01418231
Data Structures

BASIC DATA STRUCTURES

Powered by
Dr. Jirawan Charoensuk

Agenda

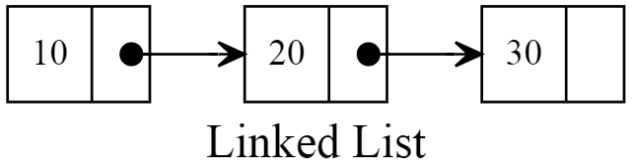
- What is Data structure?
- Data Types
  - Atomic & Composite/Structured
- Data structure?
  - Primitive Data Structures
  - Non-Primitive Data Structures
- Common operations of data structures
- Advantages of Data Structure

What is Data structure?

What is data Structure ?

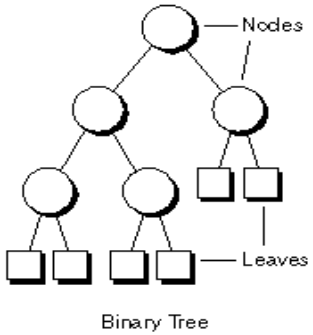
- Definition (http://en.wikipedia.org/)
  - Is a way of storing data in a computer so that it can be used efficiently.
  - Is an organization of mathematical and logical concepts of data
    - Mathematical = +, -, *, /
    - Logical = >, <, >=, <=, &, ||
  - carefully chosen data structure will allow the most efficient algorithm to be used

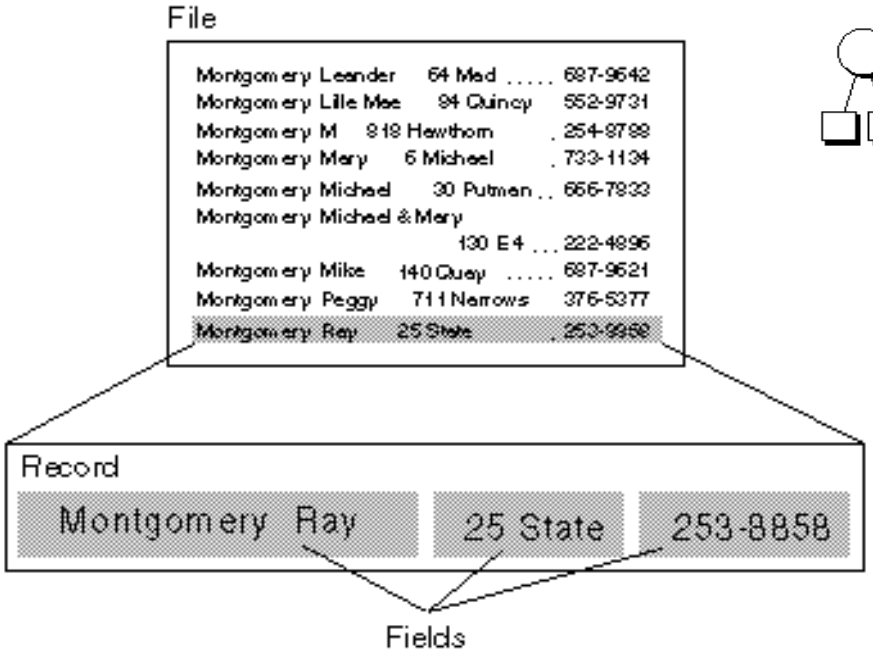
What is Data structure?



