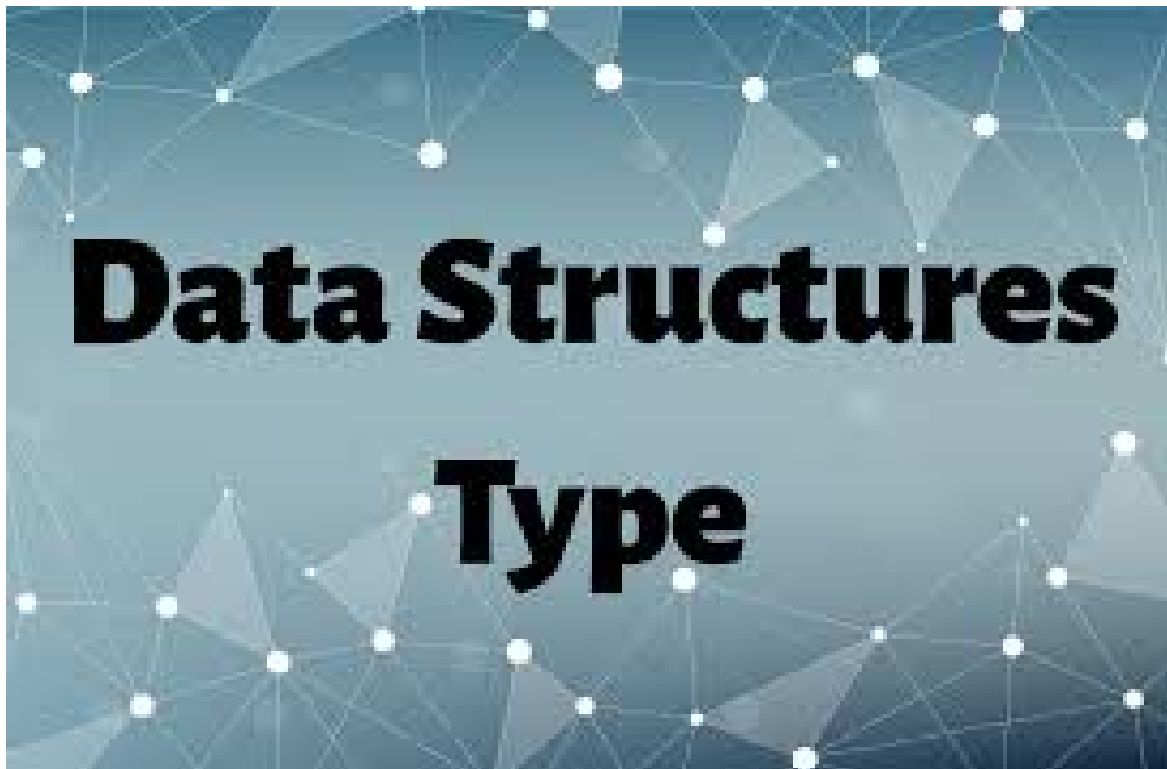# RAPPORT Data Structures

Letaief Mohamed ali

What is Data Structure:

A data structure is a storage that is used to store and organize data. It is a way of arranging data on a computer so that it can be accessed and updated efficiently.

How Data Structure varies from Data Type:

data type	data stucture
The data type is the form of a variable to which a value can be assigned. It defines that the particular variable will assign the values of the given data type only.	Data structure is a collection of different kinds of data. That entire data can be represented using an object and can be used throughout the program.
It can hold value but not data. Therefore, it is dataless.	It can hold multiple types of data within a single object.
The implementation of a data type is known as abstract implementation.	Data structure implementation is known as concrete implementation.
There is no time complexity in the case of data types.	In data structure objects, time complexity plays an important role

In the case of data types, the value of data is not stored because it only represents the type of data that can be stored.	While in the case of data structures, the data and its value acquire the space in the computer's main memory. Also, a data structure can hold different kinds and types of data within one single object.
Data type examples are int, float, double, etc.	Data structure examples are stack, queue, tree, etc

Classification of Data Structure:

