What is an Algorithm?

In this tutorial, we will learn what algorithms are with the help of examples.

An algorithm is a set of well-defined instructions in sequence to solve a problem.

Qualities of a good algorithm

1. Input and output should be defined precisely.
2. Each step in the algorithm should be clear and unambiguous.
3. Algorithms should be most effective among many different ways to solve a problem.
4. An algorithm shouldn't include computer code. Instead, the algorithm should be written in such a way

Algorithm Examples

Algorithm to add two numbers

Algorithm to find the largest among three numbers

Algorithm to find all the roots of the quadratic equation

Algorithm to find the factorial

Algorithm to check prime number

Algorithm of Fibonacci series

Examples Of Algorithms In Programming

Write an algorithm to add two numbers entered by the user.

```
Step 5: Display sum
Step 6: Stop
```

Write an algorithm to find the largest among three different numbers entered by the user.

```
Step 1: Start
Step 2: Declare variables a,b and c.
Step 3: Read variables a,b and c.
Step 4: If a > b
           If a > c
              Display a is the largest number.
           Else
              Display c is the largest number.
        Else
           If b > c
              Display b is the largest number.
           Else
              Display c is the greatest number.
Step 5: Stop
```

Write an algorithm to find all roots of a quadratic equation $ax^2+bx+c=0$.

```
            r1 ← (-b+√D)/2a
            r2 ← (-b-√D)/2a
            Display r1 and r2 as roots.
      Else
            Calculate real part and imaginary part
            rp ← -b/2a
            ip ← √(-D)/2a
            Display rp+j(ip) and rp-j(ip) as roots
Step 5: Stop
```

Write an algorithm to find the factorial of a number entered by the user.

```
Step 1: Start
Step 2: Declare variables n, factorial and i.
Step 3: Initialize variables
          factorial ← 1
          i ← 1
Step 4: Read value of n
Step 5: Repeat the steps until i = n
     5.1: factorial ← factorial*i
     5.2: i ← i+1
Step 6: Display factorial
Step 7: Stop
```

Write an algorithm to check whether a number entered by the user is prime or not.

```
Step 4: Read n from the user.
Step 5: Repeat the steps until i=(n/2)
     5.1 If remainder of n÷i equals 0
            flag ← 0
            Go to step 6
     5.2 i ← i+1
Step 6: If flag = 0
           Display n is not prime
        else
           Display n is prime
Step 7: Stop
```

Write an algorithm to find the Fibonacci series till term≤1000.

```
Step 1: Start
Step 2: Declare variables first_term,second_term and temp
Step 3: Initialize variables first_term ← 0 second_term ←
Step 4: Display first_term and second_term
Step 5: Repeat the steps until second_term ≤ 1000
     5.1: temp ← second_term
     5.2: second_term ← second_term + first_term
     5.3: first_term ← temp
     5.4: Display second_term
Step 6: Stop
```

(https://twitter.com/intent/tweet?text=Check this amazing article: What is an Algorithm?&via=programiz&url=https://www.programiz.com/dsa/algorithm)
(https://www.facebook.com/sharer/sharer.php?u=https://www.programiz.com/dsa/algorithm)

Was this article helpful?

Related Tutorials

DS & Algorithms

Merge Sort Algorithm

(/dsa/merge-sort)

DS & Algorithms

Bellman Ford's Algorithm

(/dsa/bellman-ford-algorithm)

(/dsa/selection-sort)

DS & Algorithms

Prim's Algorithm

(/dsa/prim-algorithm)