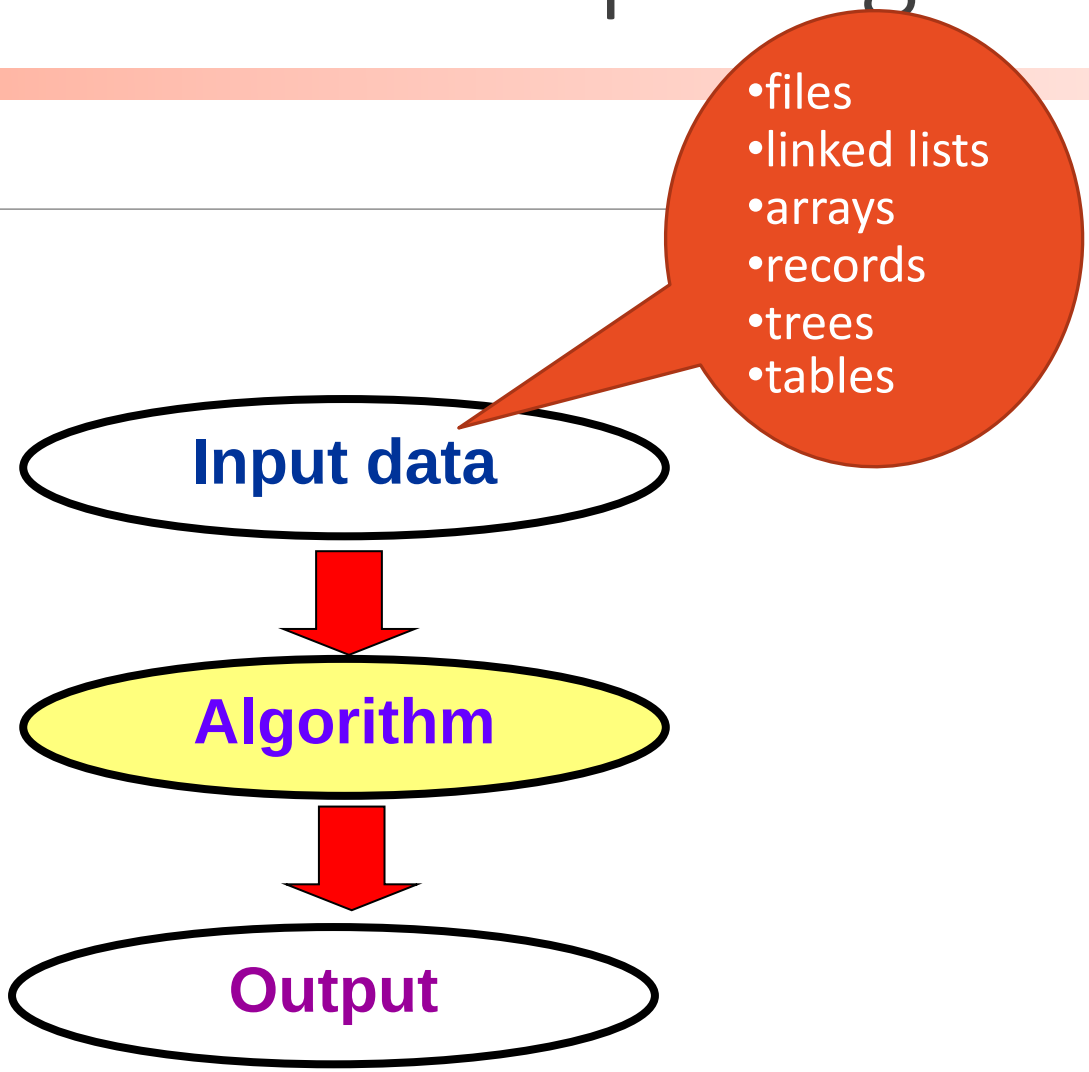
Definition

- refers to a scheme for organizing related pieces of information
- The basic types of data structures include:
  - files
  - linked lists
  - arrays
  - records/structure
  - trees
  - tables





Algorithms in Computing



Example of Programming languages

```
include<stdio.h>
main()
{
    printf(“Hello Word”);
}
```

```
// Hello.java
public class Hello
 {
       public static void main(String[] args)
        {
          System.out.println("Hello, world!");
        }
 }
```

```
<?php
echo 'Hello, World!';
?>
```

Data Types

Data Types

A description of the set of values

The basic set of operations that can be applied to values of the type

Type	Values	Operations
integer	-∞, ... , -2, -1, 0, 1, 2,..., ∞	*, +, -, %, /, ++, --, ...
floating point	-∞, ... , 0.0, ..., ∞	*, +, -, /, ...
character	\0, ..., 'A', 'B', ..., 'a', 'b', ..., ~	<, >, ...

Data Type

1. ATOMIC DATA TYPE
   - The single element of data type
   - Each element have specifically properties
   - All programming language have atomic data type
   - Example
     - Integer / Floating point
     - Character
     - Boolean

2. COMPOSITE/STRUCTURED
   - A data type whose elements are composed of multiple data items.
   - Example
     - Array
     - String
     - Record

