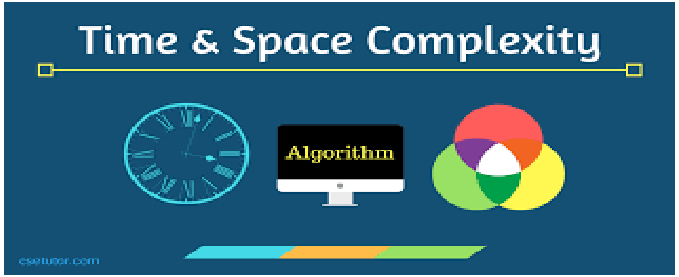
AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

CHAPTER 1
INTRODUCTION TO DATA STRUCTURE



SUBJECT:DATA STRUCTURE CODE:3130702	**PREPARED BY: ASST.PROF.PARAS NARKHEDE (CSE DEPARTMENT,ACET)**	AMIRAJ COLLEGE OF ENGINEERING & TECHNOLOGY

BASIC TERMINOLOGIES

- Data Structure
  - Data: are simply a value are set of values of different type which is called data types like string, integer, char etc.
  - Structure: Way of organizing information, so that it is easier to use
- In simple words we can define data structures as
  - Its a way organizing data in such a way so that data can be easier to use.

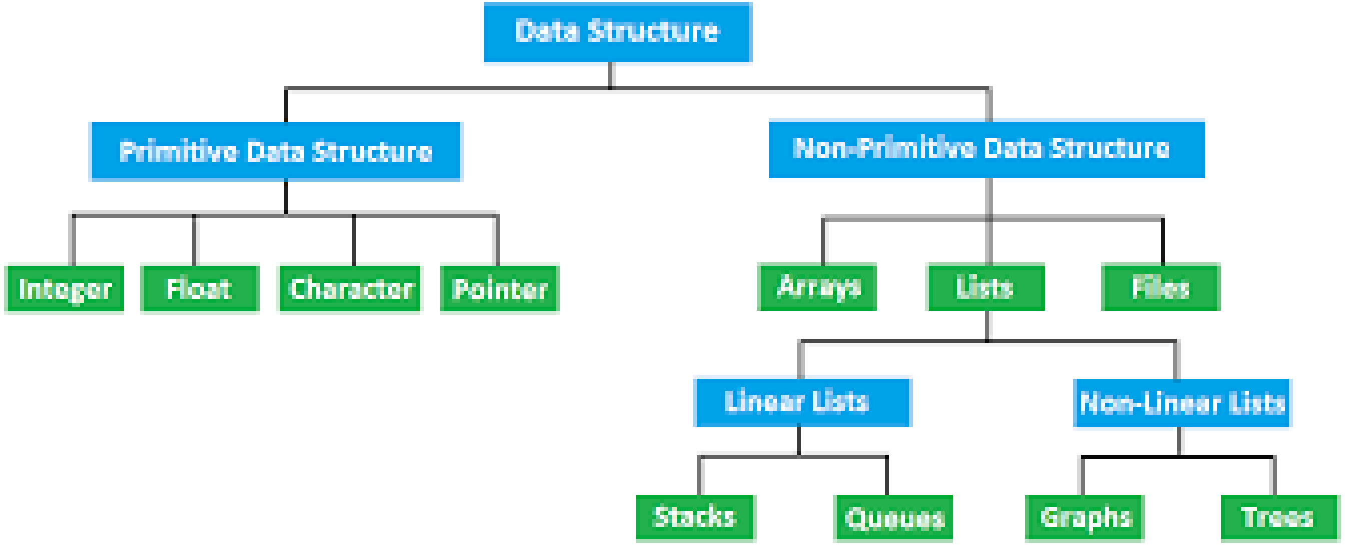
BASIC TERMINOLOGIES

- Data Structure ..
- A data structure is a particular way of organizing data in a computer so that it can be used efif ciently.
- A scheme for organizing related pieces of
- information.

WHY DATA STRUCTURE

- Human requirement with computer are going to complex day by day. To solve the complex requirements in efficient way we need this study.
- Provide fastest solution of human requirements. Provide efficient solution of complex problem.
  - Space
  - Time

CLASSIFICATION OF DATA STRUCTURE



AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

CLASSIFICATION OF DATA STRUCTURE

- Simple Data Structure: Simple data structure can be constructed with the help of primitive data structure. A primitive data structure used to represent the standard data types of any one of the computer languages (integer, Character, float etc.).

- Compound Data Structure: Compound data structure can be constructed with the help of any one of the primitive data structure and it is having a specific functionality. It can be designed by user. It can be classified as Linear and Non-Linear Data Structure

CLASSIFICATION OF DATA STRUCTURE

- **Linear Data Strcutre**:In the **linear data structure**, the **data** is organized in a **linear** order in which elements are linked one after the other. ... Examples of the **linear data structure** are array, queue, stack, linked list, etc. In contrast, tree and graph are the examples of the non-**linear data structure**.a

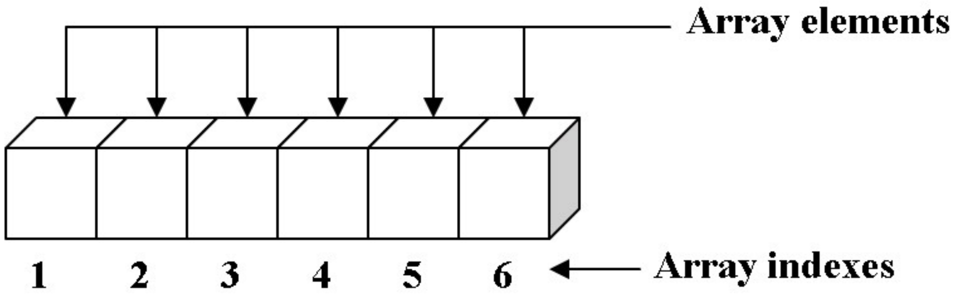
- **Non-Linear Data Structures**: Every data item is attached to several other data items in a way that is specific for reflecting relationships. The data items are not arranged in a sequential structure. Ex: Trees, Graphs

OPERATION ON LINEAR AND NON LINEAR

- Add an element
- Delete an element
- Sort the list of elements
- Search for a data element
- Traverse / Display

TYPES OF DATA STRUCTURE

ARRAY: is commonly used in computer programming to mean a contiguous block of memory locations, where each memory location stores one fixed-length data item. e.g. Array of Integers int a[10], Array of Character char b[10]



One-dimensional array with six elements

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

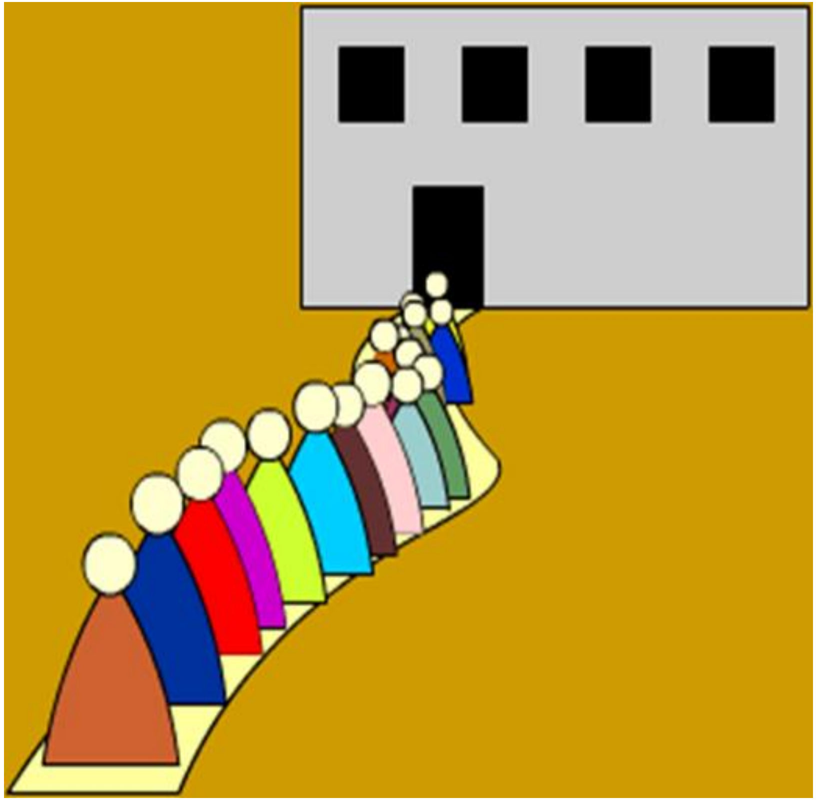
STACK

Stack: A stack is a data structure in which items can be inserted only from one end and get items back from the same end. There , the last item inserted into stack, is the the first item to be taken out from the stack. In short its also called Last in First out [LIFO].

AMIRAJ COLLEGE OF ENGINEERING & TECHNOLOGY

QUEUE

Queue is useful in CPU scheduling, Disk Scheduling. When multiple processes require CPU at the same time, various CPU scheduling algorithms are used which are implemented using **Queue data structure**. When **data** is transferred asynchronously between two processes. ... This Algorithm **uses Queue data structure**.



AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

LINKED LIST

Could alternately used to store items. In linked list space to store items is created as is needed and destroyed when space no longer required to store items. Hence linked list is a dynamic data structure space acquire only when need.

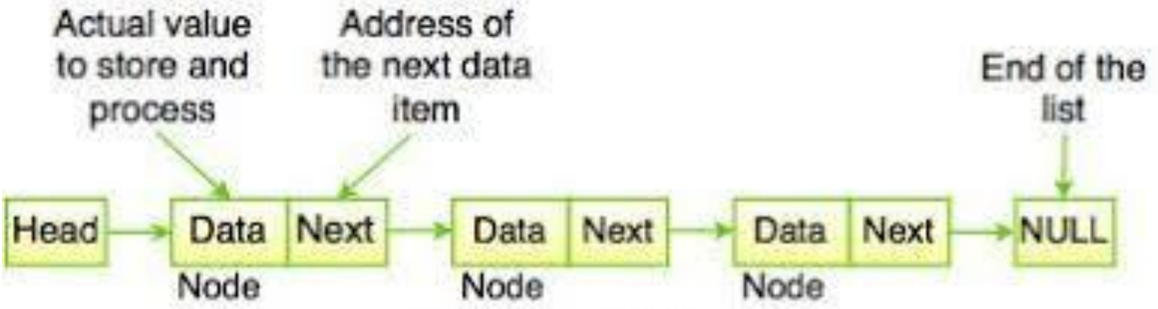
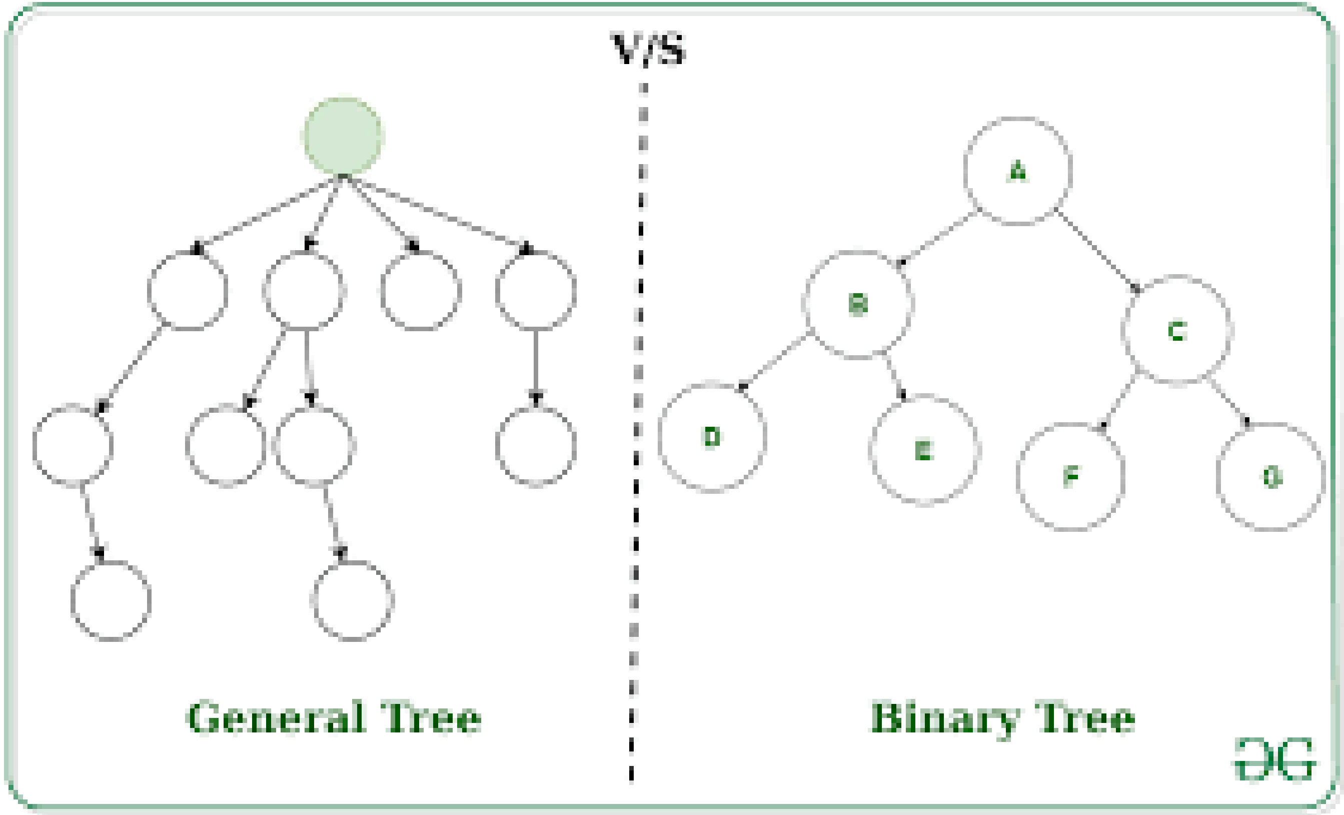

Actual value to store and process
Address of the next data item
End of the list
Head
Data
Next
Data
Next
Data
Next
NULL
Node
Node
Node

Fig. Linked List

TREE:

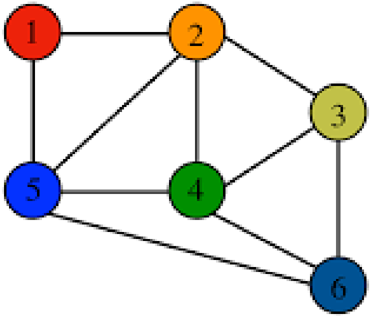
Tree: is a non-linear data structure which is mainly used to represent data containing a hierarchical relationship between elements.

Binary Tree: A binary tree is a tree such that every node has at most 2 child and each node is labeled as either left of right child.


V/S
A
B
C
D
E
F
G
General Tree
Binary Tree

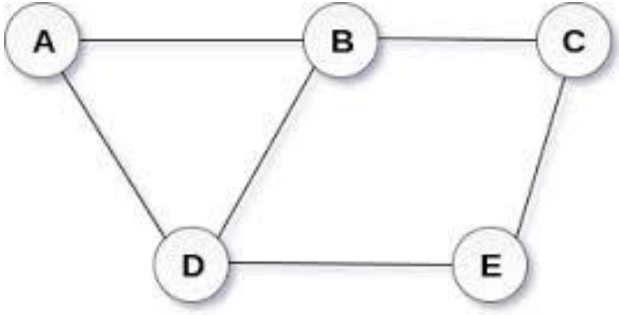
GRAPH

It is a set of items connected by edges. Each item is called a vertex or node. Trees are just like a special kinds of graphs. Graphs are usually represented by G = (V, E), where V is the set vertices and E is the set of Edges.



UNDIRECTED GRAPH

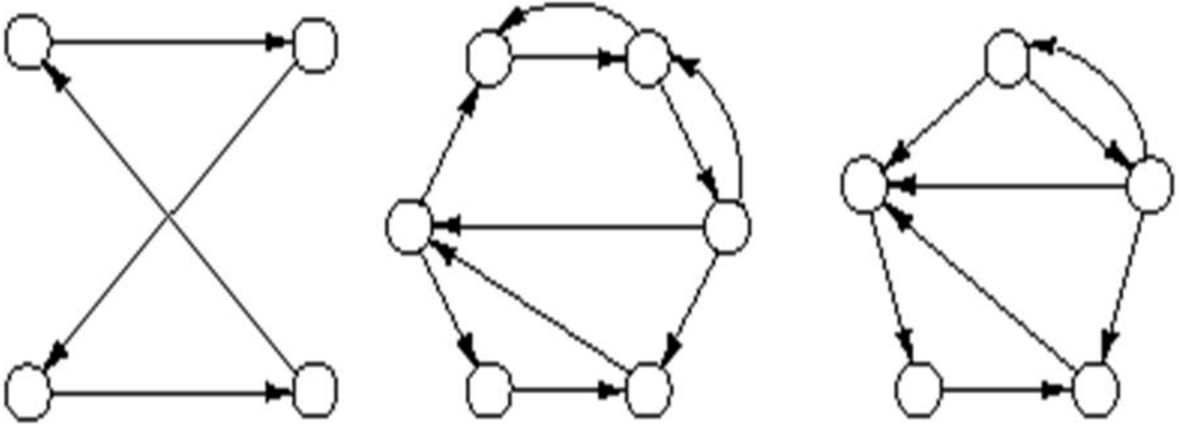
A graph whose edges are unordered pair of vertices. That is each edge connects two vertices. In an undirected graph, direction is not important, if the path is available, it can be traversed in any direction.



