

DATA STRUCTURE

PCC-CS-301

UNIT -I



LECTURE-1

Presented By

Neeta Sharma

Computer Science & Engineering

Noida International University

CONTENTS

3

- Basic Terminology
- Elementary Data
- Data Organization

Basic Terminology of Data Structure

4

➤ Data Structure

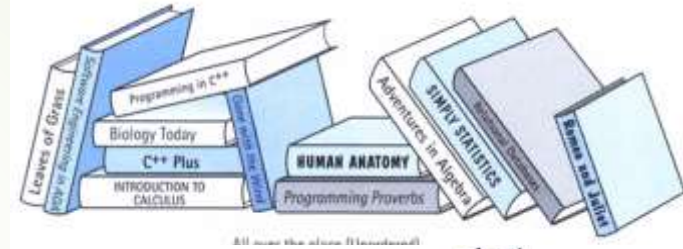
- The logical arrangement of **data elements**, combined with

OR

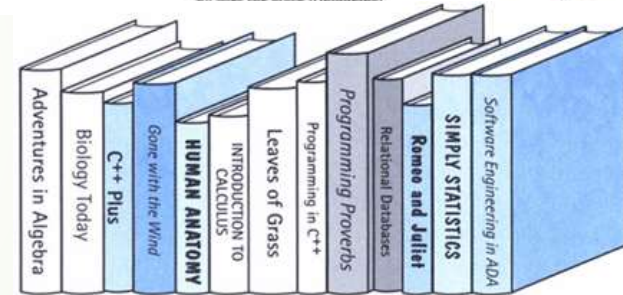
- The set of operations we need to access the elements.

➤ For example: Library

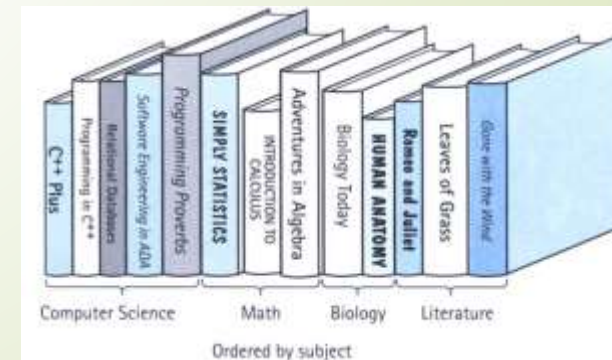
- It is composed of elements (books)
- Accessing a particular book requires knowledge of the arrangement of the books
- Users access books only through the librarian



All over the place (Unordered)



Alphabetical order by title



Ordered by subject

➤ **Data Element**

It is most fundamental level of data & data are simply values/set of values.

For example: Date is made up of three parts: day, month & year

Date is called a data object & day, month & year are called the **data elements** of data items (atomic).

Note: A data item is either the value of a variable or a constant.

➤ **Data Object:** It is a set of elements.

- Date,
- Fruit,
- Vegetable etc.

➤ **Atomic Variables**

- Atomic variables can only store one value at a time.

int num; or float s;

- A value stored in an atomic variable cannot be subdivided.

Cont....

6

- ▶ **Variable:** It is the most fundamental aspect of any computer language and can be defined as a location in the memory wherein a value can be manipulated, i.e., store, access, and modified.

For every variable there are two attributes: address and value

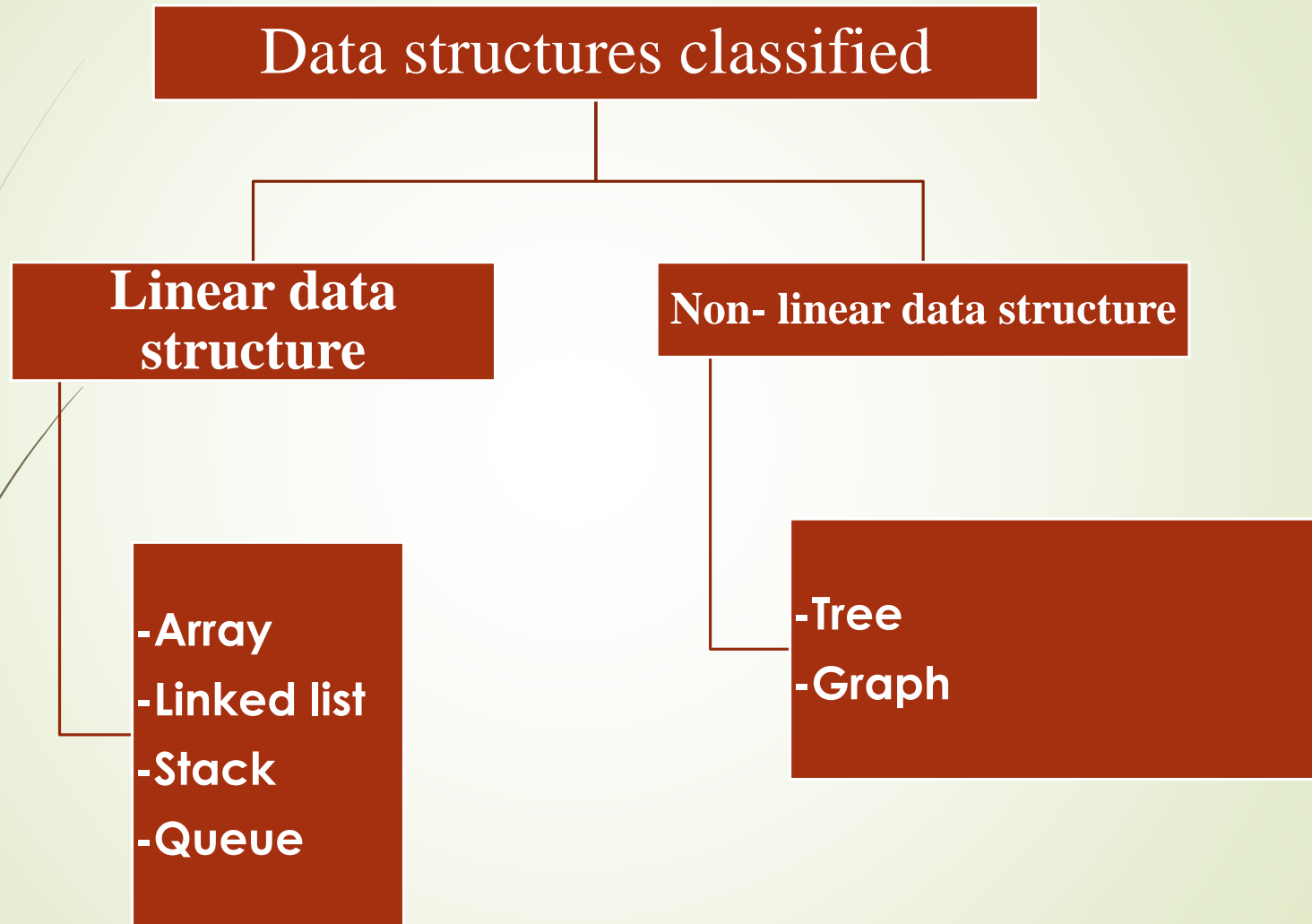
In memory with address 3: value: 45.

In memory with address 2: value "Dave"

- ▶ **Constant:** It is the memory location that does not change its constants during the execution of a program.
- ▶ **Primitive data type:** It is type of data that a variable may hold.
 - ▶ int,
 - ▶ float
 - ▶ char etc.

1	4096
2	"Dave"
3	45
4	"Matt"
5	95.5
6	"wbru"
7	0
8	"zero"

8	"zero"
7	0



➤ **Linear data structure:**

- Its elements form a sequence. This can be represents into two ways in the memory:

Linear relationship between the elements by means of

```
graph TD; A[Linear relationship between the elements by means of] --> B[sequential memory locations which is known as array/string]; A --> C[pointer/ links which is known as liked list];
```

sequential memory locations which is known as array/string

pointer/ links which is known as liked list

- **Non- Linear data structure:** Elements are not form a sequence and can be store anywhere in the memory .

➤ DATA STRUCTURES - AN OVERVIEW

Data structure is a particular way of storing and organizing data in a computer so that it can be used efficiently. Like array data structure, linked list, stack, queue and tree (database) etc.

➤ Arrays:

- An array is a data structure
- Used to process multiple elements with the same data type when a number of such elements are known.
- An array is a composite data structure; that means it had to be constructed from basic data types such as array integers.

1. `int a[5];`
2. `for(int i = 0;i<5;i++)`
 1. `{a[i]=i; }`

STRUCTURES

- Structures are used when you want to process data of multiple data types
- But you still want to refer to the data as a single entity
- Access data: **structurename.membername**

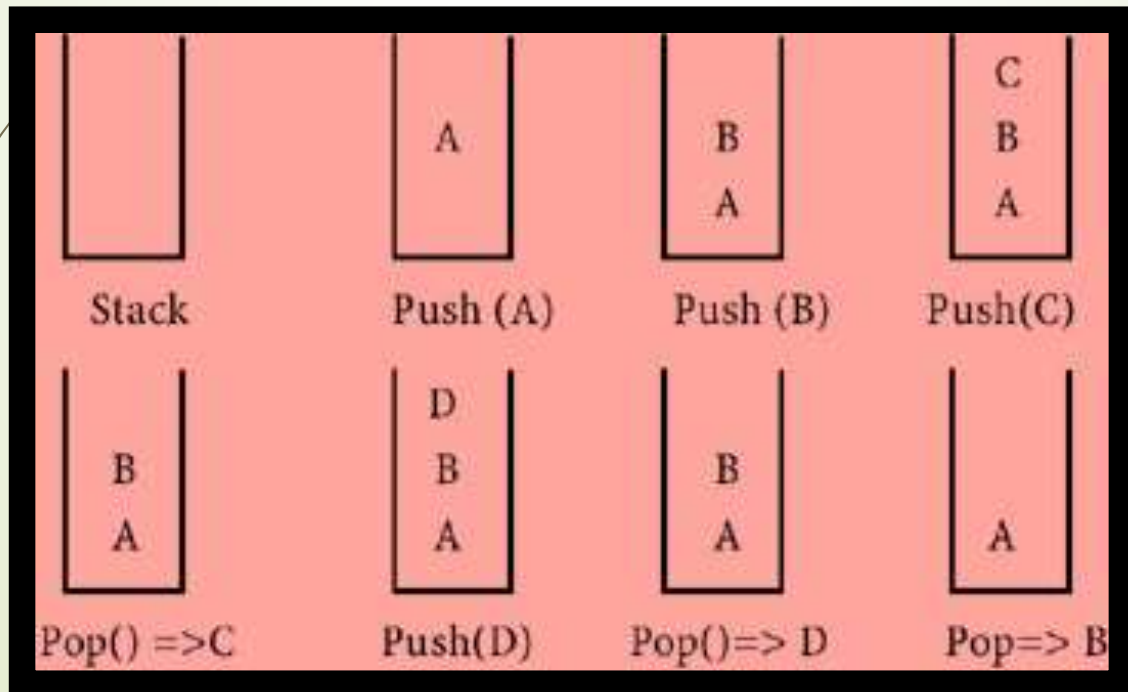
```
struct student
{
    char name[30];
    float marks;
};
```

Cont....

11

Stack

- ▶ A *stack* is memory in which values are stored and retrieved in "last in first out" manner by using operations called *push* and *pop*.



Queue

- A queue, also called a first-in-first-out (FIFO) system.
- It is a linear list in which deletions can take place only “front” of the list, and insertions can take place only “rear (last)” of the list.
- The features of a Queue are similar to the features of any queue like reservation counter.
- A queue can be implemented using arrays or linked lists. A queue can be represented as a circular queue. This representation saves space when compared to the linear queue.
- Finally, there are special cases of queues called De-queues which allow insertion and deletion of elements at both the end.

Tree:

- A tree is an acyclic, connected graph.
- A tree contains no loops or cycles.
- The concept of tree is one of the most fundamental and useful concepts in computer science.
- Trees have many variations, implementations and applications.
- Trees find their use in applications such as compiler construction, database design, windows, operating system programs, etc.
- A tree structures is one in which items of data are related by edges.
- A very common example is the ancestor tree.

Cont....

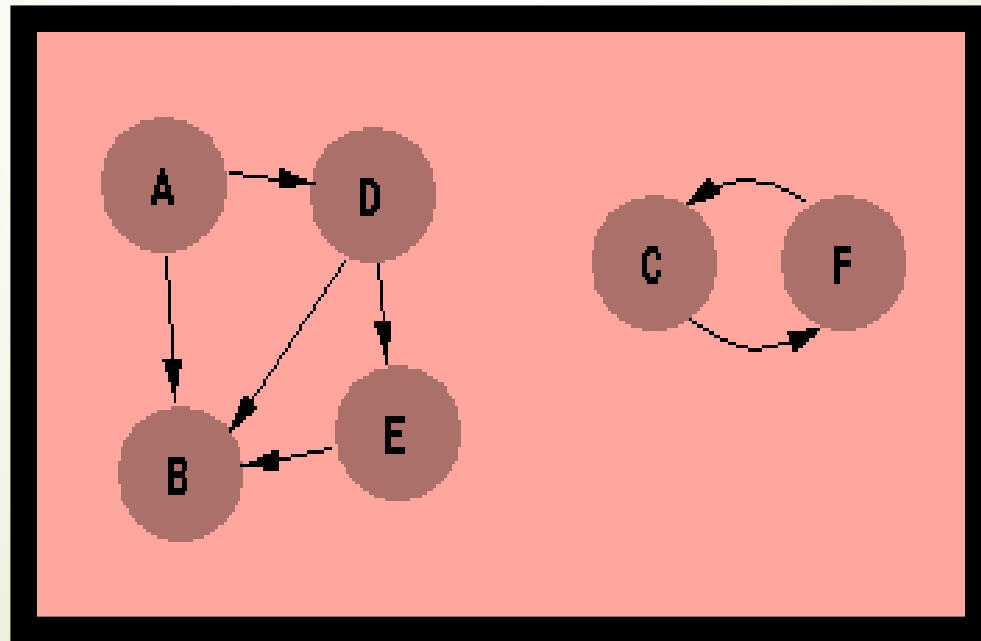
14

Graphs:

- Graphs are classified in the non-linear category of data structures.
- A graph G may be defined as a finite set V of vertices and a set E of edges (pair of connected vertices). The notation used is as follows:

Graph $G = (V, E)$

- Example of graph:



Thank You