Data Type

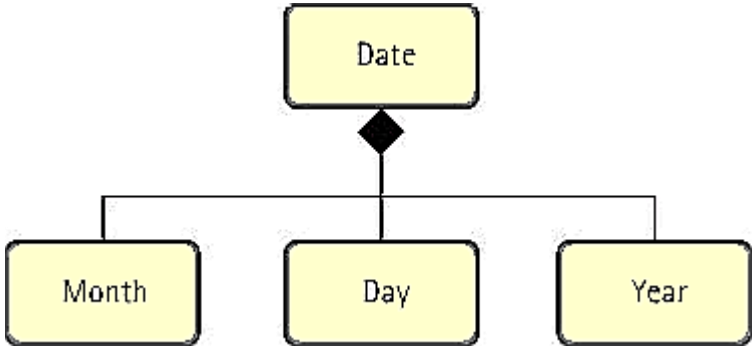
1. ATOMIC DATA TYPE
   - The single element of data type
   - Each element have specifically properties
   - All programming language have atomic data type
   - Example
     - Integer /Floating point
     - Character
     - Boolean

2. COMPOSITE/STRUCTURED
   - A data type whose elements are composed of multiple data items.
   - Example
     - Array
     - String
     - Record

Composite Data Types

Ex-> calendar date :

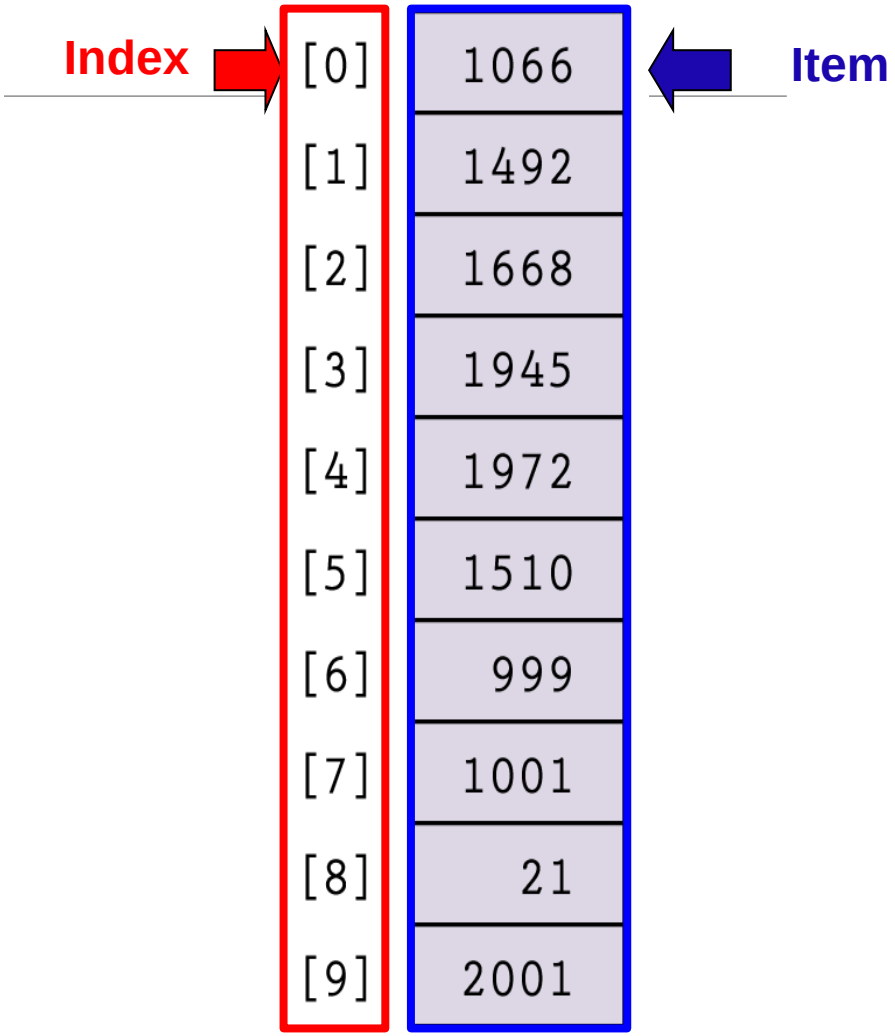
- composed of a month value, a day value, and a year value



Composite Data Types

Array

- is a named collection of homogeneous items
  - Individual items = accessed by their place within the collection *(Values in array)*
  - Index =the place within the collection



