUAGES, TOOLS, TEXTBOOKS & OTHER RESOURCES:

- Students are expected to know programming. They will be required to do programming in assignments, project and exams. Although the course deals with concepts of Data Structures, their implementation is necessary to ensure correct understanding.
- Although there is no restriction of programming language, however C++ with object oriented approach will be followed in the classroom.
- There is no restriction of IDE. Students can use any of ANSI and Visual C++ Coding Standards

The following books are recommended for the course. Students are expected to have at least one book with them, whether their own, or from the library.

1. Data Structures Using C++

Author: D. S. Malik

2. Fundamentals of Data Structures in C++

Author: E. Horowitz, S. Sahni, and D. Mehta, Computer Science Press.

3. Data Structures, Algorithms, And Applications In C++, 2nd Edition,

Author: Sartaj Sahni

4. Data Structures and Algorithms in C++, 2nd Edition
Author: Micheal T. Goodrich, Roberto Tamassia and David Mount John Wiley and Sons, Inc

5. Algorithms (4th Edition)

Author: Robert Sedgewick (Author), Kevin Wayne (Author)

6. Introduction to Algorithms

Author: Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein. MIT Press

7. Classic Data Structures

Author: Samanta Debasis (Author)

8. Data Structures and Algorithms Analysis in C++

Author: By Clifford A Sheffard

9. Data Structures and Algorithms Analysis with C++ (4th Edition)
By Mark Allen Weiss

Course Outline:

The lecturers should aim to complete the following topics before the mid/final term examination as prescribed in the course outline below:

	Lecture	Topics / Sub-Topics
1	1	Introduction of Data Structures and Algorithm, Memory Organization, Pointers, Arrays and their working.
	2	Static Array, Dynamic Arrays in detail, Sorted/Un Sorted Array, Issues with Arrays, Sparse Array and Sparse Matrix
2	3	Asymptotic Analysis: Basics, Efficiency of an algorithm. Measuring runtime complexity of an algorithm Linear & binary searching (implementation, run time analysis & asymptotic analysis)
	4	Sorting Algorithm (Bubble, Selection, Insertion), implementation, run time analysis & asymptotic analysis
3	5	Stack, basic concepts
	6	Implementation of Stack using arrays, Applications of Stack.
4	7	Evaluation of Expression, Reverse Polish notation, Infix, Postfix and Prefix expressions using stack
	8	Queue basic concept. Implementation using arrays.
5	9	Applications of Queue. Circular Queue, Priority Queue
	10	Dynamic Data Structures Introduction to Node and Linking, Single Linked List and List Data Structure in detail.
6	11	Operations of a Single Linked List & implementation, Run time analysis & asymptotic analysis of Single Linked List operations Benefit of Single Linked List, Issues with Single Linked List.

	12	Double Linked List, Concept, implementation, different operations Comparison of Single Linked List with Double Linked List

7	13	Circular Linked List, Basic concept, characteristics Comparison of Circular with Single & Doubly Linked List
	14	Implementation of Stack and Queue using Linked List
8	**MID-TERM EXAMINATION**	
9	15	Solution of mid and discussion Graph Theory in detail, Their Applications
	16	Recursion Divide & Conquer Merge Sort (Algorithm, Implementation)
10	17	Quick Sort (Algorithm, Implementation)
	18	Trees: concepts, their applications Binary Tree, Binary tree Insertion, deletion.
11	19	Breadth First Search, Depth First Search Tree Analysis
	20	Introduction to Binary Search Tree Binary Search Tree Implementation
12	21	BST Deletion & Updation BST Analysis
	22	Pre Order, In order, Post Order Traversal Example of Expression Tree, Post Fix and Infix
13	23	Balanced Trees, their Implementation
	24	Balanced Trees Algorithmic Analysis
14	25	Heap Concepts
	26	Heap Operations, Analysis

15	27	Static Hashing, Hash Functions, Collision Resolution Techniques, Chaining, Open Addressing
	28	Re Hashing Implementation and their Algorithmic Analysis
16	29	Adjacency Matrix, Adjacency List
	30	Graph Traversals, Breadth First Search, Depth First Search

FINAL EXAMINATION

ASSESSMENT CRITERIA:		
No.	**Assessment**	**Percentage**
1.	Assignments & Project	10%
2.	Quizzes	10%
3.	Lab	20%
4.	Midterm	20%
5.	Final	40%
	Total	100%

Note: The above criterion is not fixed. So, it can be changed as per respective teacher's understanding. Mid and Final exams will be open book, but closed notes/slides.

Attendance Requirements

You are expected to attend all lectures, tutorials, make up sessions and lab sessions, and any other classroom activity. Where you fail to attend classes, you cannot expect the lecturer to brief you on what you have missed. You are responsible for your attendance, not the academic staff. Attendance at lab sessions will be strictly monitored, and failure to attend will be taken into account. It is recommended to you to come in time.

Submission and Collection of Assignment

All assignments should be submitted before the time mentioned on Course website/portal when they are due. Result will be uploaded on course website/portal.

General Information

Students are required to be familiar with the university code of conduct available on the university website, study guide and prospectus, and to abide by its terms and conditions. In addition, the following guidelines must also be followed:

Copying of Copyright Material by Student

A condition of acceptance as a student is the obligation to abide by the University's policy on the copying of copyright material. This obligation covers photocopying of any material, and the recording off air, and making subsequent copies, of radio, television or Internet broadcasts, and photocopying textbooks. Students who flagrantly disregard University policy and copyright requirements will be liable to disciplinary action under the Code of Conduct.

Guidelines to Avoid Plagiarism

Whenever you copy more than a few words from any source, you must acknowledge that source by putting the quote in quotation marks and providing the name of the author. Full details must be provided in your bibliography. If you copy a diagram, statistical table, map, etc., you must acknowledge the source.

Students are encouraged to co-operate, but collusion is a form of cheating. Students may work together in obtaining references, discussing the content of the references and discussing the assignment, but when they write, they must write alone.

Referencing For Written Work

To attain these qualities, the school recommends use of IEEE style of referencing, which specifies the authors, article name, place of publishing, publisher information, date.

Approval

Checked by,	Approved by Dr. Mehtab Afzal Signature
Curriculum Review Committee	