DATA STRUCTURE INTERVIEW QUESTIONS.

1. **What is a data structure?**

⇒ • Data structure is a method that organizing the data in memory.

• Data structure is specialized format for organizing, processing, retrieving & storing data.

• Data structure includes Array, pointer, linked, list, stack, Queue, structure, graph, Searching, Sorting, programs etc.

• Data structure is not a programming language.

• It is set of algorithm is use in any programming language to store the data in memory.

• Types of data structures
  i). Linear data structure.
  ii). Non - Linear data structure.

ATUL KUMAR (LINKEDIN)
TELEGRAM - NOTES GALLERY.

2]. **What is Linear data structure?**

⇒ • In linear data structure elements are stored in sequentially.

• And each element are connected to previous and next element.

• Array, Linked list, stack and Queue are different type of linear data structures available.

• Linear data structure are easy to implement.

3]. **What is Non-Linear data structure?**

⇒ • In non-linear data structure elements are not arranged sequencially.

• Non-linear data structure is not easy to implement.

• It uses device memory efficiently.

• Trees and graph are examples of non-linear data structure.

4] What is array?

→ • An array is data structure for storing more than one data item that has similar data types.

• An array stored that position of each element can be computed from its index by its formula.

• In an array searching of element is easy by using index number.

5] What is Multidimentional array?

→ • A Multidimentional array is with more than one dimention.

• It is an array of array.

• 2-D array are most commonly used.

• They are used to store data in tubular manner.

6] What is linked list in data structure?

→ • It is a sequence of data structure, which are connected together via links.

• Each links contain a connection to another link.

• Linked list can grow & shrink its size, as per the requirement.

• It does not waste memory space.

    i>. Singly linkedlist

    ii>. Doubly linked list

    iii>. Circular linked List.
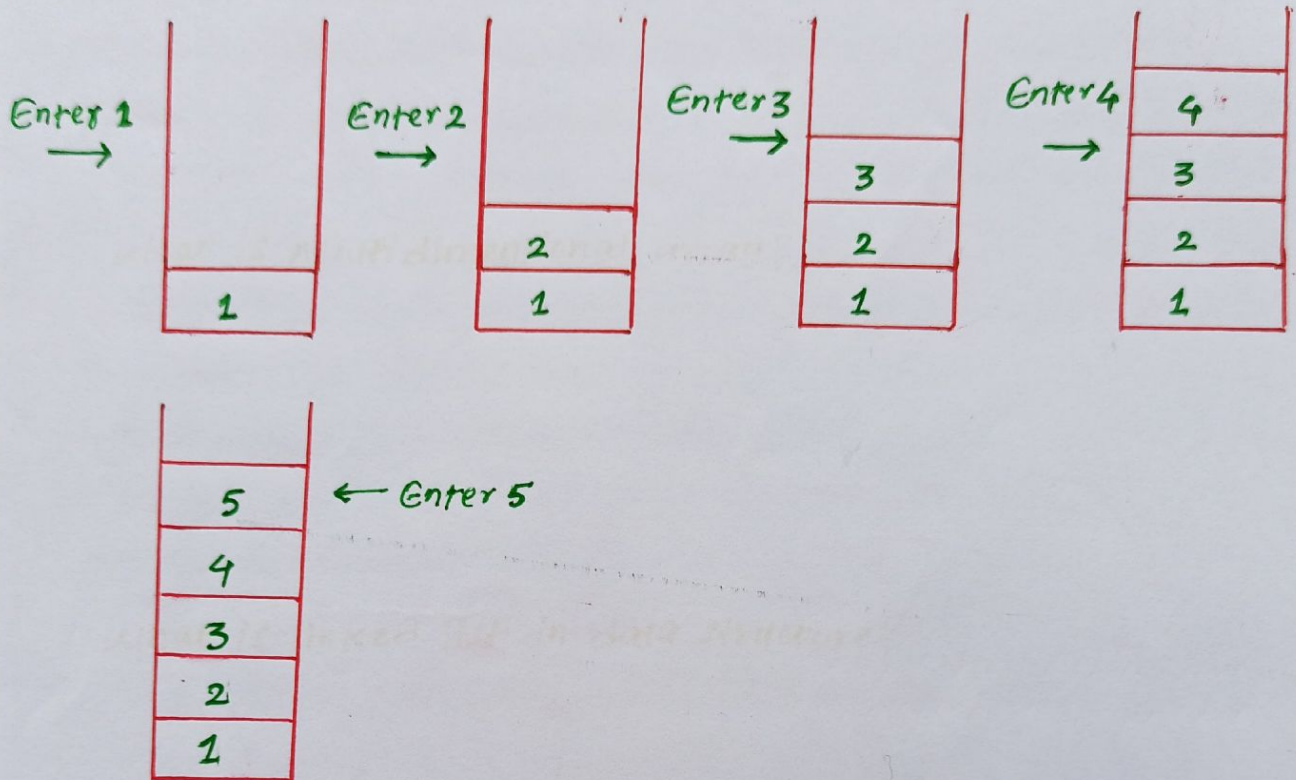
7] What is Merge sort?

