



Saveetha School of Engineering
Saveetha Institute of Medical and Technical Sciences
Department of Computer Science & Engineering



Course Code: CSA03

Course Title: Data Structures

ASSIGNMENT 1 CO/PO MAPPING (Total: 50 Marks)

Course Objectives: The course on Data structures aims to provide the students with the following:

1. Understand the abstract properties of various linear and non-linear data structures such as stacks, queues, lists, trees and graphs.
2. Identify the advanced data structures such as balanced search trees, hash tables, priority queues
3. Analyze the various searching and sorting algorithms, including linear search, binary search, insertion sort, selection sort, heap sort and quick sort.
4. Choose the appropriate data structure and algorithm for specified applications.
5. To understand about writing algorithms and step by step approach in solving problems with the help of fundamental data structures.

Course Outcomes: On successful completion of the course, the student will be able to:

1. Design and implement an appropriate linear data structures and nonlinear data structures.
2. Understand the computational efficiency of the principle algorithms for sorting, searching, and hashing.
3. Demonstrate different methods for traversing trees and graphs.
4. Demonstrate the appropriate data structure and algorithm for specified applications.
5. Ability to analyze algorithms and algorithm correctness.

Blooms Taxonomy Levels (BTL)

1. Remembering 2. Understanding 3. Applying 4. Analyzing 5. Evaluating 6. Creating

Sl No	Assignment Questions	CO / K Level								
1	<p>(i)Can you declare an array without assigning the size of an array? (ii)In the railway station, tickets are provided at a counter. Name a suitable data structure for this. Explain the various operations performed in that data structure. Write a C program to insert and delete operations using array implementation. (10 Marks)</p> <p>Rubrics:</p> <table><tr><td>Declaring array in correct size</td><td>(2 Marks)</td></tr><tr><td>Program Logic</td><td>(4 Marks)</td></tr><tr><td>Program Execution</td><td>(2 Marks)</td></tr><tr><td>Program Result</td><td>(2 Marks)</td></tr></table>	Declaring array in correct size	(2 Marks)	Program Logic	(4 Marks)	Program Execution	(2 Marks)	Program Result	(2 Marks)	CO2 / K5/K6
Declaring array in correct size	(2 Marks)									
Program Logic	(4 Marks)									
Program Execution	(2 Marks)									
Program Result	(2 Marks)									
2	<p>(i)Convert the infix expression $((A-B)+C*D)/((E+F)-C)$ into postfix expression (ii)The objective of the game is to move the entire stack to another rod, obeying the rules: Only one disk may be moved at a time. Each move consists of taking the upper disk from one of the rods and sliding it onto another rod, on top of the other disks that may already be present on that rod. No disk may be placed on top of a smaller disk. You are expected to implement two algorithms to solve this puzzle for n disks: Using recursion Without using recursion: Here, we expect you to implement recursion by hand. We expect you to use a stack to emulate the recursive call. Please do not use other non-recursive solutions that you may find on the internet. (10 Marks)</p> <p>Rubrics:</p> <table><tr><td>Conversion</td><td>(3 Marks)</td></tr><tr><td>Data structure</td><td>(1 Marks)</td></tr><tr><td>Using Recursion</td><td>(3 Marks)</td></tr><tr><td>Without using Recursion</td><td>(3 Marks)</td></tr></table>	Conversion	(3 Marks)	Data structure	(1 Marks)	Using Recursion	(3 Marks)	Without using Recursion	(3 Marks)	CO2/K6
Conversion	(3 Marks)									
Data structure	(1 Marks)									
Using Recursion	(3 Marks)									
Without using Recursion	(3 Marks)									
3	<p>(i) When is Binary Search preferred over Linear Search? (ii)Explain the working of binary search for the elements 10,14,19,26,27,31,33,35,42,44 for searching the element 25. Explain with an algorithm. (10 Marks)</p> <p>Rubrics:</p> <table><tr><td>Preferred</td><td>(2 Marks)</td></tr><tr><td>Search Element</td><td>(4 Marks)</td></tr><tr><td>Algorithm</td><td>(4 Marks)</td></tr></table>	Preferred	(2 Marks)	Search Element	(4 Marks)	Algorithm	(4 Marks)	CO2/K5/ K2		
Preferred	(2 Marks)									
Search Element	(4 Marks)									
Algorithm	(4 Marks)									