Composite Data Types

Records or Structure

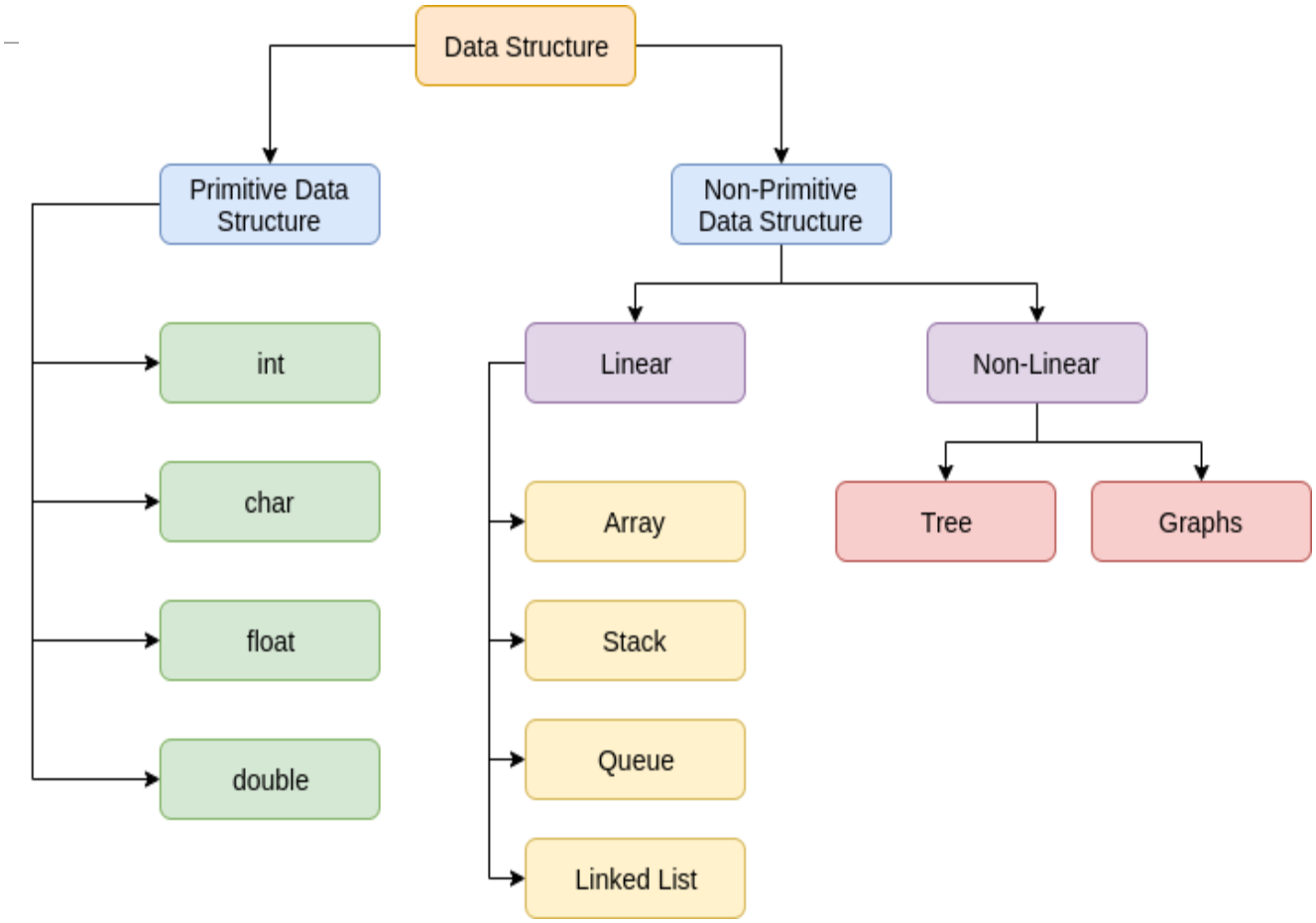
- A record is a named heterogeneous collection of items in which individual items are accessed by name
- The elements in the collection can be of various types
  - Integer
  - Floating Point
  - Characters
  - Boolean values
  - Array

Data structures

Data structures

Data structures are primarily categorized into two parts:

