CS2001 – Data Structures

Course Outline

Instructor

Waqas Ali
Department of Computer Science
National University of Computer and Emerging Sciences
Email: waqas.ali@nu.edu.pk

Course Objectives

- Covering well-known data structures such as dynamic arrays, linked lists, stacks, queues, trees and graphs etc.
- To Prepare students for (and is a prerequisite for) the more advanced material students will encounter in later courses.
- Implementing data structures and classical computer science problems in C++.

Textbook

Fundamentals of Data Structures in C++ by horowitz, sahni and mehta.

Course Organization

Lecture No	Duration	Topics
1	1.5 hrs	Course Intro Revision of Pointers - Dereferencing - Dynamic Memory Allocation (new and delete operators) - Inaccessible objects and Dangling pointers
2	1.5 hrs	List Data Structres - List Operations - Implementation - Array based implementation of Lists
3	1.5 hrs	Linked List - Creating a node in linked list - Linking nodes via pointers - Representation in memory - Linked list vs Array based list - Operations on linked list
4	1.5 hrs	Linked List (implementation) - Creating a linked list in C++ - Insertion in a linked list in C++

5	1.5 hrs	Linked list (implementation) - Searching in a linked list in C++ - Deletion from a linked list in C++ Doubly linked list - Intro
6	1.5 hrs	Doubly linked list - insertion - deletion Circular linked list (ring) - insertion - deletion
7	1.5 hrs	Josephus Problem Doubly Ring (circular doubly linked list) Stack and its operations
8	1.5 hrs	Stack - Array based Implementation - Linked List based implementation
9	1.5 hrs	Infix, prefix, postfix Stack Uses - Postfix evaluation
10	1.5 hrs	Stack Uses - Bracket matching
11	1.5 hrs	Queues - Array based Implementation - Linked List based Implementation
12	1.5 hrs	Rearranging rail road cars using stack and queues
13	1.5 hrs	- Recursion - Trees
14	1.5 hrs	Binary Trees Complete Binary Trees Applications of Binary Trees - Checking duplicate values
15	1.5 hrs	Binary Trees – Implementation
16	1.5 hrs	Binary Search Trees
17	1.5 hrs	AVL Trees
18	1.5 hrs	AVL Trees (Insertion)
19	1.5 hrs	AVL Trees (Deletion)
20	1.5 hrs	Uses of binary trees - Expression Trees - Huffman Codes
21	1.5 hrs	Threaded Binary Trees
22	1.5 hrs	Heap
23	1.5 hrs	Heap
24	1.5 hrs	Sorting Algorithms
25	1.5 hrs	Searching Algorithms
26	1.5 hrs	Hashing
27	1.5 hrs	Hashing
28	1.5 hrs	Graphs
29	1.5 hrs	Breadth first search and Depth first search