CS-2101: Data Structures and Algorithms

Instructor Dr. Muhammad Safyan Semester 3, 2023

# Course Description

Data structures play a central role in modern computer science. Data structures are essential building blocks in obtaining efficient algorithms. This course is all about how to store your data and how to access it efficiently. Data Structures and Algorithm Analysis in C++ describes data structures, methods of organizing large amounts of data, and algorithm analysis, the estimation of the running time of algorithms. As computers become faster and faster, the need for programs that can handle large amounts of input becomes more acute. Paradoxically, this requires more careful attention to efficiency, since inefficiencies in programs become most obvious when input sizes are large. By analyzing an algorithm before it is actually coded, students can decide if a particular solution will be feasible.

# Required Materials

Required materials could be found at the course page: https:// github.com/safyanch/DSA-2023

# Prerequisites

Prerequisites: Programming Fundamentals

# Course Objectives

* To understand the design of fundamental data structures as well as algorithms that operate on them
* To understand the fundamental trade offs in the design of the data structures
* To introduce tools for analyzing the time and space complexity of data structures
* To provide rigorous hands-on experience with implementing different data structures in a programming language

# Learning Outcomes

1. Students will be able to understand basic data structures
2. Students will become aware of how data structures are used in real-world applications
3. Students will understand the fundamental tradeoffs that exist in the design of data struc­tures
4. Students will be able to compare the time and space efficiency of different data structures
5. Students will be able to appreciate how changing application requirements can lead to new data structures
6. Students will be able to write programs to efficiently manipulate, store, and retrieve data

# Course Structure

**Schedule and weekly learning goals**

The schedule is tentative and subject to change. The learning goals below should be viewed as the key concepts you should grasp after each lecture, and also as a study guide before each exam, and at the end of the semester.

## lecture 1

**Introduction**

* + - classes
    - Inline Function
    - Header Files
    - Templates
    - Template Function
    - Operator Overloading

## lecture 2

* + - Pointers

## lecture 3

* + - Arrays-mauplation

## lecture 4

**Memory Allocation**

* + - New operator
    - Delete operator
    - Memory Leak
    - Dangling pointer
    - Constructor
    - Destructor

## lecture 5

**Analysis**

* + - Asymptotic notations
    - Big O notation
    - Computational complexity
    - Best case
    - Average case
    - Worst case

## lecture 6

**Quiz 1**

## lecture 7

**Linked List**

* + - node- structure

## lecture 8

* + - Add Node
    - Delete Node
    - Insert Node

## lecture 9

Circular Linked List

## lecture 10

Double Linked List

## lecture 11

**Revision**

## lecture 12

Stack

## lecture 13

Queue

## lecture 14

* + - Prefix
    - Infix
    - Postfix

## lecture 15

**Quiz 2**

## lecture 16

**Mid Term Exam**

## lecture 17

Queue

## lecture 18

* + - Assignment-1
    - Priority Queue-
    - Heap

## lecture 19

**Tree**

## lecture 20

Binary Tree

## lecture 21

Binary search Tree

## lecture 22

Revision

## lecture 23

Avl Tree

## lecture 24

**Quiz 3**

## lecture 25

**Graphs**

## lecture 26

**Minimum Spanning Tree**

## lecture 27

Kruskal Algorithm

## lecture 28

Revision

## lecture 29

searching

## lecture 30

Assignment viva

## lecture 31

**Quiz 4**

## lecture 32

Exercise