Data structure has many different uses in our daily life. There are many different data structures that are used to solve different mathematical and logical problems. By using data structure, one can organize and process a very large amount of data in a relatively short period. Let's look at different data structures that are used in different situations.
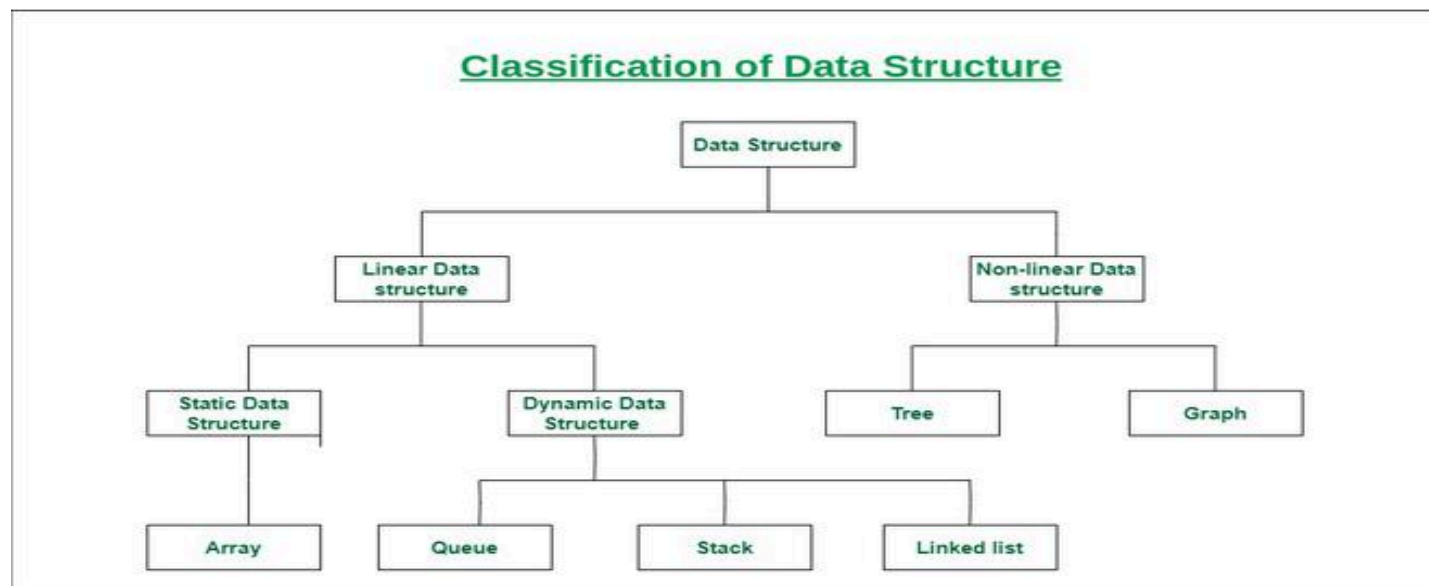
- **Linear data structure:** Data structure in which data elements are arranged sequentially or linearly, where each element is attached to its previous and next adjacent elements, is called a linear data structure.

  Examples of linear data structures are array, stack, queue, linked list, etc.

    - **Static data structure:** Static data structure has a fixed memory size. It is easier to access the elements in a static data structure.

      An example of this data structure is an array.

    - **Dynamic data structure:** In the dynamic data structure, the size is not fixed. It can be randomly updated during the runtime which may be considered efficient concerning the memory (space) complexity of the code.

      Examples of this data structure are queue, stack, etc.

- **Non-linear data structure:** Data structures where data elements are not placed sequentially or linearly are called non-linear data structures. In a non-linear data structure, we can't traverse all the elements in a single run only.

  Examples of non-linear data structures are trees and graphs.

Need Of Data structure :

The structure of the data and the synthesis of the algorithm are relative to each other. Data presentation must be easy to understand so the

developer, as well as the user, can make an efficient implementation of the operation.

Data structures provide an easy way of organizing, retrieving, managing, and storing data.

Here is a list of the needs for data.

1. Data structure modification is easy.

2. It requires less time.

3. Save storage memory space.

4. Data representation is easy.

5. Easy access to the large database.

Arrays:

An array is a linear data structure and it is a collection of items stored at contiguous memory locations. The idea is to store multiple items of the same type together in one place. It allows the processing of a large amount of data in a relatively short period. The first element of the array is indexed by a subscript of 0. There are different operations possible in an array, like Searching, Sorting, Inserting, Traversing, Reversing, and Deleting.

Memory Location

200	201	202	203	204	205	206			
U	B	F	D	A	E	C	▪	▪	▪
0	1	2	3	4	5	6			

Linked list:

A linked list is a linear data structure in which elements are not stored at contiguous memory locations. The elements in a linked list are linked using pointers as shown in the below image:

Types of linked lists:

- Singly-linked list
- Doubly linked list
- Circular linked list
- Doubly circular linked list

Linked List
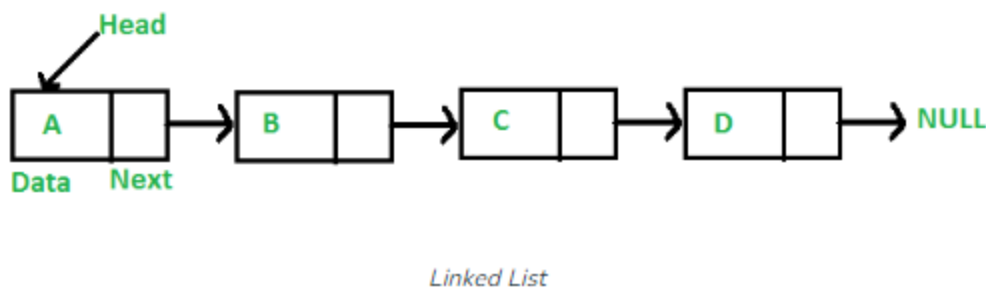
here are some data structures and sorting algorithms that i'm not as familiar with:

- steiner Tree.
- Network Flow.
- SS Shortest Path.

h