

BCA 2<sup>nd</sup> Year

Data Structure

using

C & C++

# UNIT-1

## Introduction to data structure and its characteristics array

\* Data Structure :- Data structure is representation of the logical relationship existing between individual elements of data.

Data structure is the way of organising data items. Data structure mainly specify the following four things -

- 1- Organisation of data
- 2- Accessing methods
- 3- Degree of associability
- 4- Processing alternatives for information.

Operations of data structure :-

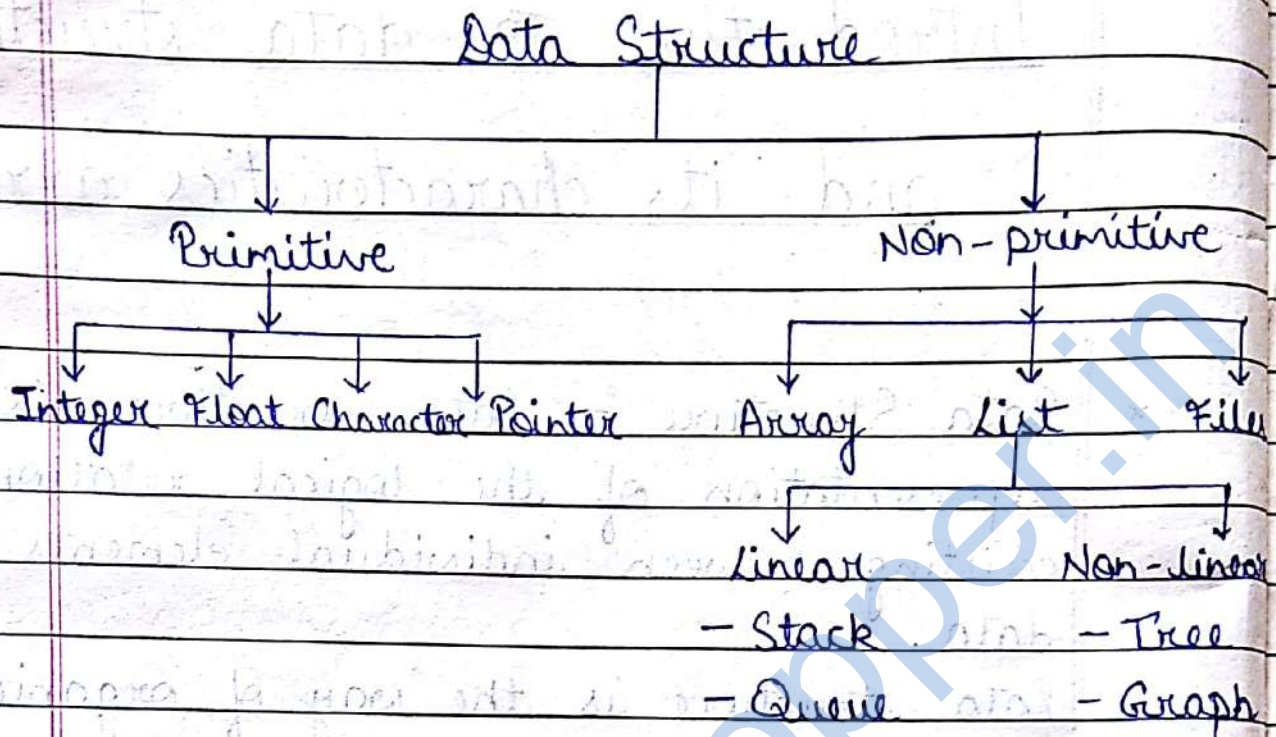
- (i) Insertion
- (ii) Deletion
- (iii) Updation
- (iv) Selection

Other operations -

- (i) Searching
- (ii) Sorting
- (iii) Merging



# Classification of data structures :-



\* **Array** :- Array forms an important part of almost all programming languages. It provides a powerful feature and can be used as such or can be used to form complex data structure like - stack and queue.

An array can be define as an infinite collection of homogeneous (similar type) elements.

1. Arrays are always stored in consecutive memory locations.
2. An array can be store multiple values which can be referenced by a single name unlike a simple variable which store one value at a time.
3. Array name is actually a pointer to



the first location of the memory block allocated to the name of the array.

4. An array either can be an integer, character or floating data types, can be initialized only during declaration time and not afterward.

### Types of array :-

- 1- One-dimensional array
- 2- Two-dimensional array
- 3- Multi-dimensional array

- 1- One-dimensional array :-

Initialization -

data-type variable-name [Expression];

Ex- `int ex[10];`

`char word ['H' 'E' 'L' 'L' 'O' ' ' 'o'];`

Accessing One-dimensional array -

To read a value -

`scanf ("%d", &[3]);`

For- `void main()`

`{`

`int a[10], i;`

`clrscr();`

`printf ("Enter the array");`

`for (i=0; i<9; i++)`



```

{
scanf ("%d", &a[i]);
}
for (i=0; i<9; i++)
{
printf ("%d\n", a[i]);
}
getch ();
}

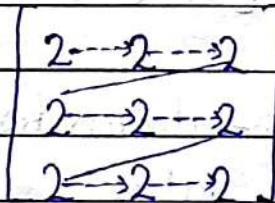
```

2- Two-dimensional array:- Two-dimensional array is of two types - row major and column major.

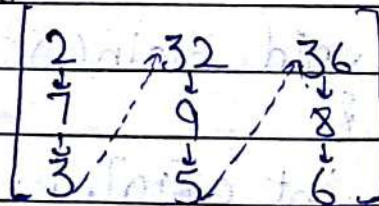
Initialization of 2-D array -

```
int a[2][2];
```

1) Row-major -



2) Column major -





### \* Triangular matrix :-

- 1- Upper triangular matrix - A square matrix, all of whose elements below principal diagonal are zero, is called an upper triangular matrix.

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix}$$

- 2- Lower triangular matrix - A square matrix, all of whose elements above principal diagonal are zero, is called a lower triangular matrix.

$$\begin{bmatrix} 1 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 5 & 6 \end{bmatrix}$$

- \* Sparse matrix :- In numerical analysis, a sparse matrix is in which most of the elements are 0.

Ex-

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

- \* Dense matrix :-

$$\begin{bmatrix} 3 & 0 & 9 \\ 2 & 1 & 6 \\ 1 & 1 & 0 \end{bmatrix}$$

\* Tridiagonal matrix :- Matrix that has non-zero elements on the main diagonal.

1. Ex- 
$$\begin{bmatrix} 2 & 3 & 4 & 0 \\ 1 & 5 & 6 & 7 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 6 & 1 \end{bmatrix} \quad \begin{bmatrix} 3 & 4 & 0 \\ 1 & 2 & 5 \\ 0 & 2 & 3 \end{bmatrix}$$