

Unit 1

Introduction

- **1 Introduction to data structure (8)**
- 1.1 Linear & Non linear
- 1.2 Algorithm Basic Concepts
- 1.3 Time and Space complexity of algorithms, Big O Notation and theta
- notations
- 1.4 Definition, implementation and notation of Array
- 1.5 Basic operation such as addition, deletion

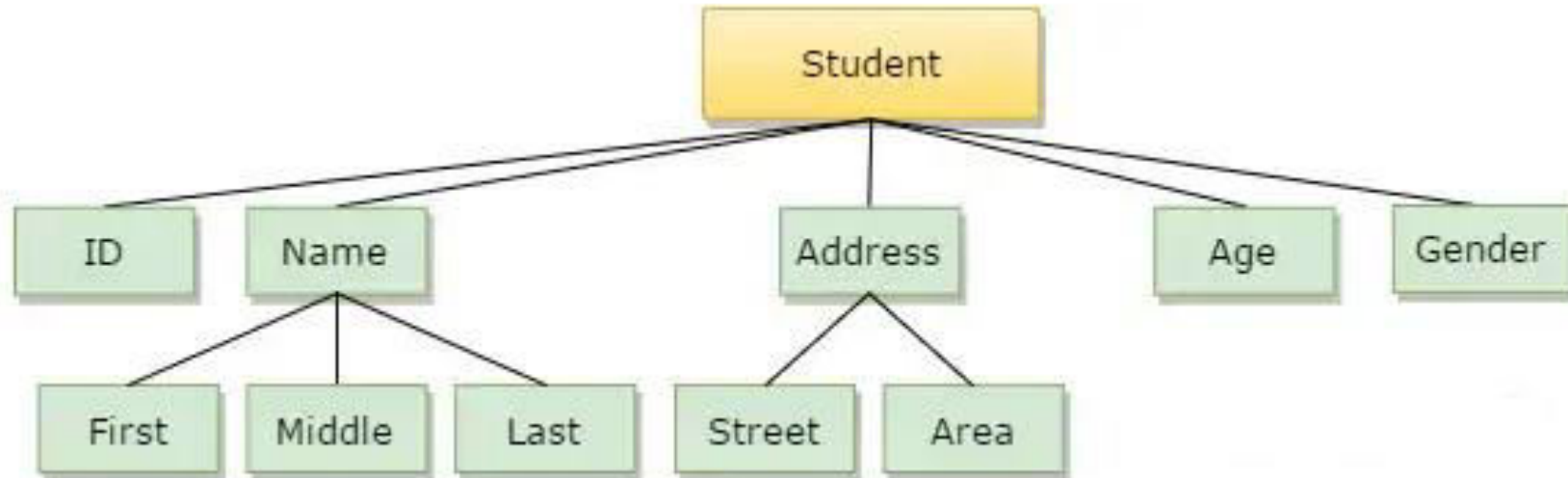
What is Data Structure?

In the context of computers, the data structure is a specific way of storing and organizing data in the computer's memory so that these data can be easily retrieved and efficiently used when needed later. The data can be managed in many different ways, such as a logical or mathematical model for a particular organization of data is called a data structure.

The variety of a specific data model depends on the two factors:

- First, it must be loaded enough into the structure to reflect the actual relationship of the data with a real-world object.
- Second, the formation should be so simple that one can efficiently process the data whenever necessary.

Let's take an example where a student's name can be broken down into three sub-items: first, middle, and last. But an ID assigned to a student will usually be considered a single item.



The example mentioned above, such as ID, Age, Gender, First, Middle, Last, Street, Area, etc., are elementary data items, whereas the Name and the Address are group data items.

Categories of Data Structure

Data structures can be subdivided into two major types:

- Linear Data Structure
- Non-linear Data Structure

Linear Data Structure

A data structure is said to be linear if its elements combine to form any specific order. There are two techniques for representing such linear structure within memory.

- The first way is to provide a linear relationship between all the elements represented using a linear memory location. These linear structures are called arrays.
- The second technique provides a linear relationship between all the elements represented using the concept of pointers or links. These linear structures are called linked lists.

