

# DATA STRUCTURES

## INTRODUCTION

What is a data structure?

- Way of organising data in memory

Primitive vs non-primitive:

- Primitive = data structures that only hold a single value (int, char, float, double, etc.)
- Non-primitive: linear and non-linear

Linear = data arranged sequentially, one element is connected to only one other element (array, linked list, stack, queue)

Non-linear = When one element is connected to n elements (trees and graphs)

Also classified as:

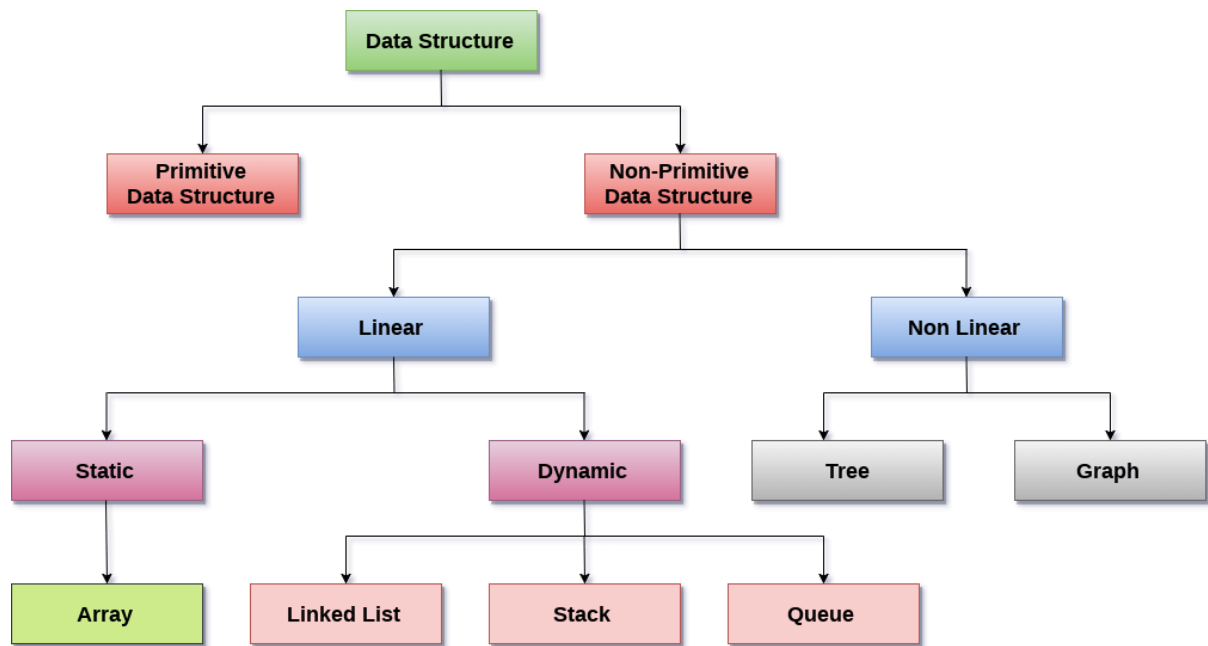
- Static (fixed maximum size, size allocated at compile time)
- Dynamic (flexible maximum size, allocated at runtime)

Major operations:

- Searching (search for element)
- Sorting (ascending or descending order)
- Insertion (insert new element)
- Updation (update an element)
- Deletion (delete an element)

Advantages of data structure:

- Efficiency: makes program very efficient in terms of time and space
- Reusability: multiple client programs can use the data structure
- Abstraction: client cannot see internal working so does not need to worry about implementation



### Linear data structures:

- Arrays = collection of similar data items. Each item is an element. All elements share same variable name but different index. Can be one dimensional, two dimensional or multidimensional.
- Linked List = collection of nodes stored at non-contiguous memory locations. Each node contains a pointer to adjacent node.
- Stack = Linear List where insertion and deletion allowed at one end, the top.
- Queue = Linear List where elements can be inserted only at the rear and deleted only at the front (FIFO, first-in-first-out).

### Non-linear data structures:

- Trees = Multilevel with a hierarchical relationship between nodes. Bottom nodes called leaf nodes and top is called root node. Each node contains pointers to adjacent nodes. Based on parent-child relationship between nodes.
- Graphs = Pictorial representation of set of elements connected by links known as edges.

### Operations:

- Traversing = Visiting each element to perform operation such as searching or sorting.
- Insertion = Add elements to location in data structure
- Deletion = Removing element from any location
- Searching = Finding location of an element in data structure. Linear vs binary search.
- Sorting = Arrange data structure into specific order. Many sorting algorithms.
- Merging = Produce a third list from two lists of similar element types by joining them together.