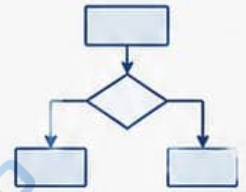
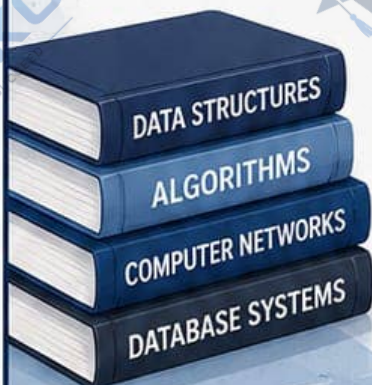
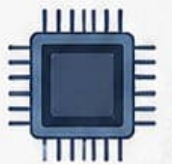


Unit-I



10110
01001
10101



Unit-I

classmate

Date _____

Page _____

SA-

1) What is Data Structure?

→ Data structure is a branch of computer science. The data structure is a way of organizing storing and managing data in a computer so that it can be accessed and modify efficiently. It defined the relationship between the data and the operations that can be performed on them.

2) Enlist example of Linear Data Structure?

→ In linear data structure the data stored in linear way and in contiguous memory location.

eg-

Array - Linked lists Stack Queue

3) Enlist example of Non-Linear Data Structure?

→ In Non-Linear data structure the data stored in herarchical way.

eg-

Tree Graph

4) What is linear data structure?

→ The linear data structure is a type of data structure where elements are stored in sequence and are connected one after another. The element of linear data structure are represented by means of sequential memory location.

5) What is Non-linear data structure?

→ The non-linear data structure is a type of data structure where elements are not arranged sequentially.

Instead, each element can be connected to multiple other elements, forming a hierarchical or network-like structure.

6) Enlist the example of stack?

→

6) What is Merging?

→ Merging is the process of combining two or more sorted lists into a single sorted list by arranging all elements in order.

7) What is array?

→ An array is a linear data structure. An array is a collection of multiple element of the same data type, stored in contiguous memory locations, and accessed using an index value.

8) What do you mean by Traversing?

→ Traversing means visiting each element of a data structure exactly once to perform some operation on it, such as displaying, searching, or modifying the elements.

9) What is Stack?

→ A stack is a linear data structure that stores elements in a last In, First Out (LIFO) order, meaning the last element inserted is the first one to be removed.

10) What is Queue?

→ A queue is a linear data structure that stores elements in a First In, First Out (FIFO) order, meaning the first element inserted is the first one to be removed.

11) What is Linked List?

→ A linked list is a linear data structure in which elements, called nodes, are stored in non-contiguous memory locations and connected using pointers.

L-A -

1) What is data structure? Explain the type with example?

→ Data Structure -

A data structure is a way of organizing storing and managing data in a computer so that it can be accessed and modify efficiently.

It defined the relationship between the data and the operations that can be performed on them.

Types of Data Structure.

Data structure are mainly divided into two broad categories -

- 1) Primitive Data Structure
- 2) Non-Primitive Data Structure.

1) Primitive Data Structure.

These are basic building block provided by programming languages.

eg. Integer (int) (1, 2, 3) -

Float (float) (1.23)

character (char) ('A', 'a').

Boolean (bool) (True or False)

Pointer

Holds memory location of another variable.

2) Non-Primitive Data Structure -

These are more complex and built using primitive data types. They are divided into two subtypes.

i) Linear Data Structure

ii) Non-Linear Data Structure.

i) Linear Data Structure -

In linear data structures, elements are arranged in a sequential (one-after-one) manner. Each element is connected to the next and/or previous one.

Example of Linear Data Structure -

i) Array -

An array is a collection of homogeneous (same type) data elements stored in contiguous memory location.

eg- `int arr[5] = {90, 1000, 10 10, 20, 30, 40, 50};`

ii) Linked List -

A linked list is a linear data structure where each element is called a node.

eg- `[data | next] → [data | next] → [data | next]`

iii) Stack -

A stack is a linear data structure that follows the LIFO principle. The last item added is the first to be removed.

iv) Queue -

A queue is a linear data structure that follows the FIFO principle. The first item added is the first to be removed.

ii) Non-Linear Data Structure -

In non-linear data structure, elements are not stored in sequential manner. They form a hierarchical or networked like structure.

eg -

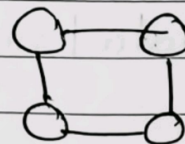
1) Tree -

It is made of nodes connected by edges. It resembles an upside-down tree.



2) Graph -

A graph is a non-linear data structure that consists of a set of nodes and a set of edges connecting them.



Array

Array

Array is the collection of homogeneous (same type) data elements stored in a contiguous memory location.

An array is used to store multiple values in a single variable.

The elements in array is accessed by using index of that element.
(Starting from 0 in most languages).

eg-

`int arr[5] = {1, 2, 3, 4, 5};`

index [0] = 1

index [1] = 2

index [2] = 3

and so on,

Declaration of Array

for declaring an array you have to specify its data type, name and its size

eg-

`int arr[5];`

↑
datatype

↓
name

↓
Size of array.

Types of Array-

Array is classified into three types

- 1) 1 D Array (One-dimensional)
- 2) 2 D (Two-dimensional array)
- 3) Multi Dimensional Array.

* 1 D Array-

A One-Dimensional array is the linear data structure in which the elements of the same datatype are stored in contiguous memory locations and are accessed using single index value.

Representation-

A one-dimensional array is represented as a single row of elements.

Eg-

`int A[5] = {10, 20, 30, 40, 50};`

Here,

`int` is the data-type, `A` is the name of array and `5` is the size of array. and elements are stored sequentially in memory.

* Memory Representation-

In one dimensional array, memory is allocated in a sequential manner.

This means that the address of the next element can be calculated from previous element by using simple formula.

Address of $A[i] = \text{Base Address} + [i] \times \text{Size of datatype}$

Eg-

$\text{int } A[5] = \{10, 20, 30, 40, 50\};$

If the base address of the array A is 1000 and the size of int is 4 bytes, then,

Element	Index	Address
$A[0]$	0	1000
$A[1]$	1	1004
$A[2]$	2	1008
$A[3]$	3	1012
$A[4]$	4	1016

Address 1000 1004 1008 1012 1016

10	20	30	40	50
0	1	2	3	4

Formula :-

Address of $A[i] = \text{Base Address} + [i] \times \text{Size of datatype}$

$\swarrow$ element at index $\swarrow$ of the first element $A[0]$ $\swarrow$ index $\swarrow$ size of datatype

Operations on Array-

An array supports various types of operations that allow storing, accessing, sorting and manipulating elements efficiently.

The most common operations performed on array are traversing, insertion, deletion, searching and sorting.

→ Traversing -

Traversing means accessing and processing each element of array one by one.

- Each element of array visited sequentially from index 0 to $n-1$.

- It is used to display or process all elements of array.

eg-

Printing all elements of array.

2) Insertion - (Time Consuming) $O(n^2)$

Insertion is the process of adding a new element at a specific position in the array.

Steps:-

- * Check if there is sufficient space in array
- * Shift the element to the right from given position.
- * Insert the new element at the desired index.

2) Deletion -

Deletion is the process of removing an element from a specific position in the array.

Steps-

Locate the element to be deleted

Shift element to the left to fill the gap

Reduce the size of array.

4) Searching -

Searching is used to find the location of a particular element in the array.

5) Sorting -

Sorting is used to arrange elements in ascending or descending order.