→ • Merge sort divides the array into two parts then sort it and combine it.

• It takes time of (n logn) in worst case.

8]. What is stack?

⇒ • Stack is a linear data structure.
• Stack follows LIFO (Last In First Out) or FILO (First In Last Out).
• In stack elements are added from top.
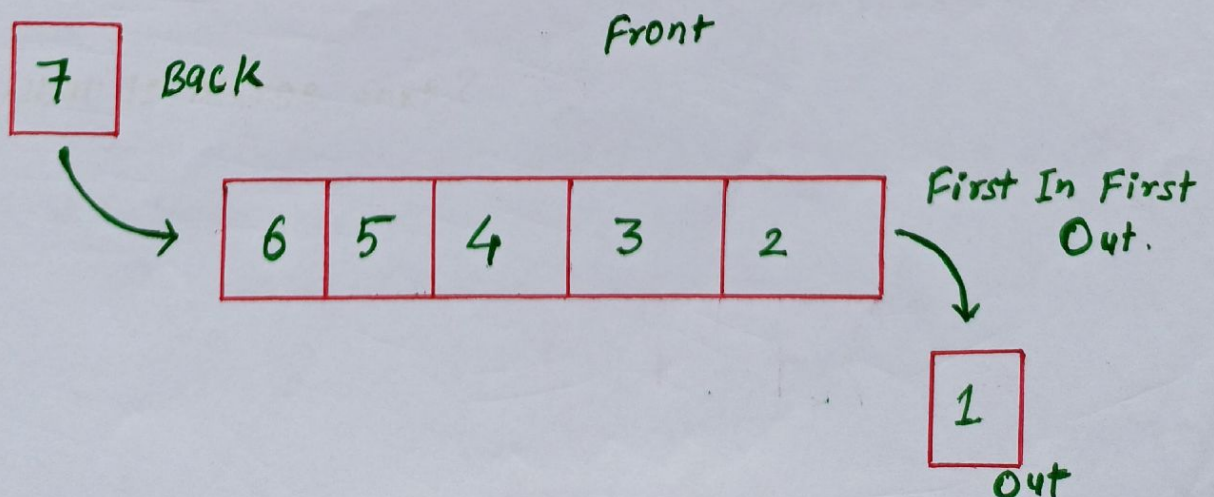• And element can deleted only from top.

```
Enter 1          Enter 2          Enter 3          Enter 4
   →                →                →      | 3 |      →      | 4 |
                         | 2 |              | 2 |             | 3 |
        | 1 |            | 1 |              | 1 |             | 2 |
                                                            | 1 |
```

```
| 5 |   ← Enter 5
| 4 |
| 3 |
| 2 |
| 1 |
```
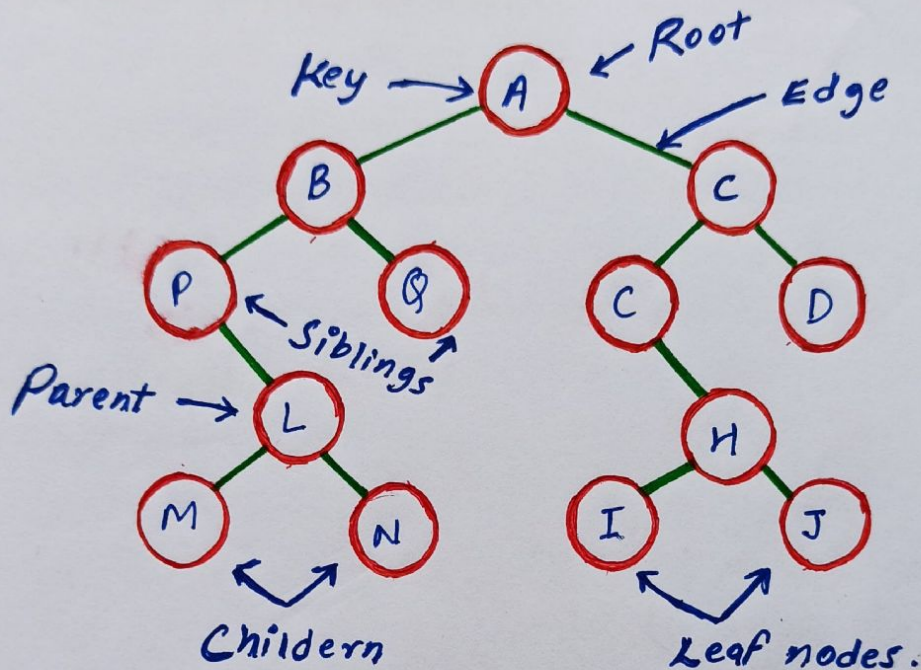
9]. What is Queue data structure?