The typical examples of the linear data structure are:

- Arrays
- Queues
- Stacks
- Linked lists

Non-linear Data Structure

This structure mainly represents data with a hierarchical relationship between different elements.

Examples of Non-Linear Data Structures are listed below:

- Graphs
- Family of trees and
- Table of contents

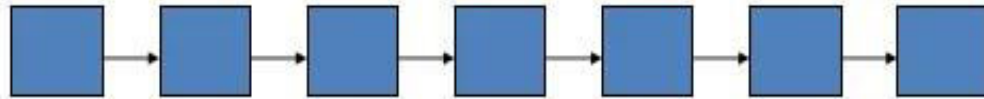
Tree: In this case, the data often has a hierarchical relationship between the different elements. The data structure that represents this relationship is called a rooted tree graph or tree.

Graph: In this case, the data sometimes has relationships between pairs of elements, which do not necessarily follow a hierarchical structure. Such a data structure is called a graph.

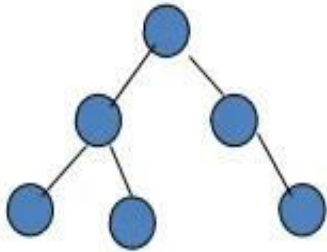
Types of data structures



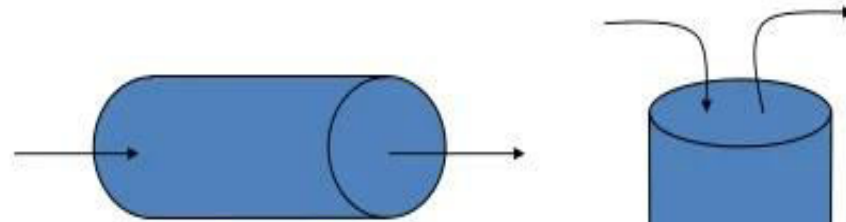
Array



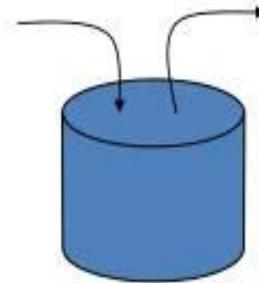
Linked List



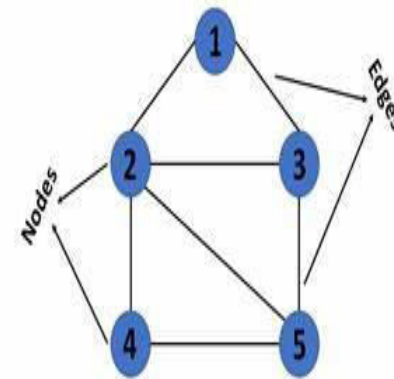
Tree



Queue



Stack



There are many, but we named a few. We'll learn these data structures in great detail!

	Linear Data structure	Non-linear data structure
1	The data elements are stored in a linear order in the case of linear data structure. Each and every element is connected to the first and the next element in the sequence.	The data elements in the case of a non-linear data structure are arranged in a non-linear way and attached hierarchically. The data elements are attached to multiple elements.
2	The structure of the data consists of a single level. There is no hierarchy in the linear data structure.	In this structure, there are multiple levels involved in the structure. Therefore the elements are arranged hierarchically.
3	The implementation of the linear structure of data is easy as the elements are stored in a linear way.	The implementation of the structure is a complex process compared to the linear structure.
4	Traversal of the elements in a linear data structure can be carried out in a single execution because the data is present in a single level	Traversal of the elements cannot be carried out in a single execution only. Multiple runs are required for traversing the data in a non-linear data structure.
5	There is no efficient utilization of memory in a linear data structure.	There is efficient utilization of memory in a non-linear data structure.
6	Examples of linear data structures include array, stack, queues, and linked list.	Examples of non-linear data include trees and graphs
7	The linear structure of data is applied mainly in software development.	The non-linear structure of data is mostly applied in Artificial intelligence and image processing.
8	With the increase in the size of the input, the time complexity increases.	Even if there is an increase in the size of the input, the time complexity remains the same.
9	Only one type of relationship might be present between the data elements	A one-to-one or one-to-many type of relationship can exist between the elements in a non-linear type of data structure.