Data structure

A datastructure is a specialised format for organizing, processing retrieving and storing data. While there are several basic and advanced structure types, any data structure is designed to arrange data to suit a specific purpose so that it can be accessed & worked in appropriate ways.

In computer programming, a data structure may be selected or designed to store data for the purpose of working on it with various algorithms. Each data structure contains information about the data values, relationships between the data and functions that can be applied to the data.

Types of data structure :

Data structure types are determined by what types of operations are required or what kinds of algorithms are going to be applied. These types includes:

1) Arrays:

An array stores a collection of items at adjoining memory locations. Items that are the same type get stored together so that the position of each element can be calculated or retrieved easily.

2) Stacks:

A stack stores a collection of items in the linear order that operations are applied. This order could be last in first out (LIFO) or (FIFO)

3) Queues:

A queue stores a collection of items similar to a stack; however, the operation order can only be first in first out

4) linked list:

A linked list stores a collection of items in a linear order. Each element, or node, in a linked list contains a data item as well as a reference or link, to the next item in the list.

5) Tree:

A tree stores a collection of items in an abstract, hierarchical way. Each node is linked to other nodes and can have multiple sub values, also known as children.

6) Graphs:

A graph stores a collection of items in a non linear fashion. Graphs are made up of a finite set of nodes, also known as vertices and lines that connect them also known as edges.

7) Tries:

A trie or keyword tree is a data structure that stores strings as data items that can be organized in a visual graph

8) Hash table:

A hash table or a hash map, stores a collection of items in an associative array. that plots keys to values. A hash table uses a hash function to convert an index to an array of buckets that contain the desired data item.

Array as a data structure :

An array is a group of consecutive memory locations with same name and data type. Array is a collection of different adjacent memory location. All these memory locations have one collective name and type.

The memory locations in the essay are known as element of arrays. The total number of elements in the array is called length, The element of essay is accesied with reference to it's position in array, that is called index or subscript.

Different operations :

i) Traverse
ii) Insertion
iii) Deletion
iv) Search
v) Updata

i) Insertion operation:

Add an element at given index.

Algorithm:

```
Step 1: Start
Step 2: Set J=N
Step 3: Set N=N+1
Step 4: Repeat steps 5 & 6 while j>=k
Step 5: Set LA[J+1]=LA[J]
Step 6: Set J=J-1
Step 7: Set LA[K]=ITEM
Step 8: Stop
```

ii) Deletion operation:

Delete an element at given index.

Algorithm:

Consider LA is a linear array with N elements & K is a positive integer such that K<=N ~~elements~~.

```
Step 1: Start
Step 2: Set J=k
Step 3: Repeat step 4 & 5 while J<N
Step 4: Set LA[J-1] = LA[J]
Step 5: Set J=J+1
Step 6: Set N=N-1
Step 7: Stop
```

iii) Search operation.

Search an element using given index on by values.

Algorithm :

Consider LA is a linear array with N elements & K is a positive integer such that K<=N

```
Step 1 : Start
Step 2 : Set J=0
Step 3 : Repeat steps 4 while J<N
Step 4 : If LA[J] is equal ITEM then go to step 6
Step 5 : Set J=J+1
Step 6 : print J, item
Step 7 : Stop.
```

iv) Update operation:

Update the element at given index

Algorithm.

Consider LA is a linear array with N elements & K is a positive integer such that K<=N

```
Step 1 : Start
Step 2 : Set LA[K-1] = ITEM
Step 3 : Stop.
```