⇒ • Queue is a linear data structure.
• It follows First In First Out (FIFO)
• Once a new element is inserted into queue all elements inserted before new element in queue must be removed to remove new element.

```
                              Front
| 7 |  Back

        ↘        | 6 | 5 | 4 | 3 | 2 |       First In First
                                                    Out.
                                          ↘
                                              | 1 |
                                               Out
```

10]. **What is tree in Data Structure?**

⇒ • Tree data structure is a kind of hierarchical data arrange in a tree-like structure.
• This tree consist of central node structural node, and sub nodes.
• It consist of node that node stores a value.
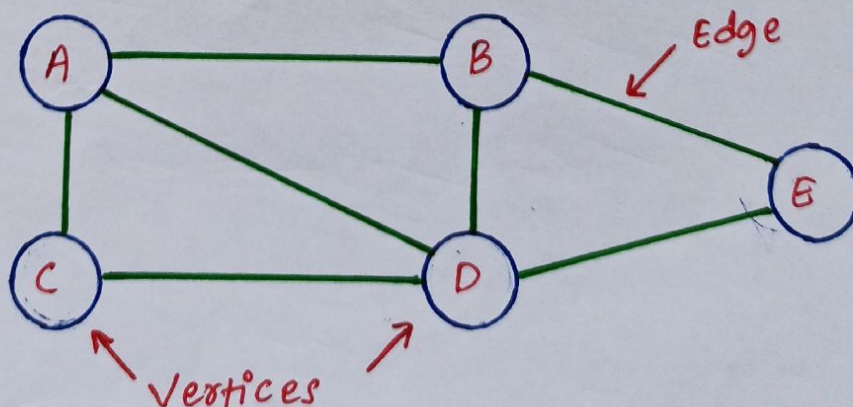• The topmost node called root node.

ATUL KUMAR (LINKEDIN).
TELEGRAM - NOTES GALLERY.

11]. **What is Graph in D.S.?**

⇒ • A Graph is consist of nodes & edges.
• Nodes are also called vertices & edges are arcs. that connects nodes in a graph.
• Graph looks like cyclic tree.

