



## Sunbeam Institute of Information Technology Pune and Karad

### Module – Data Structures and Algorithms

Trainer - Devendra Dhande

Email – [devendra.dhande@sunbeaminfo.com](mailto:devendra.dhande@sunbeaminfo.com)



Sunbeam Infotech

[www.sunbeaminfo.com](http://www.sunbeaminfo.com)

## About Course

### • Contents

- Searching and Sorting Algorithms
- Time and Space Complexity Analysis
- Stack and its Applications
- Queue, Types of Queue and its Applications
- Linked Lists
- Binary Tree and its Applications
- Hashing
- Graph and its Algorithms
- Problem Solving Techniques – Divide and Conquer, Greedy, Dynamic Programming

- Daily notes, codes and diagrams will be shared on **github.com**
- During live lecture **Code share Utility** will be used
- Lecture **Video Recording** will be available for **next 7 days** on student portal

• **Dates** – 14<sup>th</sup> Aug to 30<sup>th</sup> Aug

• **Timings** – 4:00 pm to 8:00 pm (Mon - Sat)

• **Prerequisite** – Java Programming, Class and Object, Arrays, Java Collection (Array list)  
- Eclipse (Any)



Sunbeam Infotech

[www.sunbeaminfo.com](http://www.sunbeaminfo.com)

## Data Structures - Introduction

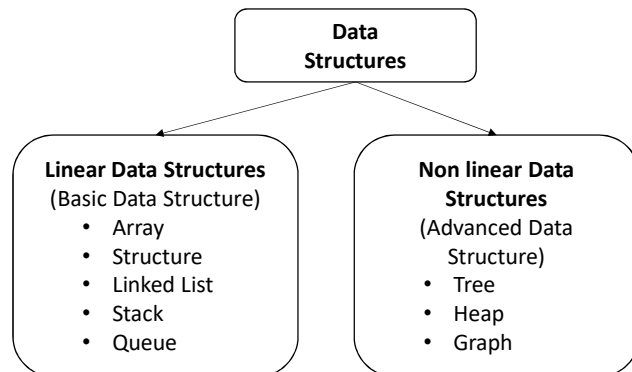
### What is Data structure?

- Organising data into memory
- Processing the data efficiently

### Why we need Data Structure?

To achieve

1. Efficiency
2. Reusability
3. Abstraction



#### Linear Data Structures

- Data elements are arranged linearly (sequentially) into the memory.
- Data elements can be accessed linearly / Sequentially.

#### Non linear Data Structures

- Data elements are arranged in non linear manner (hierarchical) into the memory.
- Data elements can be accessed non linearly.



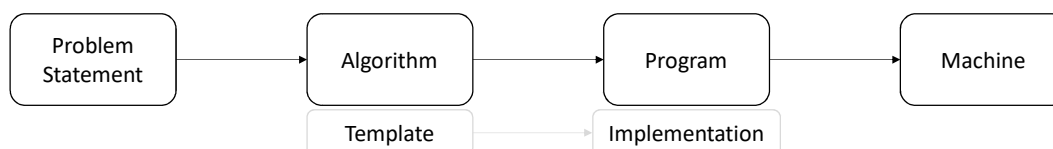
Sunbeam Infotech

www.sunbeaminfo.com

## Algorithm - Introduction

### • Algorithm

- is a clearly specified set of simple instructions.
- is a solution to solve a problem.
- is written in human understandable language.
- Algorithm is also referred as “pseudo code”.



e.g. Write an algorithm to find sum of all array elements.

Algorithm:

- Step 1: Initialize sum =0
- Step 2: Traverse array from index 0 to N-1
- Step 3: Add each element in the sum variable
- Step 4: Return the final sum

```

Algorithm SumArray(array, size) {
    sum = 0;
    For(index = 0 ; index < size ; index++)
        sum += array[index];
    return sum;
}
  
```



Sunbeam Infotech

www.sunbeaminfo.com

## Searching Algorithm : Linear Search

- Search a number in a list of given numbers (random order)

- **Algorithm**

- Step 1: Accept key from user
- Step 2: Traverse list from start to end
- Step 3: Compare key with each element of the list
- Step 4: If key is found return true else false

```
Algorithm linear_search(a, s, k)
{
    for(i = 0 ; i < s ; i++)
    {
        if(a[i] == key)
            return true;
    }
    return false;
}
```



Sunbeam Infotech

www.sunbeaminfo.com

## Searching Algorithm : Binary Search

- Given an integer x and integers A0, A1, ...An-1, which are pre-sorted and already in memory, find i such that Ai = x or return i = -1 if x is not in the input

- **Algorithm**

- Step 1: Accept key from user
- Step 2: Check if x is the middle element. If so x is found at mid
- Step 3: If x is smaller than the middle element, apply same strategy to the sorted subarray to the left of middle element.
- Step 4: If x is larger than the middle element, apply same strategy to the sorted subarray to the right of middle element



Sunbeam Infotech

www.sunbeaminfo.com



Thank you!

Devendra Dhande

<devendra.dhande@sunbeaminfo.com>



Sunbeam Infotech

[www.sunbeaminfo.com](http://www.sunbeaminfo.com)