Data Structure → Data Structure is a way to organize and store the data to use efficiently.

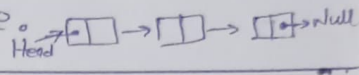
Array → An Array is a data structure which can be define as finite/fixed ordered set of homogenious elimentt.

Property → 1.) Finite 2.) Ordered 3.) Homogenious (Same type) 4.) contigous memory location 5.) Random access

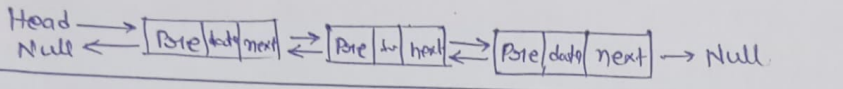
Types of Array → 1.) One dimentional 2.) multidimentional array. (It is known as array of array)

Linked List → A Linked List is linear data structure that includes a series of connected nodes, Here each node store the data and the address of the next node. (Three Types:-

Singly Linked List → It is a most common, each node has data and Pointer/address to the next node.



Doubly Linked List → We add a Pointer to the previous node in a doubly Linked List, Thus we can go in either direction forward and backward.

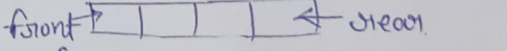


Circular Linked List → In this first node and last node also connected to each other. to form a circle. two Type:-

1. Circular Singly LL → Here, the address of the last node consists of the address of the first node.
2. Circular doubly LL → Here, in addition to the last node storing the address of the first node. The first node will also store the address of the last node.

Stack → It is a linear list where all insertion and deletions are permitted only at one end of the list. It is based on LIFO/FILO technique. (Push, Pop)

Queue → It is a linear data structure that is open at both side/ends and the operations are performed in First in First out (FIFO) order. here, insertion permits only 'rear' side and deletion permits only 'front' side.



Basic operation: — Enqueue → addition of an item.
Dequeue → Deletion of an item.

All operation: — (Enqueue, Dequeue, FrontPeek, RearPeek, IsEmpty, IsFull.)

Tree → A Tree is a set of one or more nodes. It is a non-Lenie data structure.

Binary Tree → Every node have atmost 2 children (0, 1, 2), non-linea

complete Binary Tree → when I represent (Tree in) am array from first eliment to last eliment there is no missing eliment called complete binary Tree.

Full Binary Tree → Binary Tree with maximum no of nodes is full binary Tree.

Binary Search Tree → A Binary Search Tree follows some order to arrange the elements. In a binary Search Tree The value of the left node must be Smaller than the Parent node. and the value of the right node must be greater than the Parent node. This rule is apply recursively.

BST operations → 1.) Searching O(n) 2.) Insertion O(1) 3.) Deletion O(n) 4.) Traversal.

AVL Tree → An AVL Tree is a another ballnced binary search Tree. It invented in 1962, In the AVL Tree balance factor Play major role [Bf = hl - hr] (should be -1, 0, 1). Every Sub Tree is an AVL Tree.

AVL Tree operation → 1.) Insertion O(logn) 2.) Deletion O(logn) 3.) Searching O(logn)

B Tree → B-Tree is a Self balancing tree data structure. It is a specialized m-way Tree that can be widely used for disk acces

2-3 Tree → It is a special type of B-Tree of order 3. In this each node has either 2 children or 3 children. that's why It is called 2-3 Trees.

Red-black Tree → Red-black Tree is a self balancing binary Search Tree. In which every node of the tree is either colored red or black and the color of the node is decided based on its Property —

operation → 1.) Search O(logn) 2.) Insert O(logn) 3.) Delete O(logn)

Splay Tree → Splay Tree is a self adjusting binary Search Tree in which recently accessed element will be Placed at the root of the Splay Tree.
All the operations in Splay Tree (insertion, deletion) called Splaying.

Splaying → Splaying is a process of bringing an element to its root Position by applying Suitable rotation.

Rotation → (1.) Zig (2.) Zag (3.) Zig-Zag (4.) Zag-Zig (5.) Zig-Zig (6.) Zag-Zag

Heap → Heap is a complete binary tree, in which the node can have utmost two children.

Types of Heap :- There are two types of heap.

min heap → Every Parent node having the value smaller/eavual to all its discendants, called min heap. It is complete binary Tree.

max heap → Every Parent node having the value greater than all its discendants, called max heap. It is complete binary Tree.

Heapify → Heapify is a process of creating a tree.

Priority Queue → A Priority Queue is an abstract data type that behaves similarly to the normal queue except that each element has some Priority. i.e, element with the highest Priority would come first in a Priority queue. Priority will determine the order in which element are removed from the Priority queue.

---

panning Tree → A Spanning is a Sub-graph of an undirected connected graph. In this tree is connects all the vertices of a graph with the minimum Posible number of edges.

inimum Spanning Tree → A min Spanning Tree is defined for a weighted graph. A Spanning Tree having min weight is defined as a minimum spanning Tree.

BFS → BFS is a traversal approach in which we first walk through all nodes on the same level before moving on the next level. It implement using queue.

FS → DFS is an algorithm for traversing or Searching tree or graph data structure which usage the idea of backtracking. It implement using Stack.

loyd-warshall Algorithm → is used to Solving the all Pair Shortest Path. Problem.

rd-Fulkerson Algorithm → The ford-fulkerson algorithm is widely used algorithm to solve the maximum flow problem in a flow network.

ology Sort → Topological Sort is a linear ordering of the vertices
ahn's algorithm) of the directed acyclic graph (DAG).

Hashing → It is a method for storing and retrieving the data from database in order of 1 [O(1)] time.

Hash Table → Hash table is one of the most used data structure that uses a special function known as hash function that maps a given value with a key to access the element faster.

Hash function → Any function that convert data of any size into fixed size value. Hash function (K mod 10, K mod n, mid Square, folding meth

Collision → The situation where newly inserted key map to an already occupied slot in the hash function table is called collision and it must be handle using some collision resolution technique.

Collision Resolution techniques → Two techniques -

1.) chaining (open hashing) → (Seperate chaining)
2.) open addressing (closed Hashing) → (Linear Probing, Quadratic Probing Double hashing.

Seperate chaining → is one of the technique that is used to resolve collision. It is Implement using Linked List.

Linear Probing → It is also used to resolve the collision in hash table In this approach searches are perform sequentially So it's known as linear probing.

Quadratic Probing → It is an open addressing scheme for resolving hash collision in hash table. where we look for i^2th slot in the ith iteration if the given hash value x collides in the hash table.

Rehashing / Double hashing → It is a collision Resolution technique used in hash tables. It works by using two hash function. The first hash function is used to compute the initial hash value and second hash function is used to compute the step size for the probing sequence.