Data Structures and Algorithms

What are Algorithms ? What are Data Structures?

Week-1

Why study data structures and algorithms?

- People learn by experience.
- Optimize the code to take less or sometime feasible time.

Pseudocode:

➔ In computer science, pseudocode is an informal high-level description of the operating principle of a computer program.

➔ They are partial English and partial code.

➔ They do not follow syntax.

➔ They are Programming Language independent.

ADT

➔ Abstract data types, commonly abbreviated ADTs, are a way of classifying data structures based on how they are used and the behaviors they provide.

➔ They do not specify how the data structure must be implemented but simply provide a minimal expected interface and set of behaviors.

➔ Data Structure is a concrete implementation of a data type. It's possible to analyze the time and memory complexity of a Data Structure but not from a data type. The Data Structure can be implemented in several ways and its implementation may vary from language to language.

Linear Data Structures:

➜ A Linear data structure have data elements arranged in sequential manner and each member element is connected to its previous and next element. This connection helps to traverse a linear data structure in a single level and in single run. Such data structures are easy to implement as computer memory is also sequential. Examples of linear data structures are List, Queue, Stack, Array etc.

Arrays-ADT

Array is a container which can hold a fix number of items and these items should be of the same type.They are stored in contiguous memory locations so that accessing any element randomly happens in constant time.Following are the important terms to understand the concept of Array.

Element − Each item stored in an array is called an element.

Index − Each location of an element in an array has a numerical index, which is used to identify the element.

Basic Operations of Array

➔ Traverse() – accessing all the array elements one by one.
➔ Insertion() – Adds an element at the given index.
➔ Deletion() – Deletes an element at the given index.
➔ Search() – Searches an element using the given index or by the value.
➔ Update() – Updates an element at the given index.

Dynamic Arrays:

Arrays are usually created with a defined size and you can't change it after its creation.

Dynamic arrays are implemented in a way they are capable of resizing and holding more elements as needed.

Higher-level languages like Python, Golang, Java and, Javascript implement dynamic arrays.

Subarray with given Sum

Given an unsorted array **A** of size **N** that contains only non-negative integers, find a continuous sub-array which adds to a given number **S**.

Example 1:

```
Input:
N = 5, S = 12
A[] = {1,2,3,7,5}
Output: 2 4
Explanation: The sum of elements
from 2nd position to 4th position
is 12.
```

Example 2:

```
Input:
N = 10, S = 15
A[] = {1,2,3,4,5,6,7,8,9,10}
Output: 1 5
Explanation: The sum of elements
from 1st position to 5th position
is 15.
```