1. Primitive Data Structures
2. Non-Primitive Data Structures



https://afteracademy.com/blog/introduction-to-data-structure

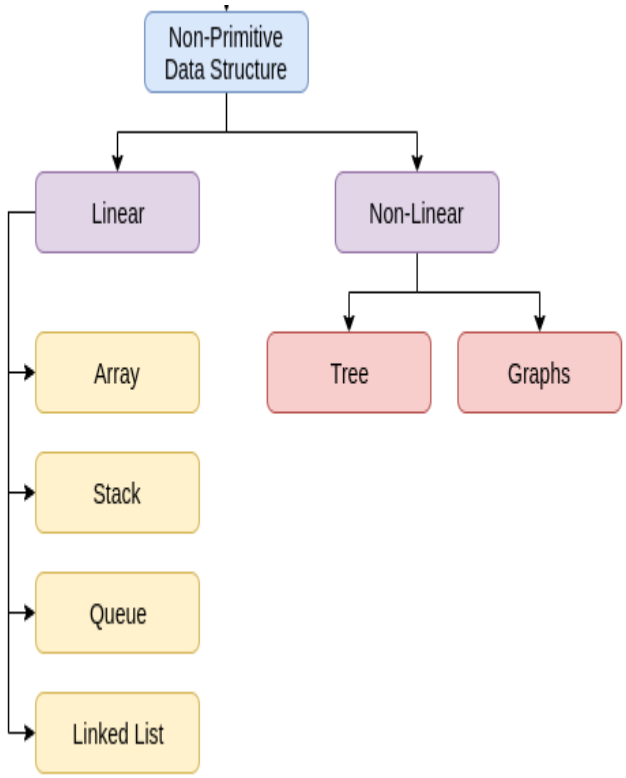
Data structures

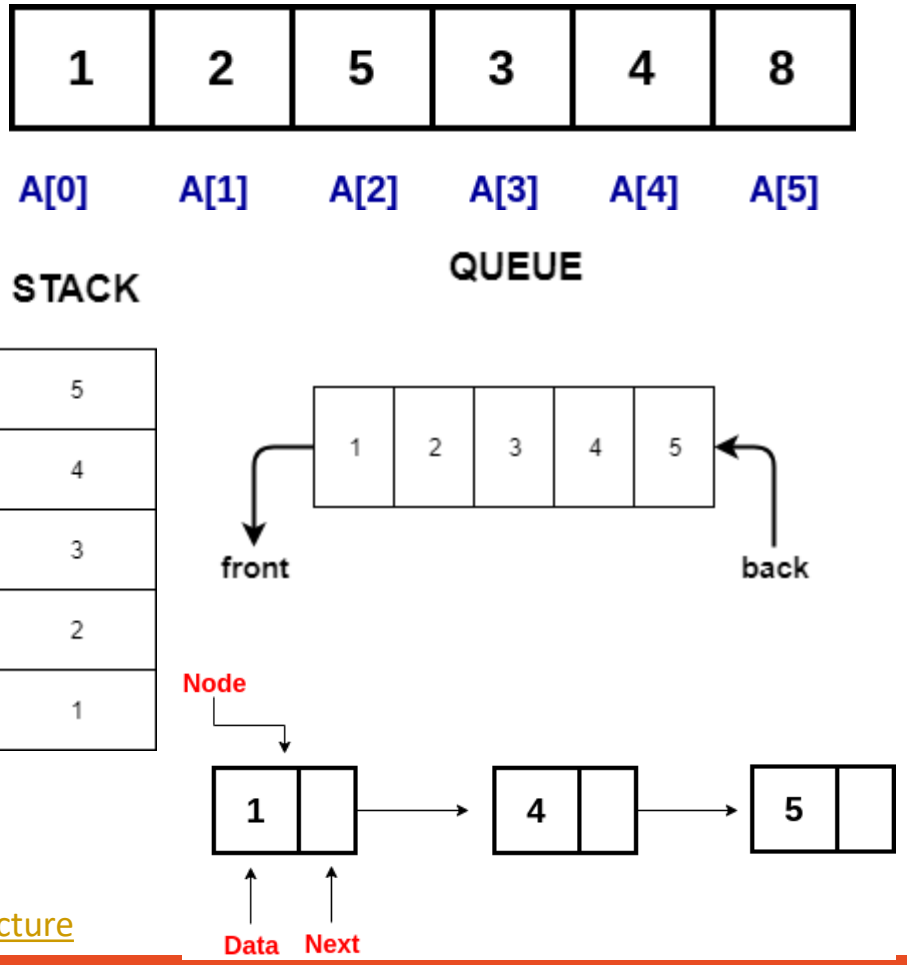
1. Primitive Data Structures
   - These are the predefined way of storing data in the system. All sets of operations are pre-defined.
     - char, int, float, double
2. Non-Primitive Data Structures
   - The data structures which are designed using primitive data structures are called non-primitive data structures.
   - They are used to store a collection of data.
   - It can be categorized into two parts:
     - Linear data structure and Non-Linear data structure

https://afteracademy.com/blog/introduction-to-data-structure

Data structures

2. Non-Primitive Data Structures

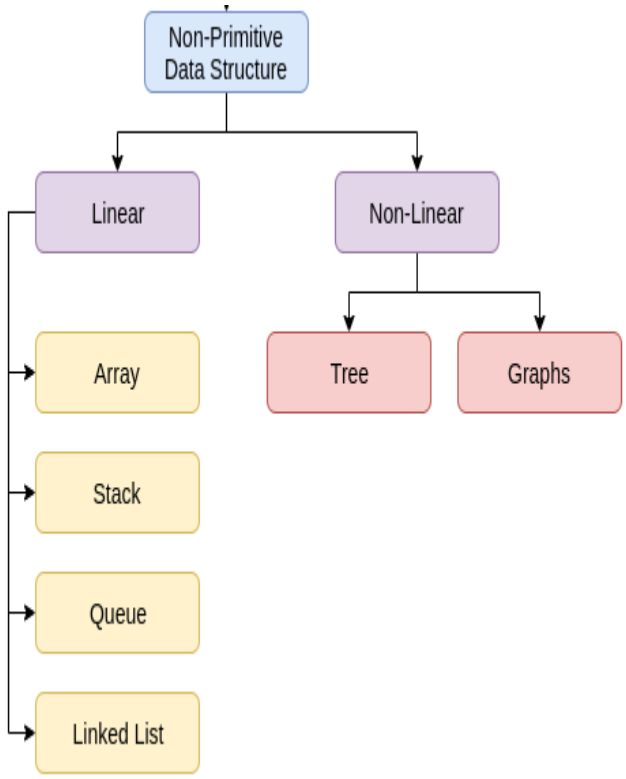


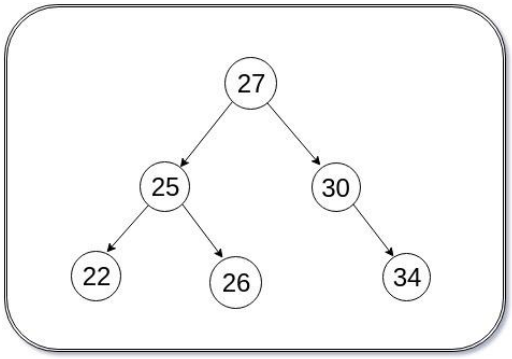


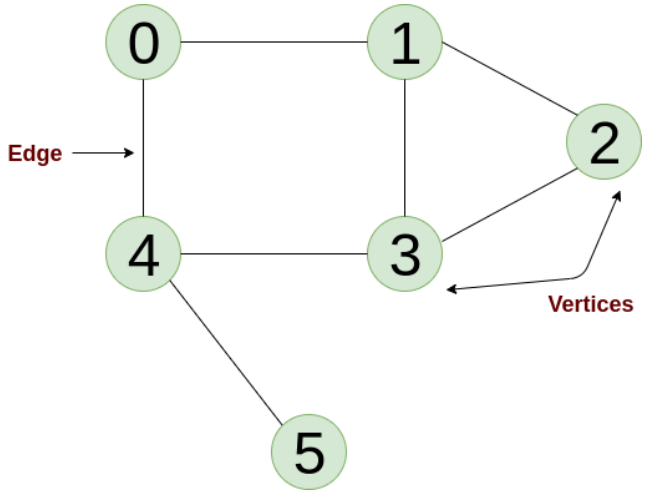
https://afteracademy.com/blog/introduction-to-data-structure

Data structures

2. Non-Primitive Data Structures



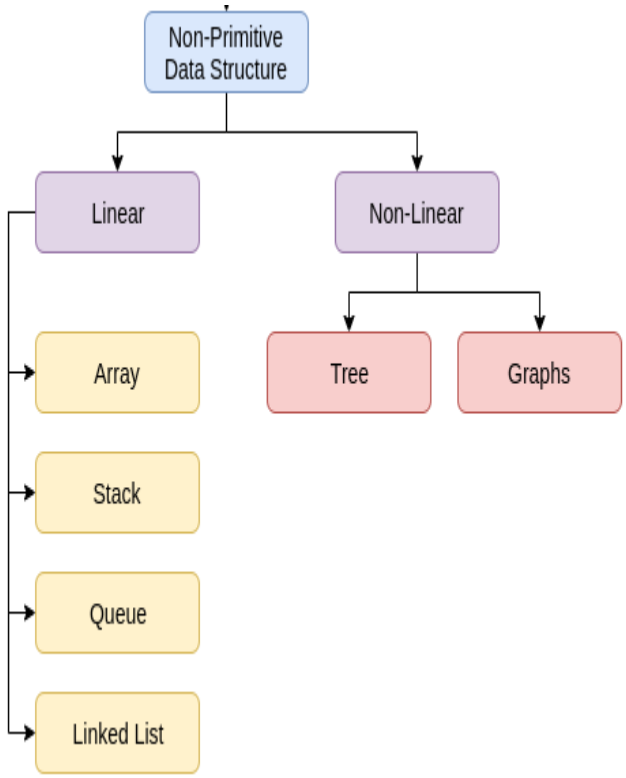


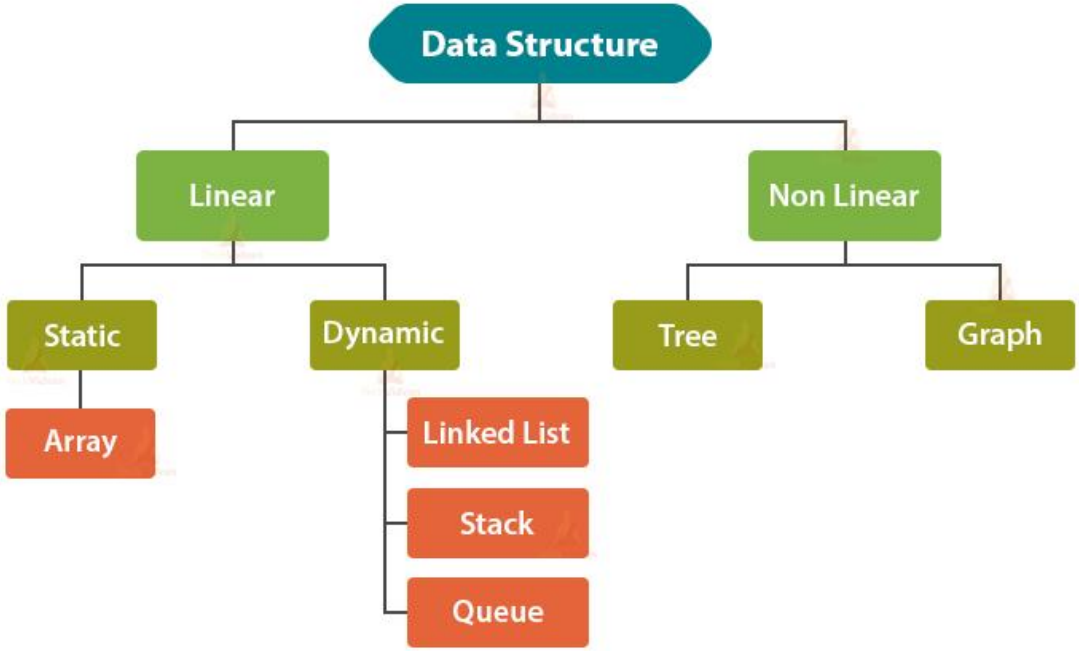


https://afteracademy.com/blog/introduction-to-data-structure

Data structures

2. Non-Primitive Data Structures





https://afteracademy.com/blog/introduction-to-data-structure https://techvidvan.com/tutorials/data-structure-in-java/

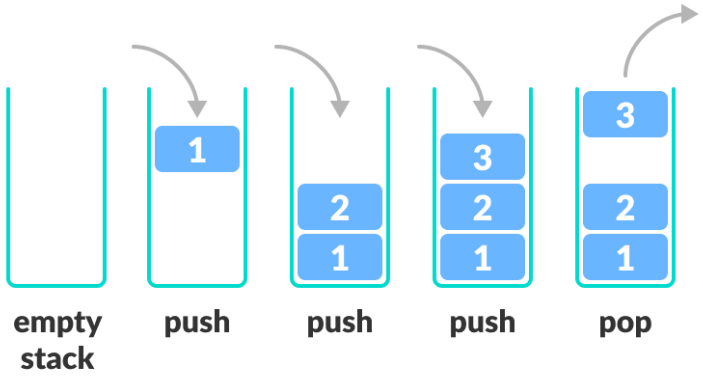
Common operations of data structures

1. Searching:
   - performed to search for a particular element or a key .
2. Sorting: .
   - involves arranging the elements in a data structure in a particular order either ascending or descending
3. Insertion:
   - deals with adding an element to the data structure
4. Deletion:
   - removes an element from the data structure.
5. Traversing:
   - traverse a data structure when we visit each and every element in the structure

https://www.softwaretestinghelp.com/data-structures-in-cpp/

Advantages of Data Structure

- **Abstraction**: Data structures are often implemented as abstract data types. The users only access its outer interface without worrying about the underlying implementation. Thus data structure provides a layer of abstraction.
- **Efficiency:** Proper organization of data results in efficient access of data thereby making programs more efficient. Secondly, we can select the proper data structure depending on our requirements.
- **Reusability:** We can reuse the data structures that we have designed. They can be compiled into a library as well and distributed to the client.

