COURSE DESCRIPTION FORM

INSTITUTION National University of Computer and Emerging Sciences (NUCES-FAST)
BS(CS), BS(SE), BS(CY), BS(AI)

PROGRAM (S) TO BE EVALUATED

A. Course Description

Course Code	CS2001
Course Title	Data Structures
Credit Hours	3+1
Prerequisites by Course(s) and Topics	Object-oriented Programming (CS1004)
Assessment Instruments with Weights (homework, quizzes, midterms, final, programming assignments, lab work, etc.)	Midterm Exam 1: 15 (1 Hour written exam) Midterm Exam 2: 15 (1 Hour written exam) Assignments (programming based) x 2: 10 Quizzes (Best 3 out of 4): 10 Final: 50 (3 Hours Written Exam)
Course Coordinator	Farrukh Hasan Syed
URL (if any)	-
Current Catalog Description	-
Textbook (or **Laboratory Manual** for Laboratory Courses)	**Textbook:** Algorithms by Robert Sedgewick and Kevin Wayne Data Structures and Algorithms in C++, 4th Edition by Adam Drozdek **Reference books:** Data Structure and Algorithms Analysis in C++ Mark Allen Using C++ -- A Practical Implementation by Sachi Nandan Mohanty and Pabitra Kumar Tripathy

Reference Material	Data Structures Using C++ by VARSHA H. PATIL Oxford University Press Data Structures and Algorithm Analysis by Clifford A. Shaffer Open Data Structures in C++ Open Data Structures in Java
Course Goals	**A. Course Learning Outcomes (CLOs)** 1. *Use & explain* concepts related to basic and advanced data structures and describe their usage in terms of common algorithmic operations **[Bloom's Taxonomy Level:** 3**, Learning Domain:** Cognitive**]** 2. *Solve* recursive problems efficiently using Backtracking **[Bloom's Taxonomy Level:** 3**, Learning Domain:** Cognitive**]** 3. *Compare* different data structures in terms of their relative efficiency and *design* effective solutions and algorithms that make use of them. **[Bloom's Taxonomy Level:** 6**, Learning Domain:** Cognitive & Psychomotor**]** 4. *Transform* cycling-bearing graphs into acyclic tree structures for minimum cost traversal **[Bloom's Taxonomy Level:** 6**, Learning Domain:** Cognitive & Psychomotor**]** **B. Program Learning Outcomes** CLO-1 **1. Computing Knowledge** — Apply knowledge of mathematics, natural sciences, computing fundamentals, and a computing specialization to the solution of complex computing problems. CLO-2 **2. Problem Analysis** — Identify, formulate, research literature, and analyze complex computing problems, reaching substantiated conclusions using first principles of mathematics, natural sciences, and computing sciences. CLO-3 **3.Design/Develop Solutions** — Design solutions for complex computing problems and design systems, components, and processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

		1	2	3	4	5	6	7	8	9	10	11	12
CLOs													
	4												

Topics Covered in the Course, with Number of Lectures on Each Topic (assume 15-week instruction and one-hour lectures)

1. Topics to be covered:

List of Topics	No. of Weeks	Contact Hours	CLO
ADT, C++/ Java Language Specification, Pointers revisited/ pass-by-reference and pass by value, Rule of Three, Dynamic Safe Arrays	**1**	**3**	**1**
List (Singly Linked List), List (Doubly Linked List), List (Circular Linked List), Linear, Binary & Interpolation Search using Arrays and Linked Lists	**1**	**3**	**1,3**
Elementary Sorting Techniques (Bubble sort, Selection Sort, Insertion Sort, Radix Sort, Shell sort, Comb sort)			

		2	**6**	**1, 3**	

	Recursion, it's types, issues and Backtracking (with examples), Stack, Queue, their implementation strategies and applications (Simulation of recursion)	**1**	**3**	**1,2,3**
	========== Mid-term 1 Exam ==========			
	Advanced Sorting Techniques (Merge sort, Quick sort)	**1**	**3**	**3**
	Binary trees and their properties (Full Binary Tree, Complete Binary Tree), Binary Search Trees, their operations and applications, skewness and issues	**2**	**7**	**1, 2, 3**
	Balance in Binary Search Trees, AVL Trees, 2-3 trees, B-trees	**2**	**3**	**2, 3**
	========== Mid-term 2 Exam ==========			
	Priority Queues, Heaps as Priority Queues, Heap Sort	**1**	**3**	**1, 3**
	Hashing, Hash Functions, Collision-resolution Techniques, Rehashing	**1**	**3**	**1, 3**
	String search (Brute force, Rabin Karp, Boyer Moore, Knuth Morris)	**1**	**3**	4
	Minimum Spanning Trees, Graph Algorithms, Topological Sort, Graphs and their representation and traversal, Shortest Path Problem	**1**	**3**	**4**
	========== Final Exam ==========			
	Total	**16**	**48**	

Laboratory Projects/Experiments Done in the Course	There will be weekly labs starting from the first week. The following is a summary of the Lab exercises given to Students: • Introduction to Data Structures and their implementation. • Writing & using dynamic safe arrays • Solving recursive problems using Backtracking in programs • Implementation of Linked Lists • Linked List based implementation of primitive Data Structures • Implementing Sorting Algorithms • Implementing Binary Trees and writing functions for their properties • Implementing Binary Search Trees using Structures and Classes • Writing functions for tree traversal and maintaining balance

	• Implementing graphs and writing functions for their traversal			
Programming Assignments Done in the Course	Assignments related to Backtracking, Stacks & Queues, Binary Search Trees and traversal			
Class Time Spent on (in credit hours)	**Theory**	**Problem Analysis**	**Solution Design**	**Social and Ethical Issues**
	15	15	13	0
Oral and Written Communications	Every student is required to submit at least __1__ written report of typically _6__ pages and to make _1__ oral presentations of typically __10__ minute's duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.			

Instructor Name: _____

Instructor Signature: _____

Date: