

DATA STRUCTURES NOTES

@curious-programmer

What is a Data structure ?

Data can be arranged in a many ways, logical or mathematical arrangement of a data is called as Data Structure.

Examples : Array, linked list, stack, queue, Tree Graph, and many more.

What is an Algorithms ?

Sequence of steps performed on the data using efficient data structures to solve a given problem.

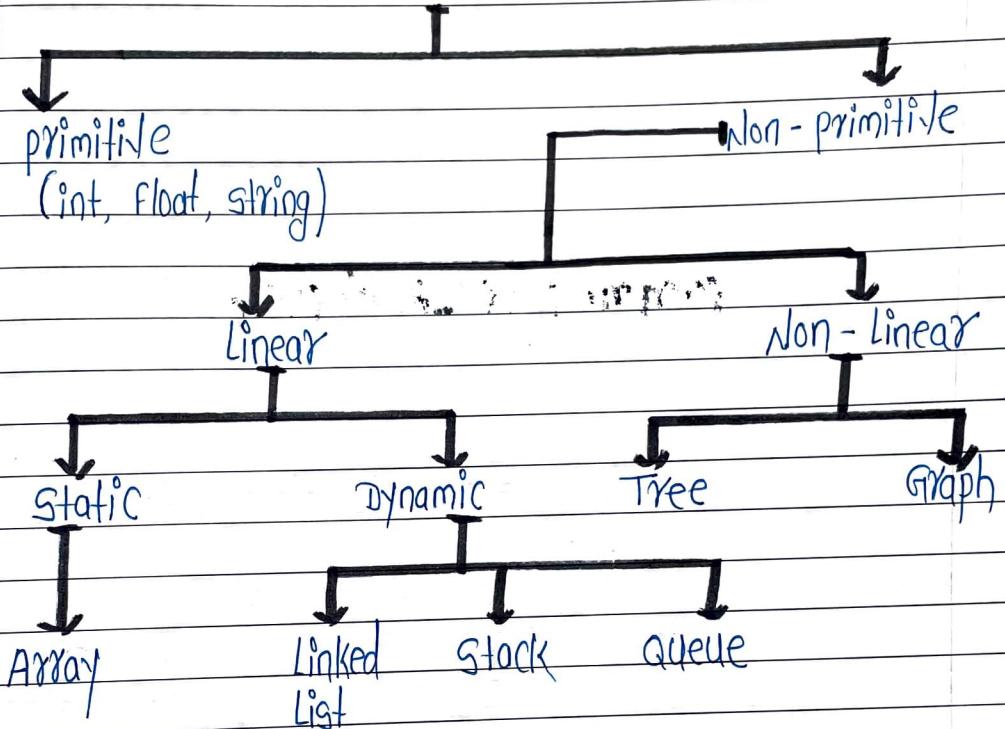
Example : Sorting an Array .

(link in Bio)

Classification of Data structures

@curious-programmer

Data structure.



Types of Data structures

- (a) primitive and Non-primitive Data structure
- (b) static and dynamic Data structure
- (c) persistent and ephemeral Data structure.

Non-primitive further divided
into two types

- i) Linear Data structure
- ii) Non-Linear Data structure

persistent further
divided into three
types

- i) partially persistent
- ii) fully persistent
- iii) confluently persistent

(Link in Bio)

Data structure operations :

The following four operations play a major role

(1) Traversing :

Accessing each record exactly once so that certain items in the record may be processed

(2) searching :

Finding the location of the record with a given key value.

(3) Inserting :

Adding a new record to the structure

(4) Deleting :

Removing a record from the structure

(5) Merging :

Combining the records in two different sorted files into a single sorted file

(6) Sorting :

Arranging the record in some logical order
e.g. Alphabetically according to some ~~order~~ key or
in numerical order according to ^{Name} some number
~~key~~

@curious_programmer

Searching Algorithms :

A search algorithm is a step-by-step procedure used to locate specific data among a collection of data.

Types of search algorithms with the complexity

1) Linear Search :

A linear search or sequential search is a method for finding an element within a list. It sequentially checks each element of the list until a match is found or the whole list has been searched.

$$C(n) = n/2$$

← complexity of linear search

@curious_programmer

2) Binary search :

In binary search approach the element is always searched in the middle of a portion of an array. Binary search can be implemented only on a sorted list of items.

If the elements are not sorted already, we need to sort them first.

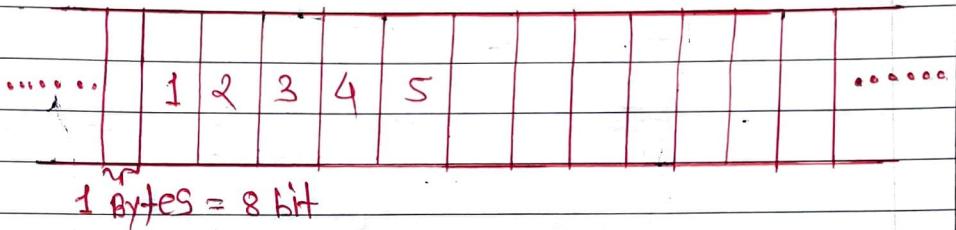
$$C(n) = \log_2 n$$

← complexity of
Binary search

ARRAY :-

OR Array is a Type of linear Data structure
Array is a collection of more than one data
but all the data items are same data types, &
stored that data in a computer in a contiguous
memory location

@curious-programmer



Memory is a long tap of bytes

Types of Array :-

① One Dimensional Array :

The array with only one subscript that array
is called as one dimensional array.

Example : `int a[s];` ← subscript

② Two Dimensional Array :

The array with two subscript that array
is called as two dimensional array.

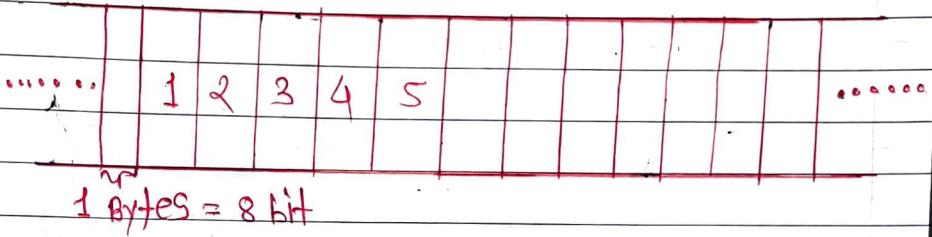
Example : `int a[s][s];` ← subscript

③ Multi-Dimensional Array : The array with more than two subscript.

ARRAY :-

OR
Array is a collection of more than one data but all the data items are same data types, & stored that data in a computer in a contiguous memory location.

@curious-programmer



Memory is a long tap of bytes

Types of Array :

① One Dimensional Array :

The array with only one subscript that array is called as one dimensional array.

Example : int a[5]; ← subscript

② Two Dimensional Array :

The array with two subscript that array is called as two dimensional array.

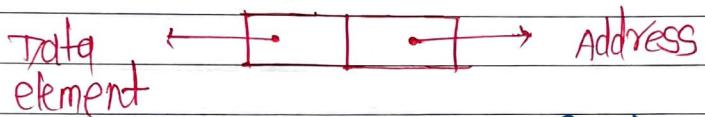
Example : int a[5][5]; ← subscript

③ Multi-Dimensional Array : The array with more than two subscript.

LINKED LIST :-

Linked list is a linear data structure. It is also a collection of more than one data items of a dissimilar data type like array but it can not stored it in contiguous memory location. It can be stored randomly in a main memory.

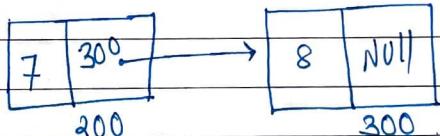
So that linked list contains two part one for **Data** and second part for the **Address** of the next data element



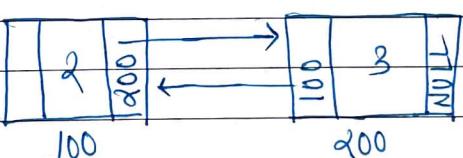
@curious..programmer

Types of linked list

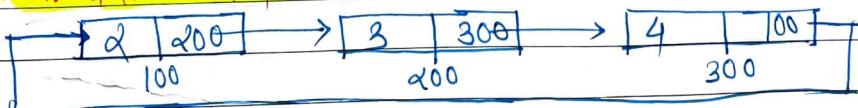
① singly linked list :



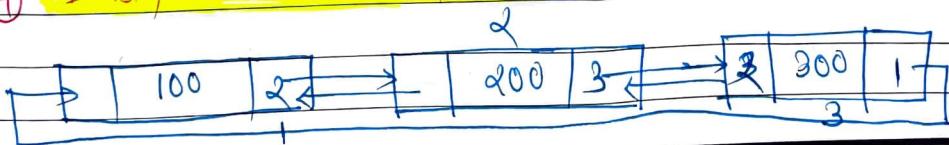
② doubly linked list :



③ circular linked list :

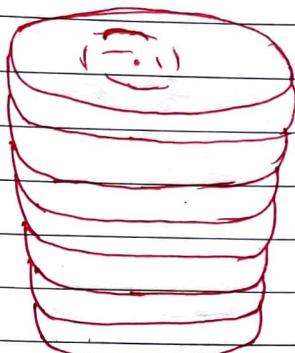


④ doubly circular linked list :



STACKS

A stack is a list of elements in which an elements may be inserted or deleted only at one end called the **TOP** of the stack.



@curious_programmer

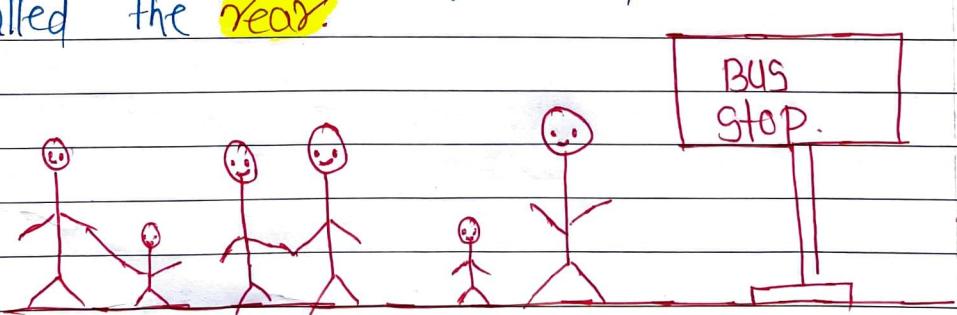
Stack of dishes

push → Insert elements into stack

Delete elements from stack ← **pop**

QUEUES

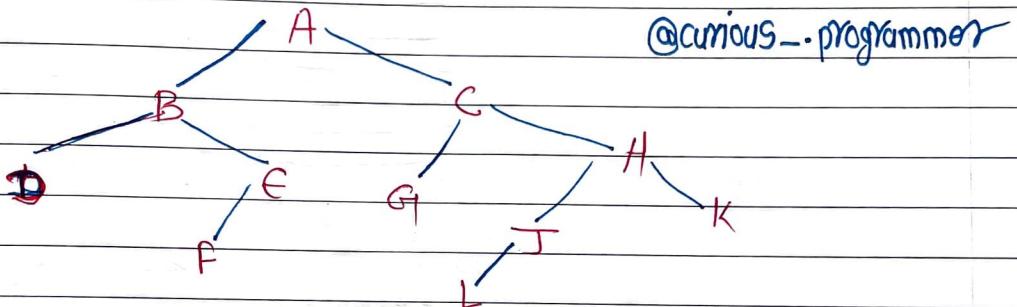
A **Queue** is a linear list of elements in which deletions can take place only at one end called **front** and insertions can take place only at the other end called the **rear**.



TREES

Trees are non-linear data structure where data are stored or data containing a hierarchical relationship between elements

A binary tree is defined as a finite set of elements called nodes.



Traversing Binary Trees

There are three ways of traversing a binary tree T with root R.

preorder

- 1) process the root R
- 2) Traverse the left subtree of R in preorder
- 3) Traverse the right subtree in preorder

Inorder

- 1) Traverse left subtree
- 2) process the root R
- 3) Traverse right subtree

postorder

- 1) Traverse left subtree
- 2) Traverse right subtree
- 3) process the root R

GRAPH

Graph is a collection of two set V and E where,

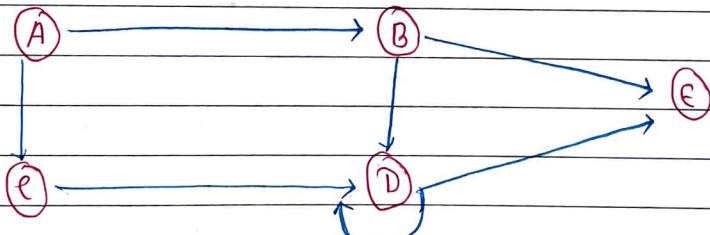
$V \rightarrow$ Vertices / Nodes.
 $E \rightarrow$ Edges

@curious_programmer

Graph is a mathematical structures that represent pair-wise relationship between objects where nodes are connected with edges.

Vertex \rightarrow vertex is nothing but the data element which is also known as Node

Edge \rightarrow Edge is a connection link between two vertices.



Representation of the graph

(A) Adjacency matrix

(B) Adjacency list