**# What is Data Type?**

-> Data types is used to defines the type of value to be used in a program. Based on the type of value specified in a program specified amount of required bytes will be allocated to the variables used in programs.

**A data type specified the type of data that a variable can store such as integer, float, character etc.**

**# Two important things about data type:**

1. Defines a certain **domain** of values.

2. Defines **Operations** allowed on those values.

**Example 1:**

int type

* Takes only integer values.
* Operations: addition, subtraction, multiplication, bitwise operations etc.

**Example 2:**

float type

* Takes only floating point values
* Operations: addition, subtraction, multiplication, division etc. (bitwise and % operation are not allowed).

**# What is Abstract Data Types?**

-> Abstract Data Types are like user defined data types which defines operations on values using functions without specifying what is there inside the function and how the operations are performed. ADT Provides Abstraction.

**Note: -** Let say, if someone wants to use the stack in the program, then he can simply use push and pop operations without knowing its implementation.

**Example: Stack ADT**

A stack consists of elements of same type arranged in a sequential order.

Operations:

Initialize () - initializing it to be empty.

Push () - Insert an element into the stack.

Pop () – Delete an element from the stack.

isEmpty () – checks if stack is empty.

isFull () – checks if stack is full.

**# There are three types of Abstract Data Types: -**

1. List ADT

2. Stack ADT

3. Queue ADT

**# What is Data Structure?**

-> Data Structure is a programming construct used to implement an ADT. It is the physical implementation of ADT.

**# Some common operations that are performed on Data Structures are: -**

1. Insertion

2. Deletion

3. Transversal

4. Search

**# Linear and Non Linear Data Structures**

**Linear Data Structure: -** A data structure is linear if all the elements are arranged in a linear order. In a linear data structure, each element has only one successor and only one predecessor. The examples of linear data structures are array, string, linked list, stack, and queue.

**Non Linear Data Structures: -** In Non Linear data structure, there is no linear order in the arrangement of the elements. The examples of non linear data structures are trees and graphs.

# What is Algorithms?

-> An Algorithm is a procedure having well defined steps for solving a particular problem. (The data stored in the data structures is manipulated by using different algorithms, so the study of data structures includes the study of algorithms.) Some of the common approaches of algorithms design are: -

1. **Greedy Algorithm**

2. **Divide and Conquer Algorithm**

3. **Backtracking**

4. **Randomized Algorithms**

5. **Analysis of Algorithms**