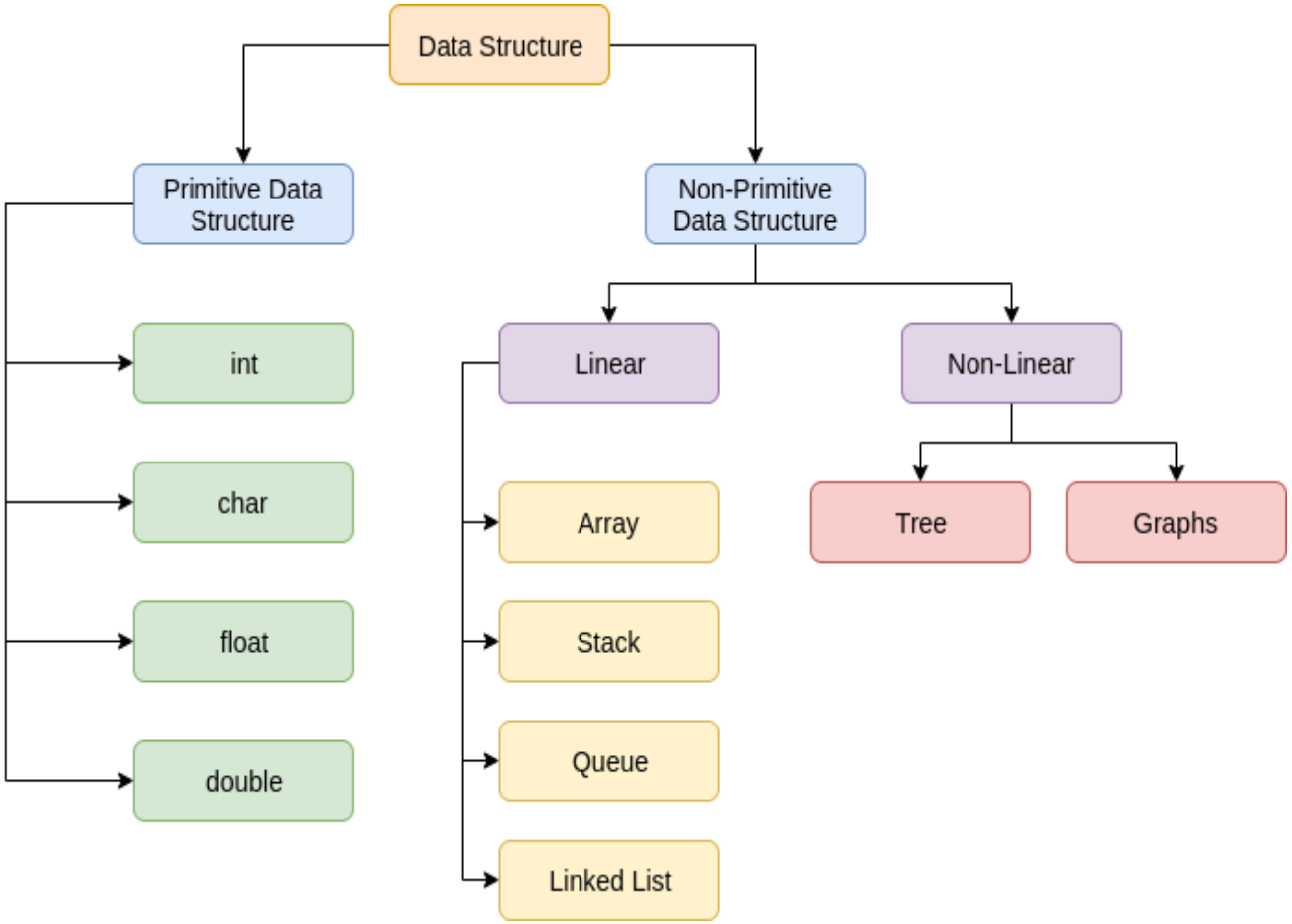


https://www.softwaretestinghelp.com/data-structures-in-cpp/

https://www.programiz.com/dsa/stack

Summary

- Introduce this tutorial on introduction to data structures



https://afteracademy.com/blog/introduction-to-data-structure

Question