Undirected Graph

DIRECTED GRAPH

In directed graph a directional edge connect two node/vertex. If there is one edge from one vertex to other then only this path can be followed.



TIME AND SPACE ANALYSIS OF ALGORITHM

- Sometimes, there are more than one way to solve a problem.
- We need to learn how to compare the performance different algorithms and choose the best one to solve a particular problem.
- While analyzing an algorithm, we mostly consider time complexity and space complexity
- ***Time complexity*** of an algorithm quantifies the amount of time taken by an algorithm to run as a function of the length of the input

TIME AND SPACE ANALYSIS OF ALGORITHM

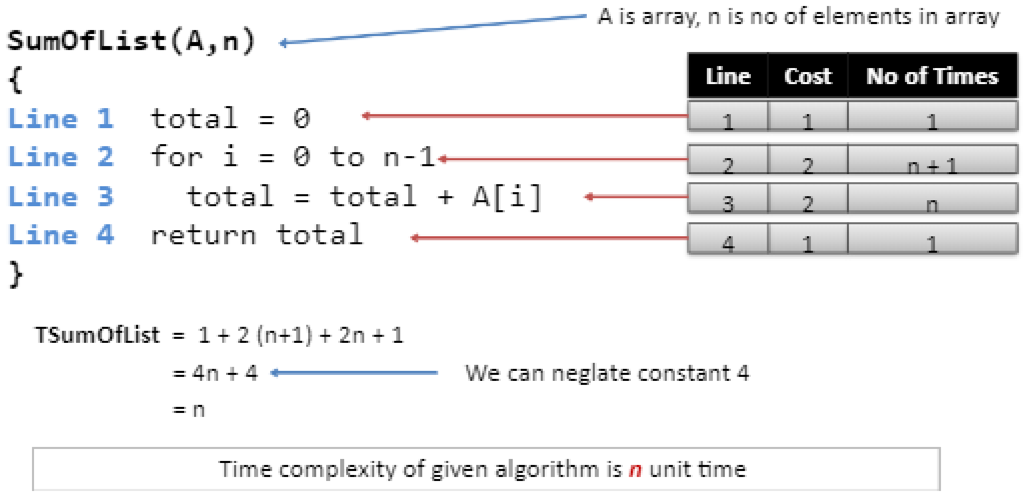
- ***Space complexity*** of an algorithm quantifies the amount of space or memory taken by an algorithm to run as a function of the length of the input
- Time & space complexity depends on lots of things like hardware, operating system, processors, etc.
- However, we don't consider any of these factors while analyzing the algorithm. We will only consider the execution time of an algorithm

CALCULATING TIME COMPLEXITY

- **Time Complexity** is most commonly **estimated** by **counting** the **number of elementary functions performed** by the algorithm
- Since the algorithm's performance may vary with different types of input data,
- hence for an algorithm we usually use the **worst-case Time complexity** of an algorithm because that is the maximum time taken for any input size

CALCULATING TIME COMPLEXITY

- Calculate Time Complexity of Sum of elements of List (One dimensional Array)



```
SumOfList(A,n)
{
Line 1  total = 0
Line 2  for i = 0 to n-1
Line 3    total = total + A[i]
Line 4  return total
}
```

A is array, n is no of elements in array

Line	Cost	No of Times
1	1	1
2	2	n+1
3	2	n
4	1	1

$$T_{SumOfList} = 1 + 2\,(n+1) + 2n + 1$$

$$= 4n + 4$$

We can neglate constant 4

$$= n$$

Time complexity of given algorithm is *n* unit time

AMIRAJ
COLLEGE OF ENGINEERING & TECHNOLOGY