# SNS COLLEGE OF TECHNOLOGY

**Coimbatore-35**
**An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

## DEPARTMENT OF INFORMATION TECHNOLOGY

## 19ITB201 – DESIGN AND ANALYSIS OF ALGORITHMS

II YEAR IV SEM

UNIT-I-Introduction

TOPIC: Introduction-Notion of an Algorithm

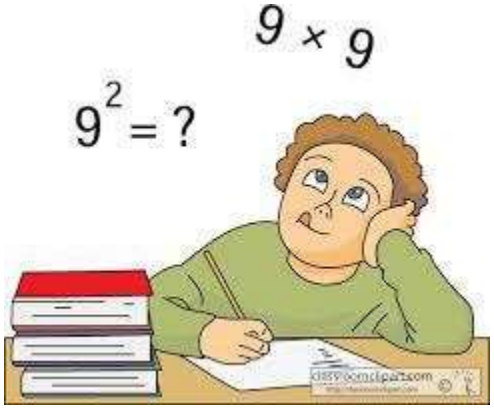
Prepared by
R.PADMAPRIYA

# INTRODUCTION-NOTION OF ALGORITHM

**Subject :Design and Analysis of Algorithm**
**Unit : I**

# What you identified from the picture?

Answer : Solving Problem



# How to solve the problem?

Answer: Problem solving methods

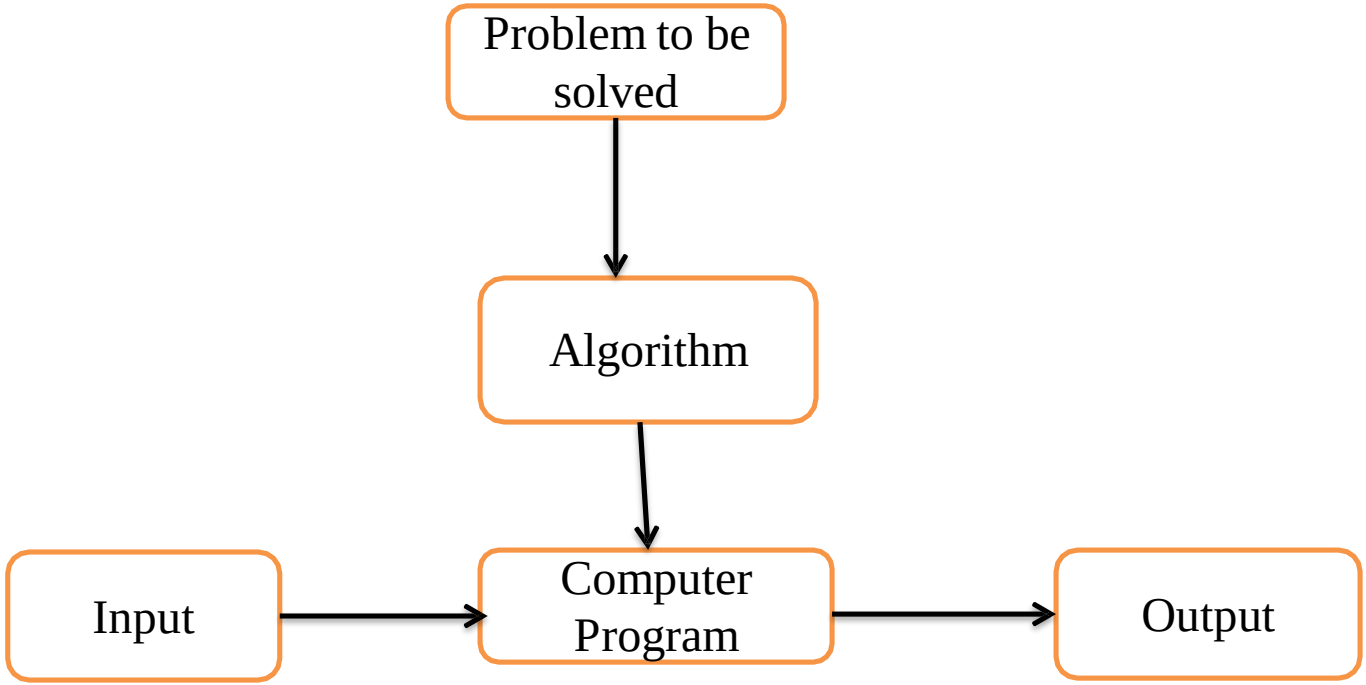
# Definition

An ***algorithm*** is a sequence of **unambiguous instructions**

for **solving a problem,** i.e., for obtaining a required outpu

for any legitimate input in a finite amount of time.

# Notion of an Algorithm



# Properties of an Algorithm

- The **non ambiguity** requirement for each step of an algorithm cannot be com- promised.


- The **range of inputs** for which an algorithm works has to be specified carefully.

# Properties of an Algorithm

- The same algorithm can be represented in **Several different ways**.

- There may exist **several algorithms** for solving the same problem

- Algorithms for the same problem can be based on very **different ideas and can solve the problem** with dramatically different speeds

# Characteristics of an algorithm

- **Input:** Zero / more quantities are externally supplied**.**
- **Output:** At least one quantity is produced.
- **Definiteness:** Each instruction is clear and unambiguous.
- **Finiteness:** If the instructions of an algorithm is traced then for all cases the algorithm must terminates after a finite number of steps.
- **Efficiency:** Every instruction must be very basic and runs in short time.

