UNIT I (GE8151 PROBLEM SOLVING AND PYTHON PROGRAMMING)

ALGORITHMIC PROBLEM SOLVING
ALGORITHM

Algorithm is defined as the effective step-by-step procedure to solve the problem in a finite number of steps.
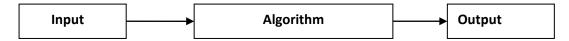

Fig 1: Algorithm and Input/Output Relationship

Algorithm is an ordered set of rules to solve a problem. It is a representation of a solution to a problem. It is a well-defined computational procedure consisting of a set of instructions that takes some value or set of values, as input, and produces some value or set of values, as output.

Program=Algorithm +Data Structures

Characteristics of an algorithm

- Each and every instruction should be precise and unambiguous.
- Algorithm should have finite number of steps.
- Algorithm should be written in sequence (step-by step).
- Algorithm should have finite number of inputs.
- Ensure that the algorithm will terminate.
- Results should be obtained only after the algorithm terminate.

Qualities of a good algorithm

- **Time:** A good algorithm should take less time to execute the program.
- **Memory:** A good algorithm should take less memory space to execute the program.
- **Accuracy:** A good algorithm should provide more accurate results.
- **Sequence:** A good algorithm should be written in sequence (step-by-step).
- **Understandability:** A good algorithm should be easily understandable.
- **Solvability:** A good algorithm should solve the problem.

Two important factors of an algorithm

- **Time complexity:** It is the amount of time required to complete a task.
- **Space complexity:** It is the amount of memory space required to complete a task.

Key features of an algorithm

- **Sequence:** Each instruction is executed in sequence (step-by-step).

- **Decision:** The result is based on some condition.

- **Repeat:** Process is repeated until condition becomes false.

Steps to develop an algorithm

i) An algorithm should be enclosed by two statements **START** and **STOP**.

ii) To read data from user **INPUT** or **READ** statement is used.

iii) To display the output **PRINT** statement is used.

iv) The arithmetic operators used are

+ - Addition operator

- - Subtraction operator

* - Multiplication operator

/ - Division operator

= - Assignment operator

v) Commonly used relational operators are

>   -- Greater than

<   -- Less than

>=   -- Greater than or equal to

<=   -- Less than or equal to

==   -- Equal to

vi) The commonly used logical

operators are AND, OR,

NOT

Example 1: Write an algorithm to find the sum of two numbers.

1. Start

2. Print "Enter two numbers:"

3. Read A, B

4. C=A+B

5. Print C

6. Stop

Example 2: Write an algorithm to swap two numbers.

1. Start

2. Print "Enter two numbers:"

3. Input a, b

4. c = a

5. a = b

6. b = c

7. Print a, b

8. Stop

Example 3: Construct an algorithm to check whether the given number is odd or even.

1. Start

2. Print "Enter numbers:"

3. Read n

4. r = n%2

5. If r = 0 then

6. Print "Number is even"

7. If r! =0 then

8. Print "Number is odd"

9. Stop

Example 4: Write an algorithm to find the area of the circle.

1. Start

2. Print "Enter radius:"

3. Input r

4. a = 3.14*r*r

5. Print a

6. Stop