ATUL KUMAR (LINKEDIN).   4.
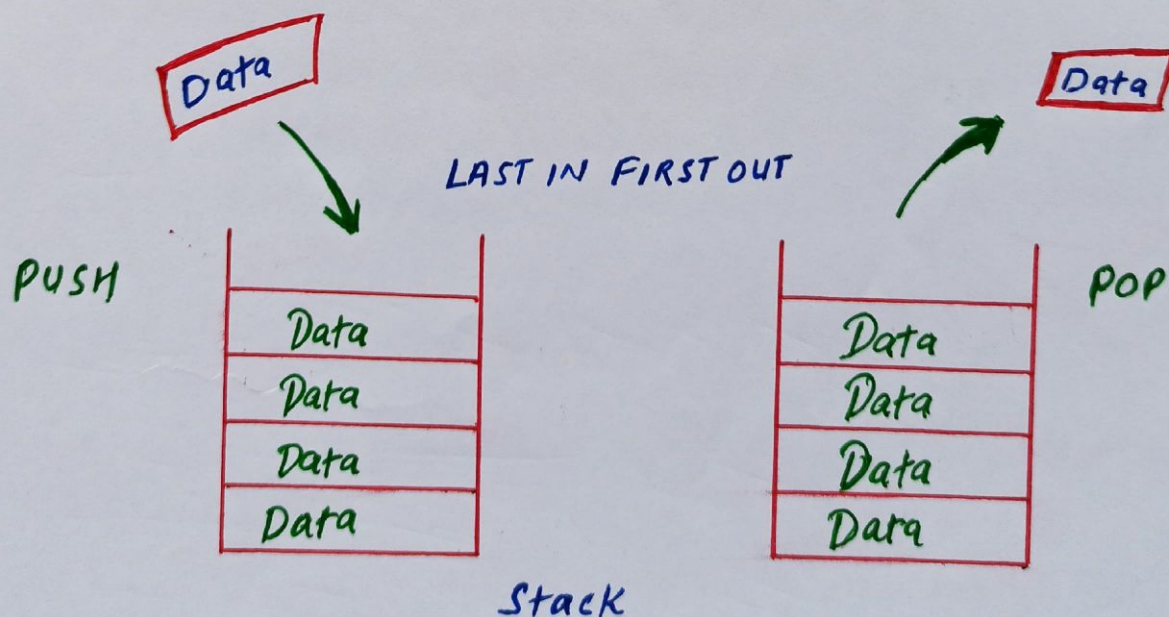
12]. <mark>What is an algorithm?</mark>

⇒ • An algorithm is a step by step method of solving a problem.

• It defines a set of instruction to be executed in certain order to get correct output.

• The algorithm designed are language independent.

13]. <mark>What is selection sort?</mark>

⇒ • In selection sort to find the minimum element in every iteration and place it in the array beginning from first index.

• Selection sort also divided into sorted and unsorted subarray.

14]. <mark>What is Push in D.S.?</mark>

⇒ • The PUSH Operator is used to insert a new element in the stack.

• PUSH Operation insert a new element at the top of stack.

• If insert a new element in full stack, a Overflow condition Occurs.

• PUSH add one element on top of the stack.

Data → PUSH

LAST IN FIRST OUT

Data → POP

PUSH		POP
Data		Data
Data		Data
Data		Data
Data		Data

Stack

15]. What is POP in D.S.?

⇒ • The POP operator is used to remove a element from stack.

• POP operation removes a element from top of stack.

• Stack underflow condition occurs when stack is empty and we try to delete element from stack.

16] What are dynamic D.S.?

⇒ • It is collection of data in memory that expand and contract to grow or shrink in size as a program runs.

• This enables the programmer to control exactly how much memory is to be utilized.

• Examples: dynamic array, linked list, stack, queue, and heap.

17]. What are the some application of D.S.?

⇒ Numerical analysis, Operating system, Artifical Intelligence, Simulation, compiler design, statistical analysis, graphics, database management.

18]. What is a Postfix expression?

⇒ • A Postfix expression is a collection of operator and operands in which the operator is placed after the operands.

• In a postfix expression the operator follows the operands.

19]. What is a dequeue?

⇒ • Dequeue is a type of queue in which insertion & removal of element can either be performed from the front or the rear.

• It not follows FIFO Rule.

20]. What are binary trees?

⇒ • A binary tree is a tree data structure in which each parent node can have at most two children.

• We named it Left child and right child.

• Binary tree contain :

i). Data

ii). Pointer to Left child.

iii). Pointer to Right child.

```
            ┌───┐
            │ 1 │
            └───┘
           /     \
      ┌───┐       ┌───┐
      │ 2 │       │ 3 │
      └───┘       └───┘
     /     \
┌───┐       ┌───┐
│ 6 │       │ 4 │
└───┘       └───┘
```