# Characteristics of an algorithm

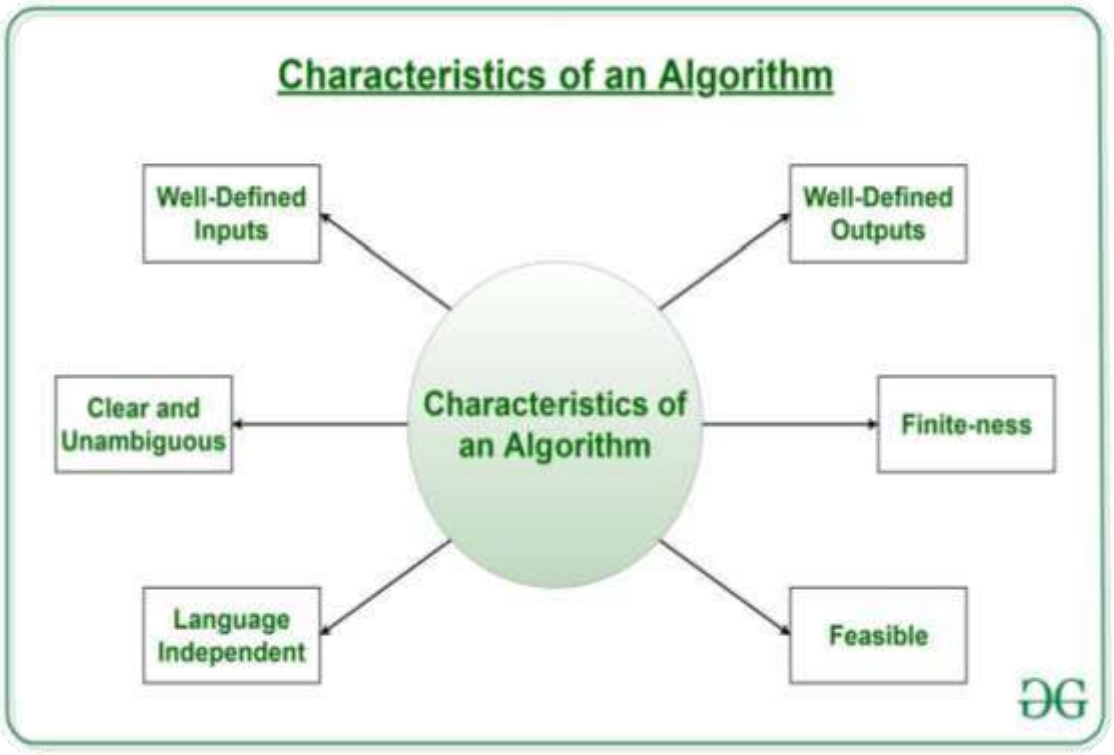



Image Source :https://www.geeksforgeeks.org/introduction-to-algorithms/

# Steps for writing an algorithm

- An algorithm is a procedure. It has two parts; the first part is **head** and the second part is **body**.
- The Head section consists of keyword **Algorithm** and Name of the algorithm with parameter list. E. g. Algorithm name1(p1, p2,…,p3). The head section also has the following:

**//Problem Description**

**//Input:**

**//Output:**

- In the body of an algorithm various programming constructs like **if, for, while** and some statements like assignments are used.
- The compound statements may be enclosed with **{** and **}** brackets. **if**, **for**, **while** can be closed by **en dif**, **endfor**, **endwhile** respectively. Proper indention is must for block.
- The **identifier** should begin by a letter and not by digit. It contains alpha numeric letters after first letter. No need to mention data types.
- Input and Output can be done using **read** and **write**.

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

Algorithmic problem solving involves a structured approach to creating step-by-step instructions (algorithms) to solve computational problems. It emphasizes understanding the problem, choosing appropriate data structures and algorithms, and analyzing their efficiency. The key is to find efficient solutions, considering both time and space complexity.

---

---

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

a) Algorithm Design Techniques

• An algorithm design technique (or “strategy” “paradigm”) is a general approach to solving problems algorithmically th is applicable to a variety of problems from different areas of computing.

• Though Algorithms and Data Structures are independent, but they are combined together to develop program. Hence the choice of proper data structure is required before designing the algorithm.

•

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

Implementation of algorithm is possible only with the help of Algorithms
and Data Structures

• Algorithmic strategy / technique / paradigm are a general approach by
which many problems can be solved algorithmically. E.g., Brute Force,
Divide and Conquer, Dynamic Programming,
Greedy Technique and soon.

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).

# FUNDAMENTALS OF ALGORITHMIC PROBLEM SOLVING

1.Understanding the Problem:

Read the Problem Statement:

Carefully examine the problem description understand the input, output, and constraints.

Clarify Doubts:

Ask questions to ensure a comple understanding of the problem's requirements.

Identify Problem Types:

Determine if the problem belongs to a known type (e.g., sorting, searching, graph traversal).


Thank you!