Writing the Algorithm

A finite set of steps that must be followed to solve any problem is called an **algorithm**. Algorithm is generally developed before the actual coding is done. It is written using English like language so that it is easily understandable even by non-programmers.

Sometimes algorithms are written using **pseudocodes**, i.e. a language similar to the programming language to be used. Writing algorithm for solving a problem offers these advantages −

- Promotes effective communication between team members
- Enables analysis of problem at hand
- Acts as blueprint for coding
- Assists in debugging
- Becomes part of software documentation for future reference during maintenance phase

These are the characteristics of a good and correct algorithm −

- Has a set of inputs
- Steps are uniquely defined
- Has finite number of steps
- Produces desired output

Example Algorithms

Let us first take an example of a real-life situation for creating algorithm. Here is the algorithm for going to the market to purchase a pen.

1. Get dressed to go the market.
2. Check your wallet for money.
3. If there is no money in the wallet, replenish it.
4. Go to the shop.
5. Ask for your favorite brand of pen.
6. If pen is not available, go to step 7 else go to step 10
7. Give money to the shopkeeper.
8. Keep the purchased pen safely.
9. Go back home.
10. Ask for any other brand of pen.
11. Go to Step 7.

Step 4 in this algorithm is in itself a complete task and separate algorithm can be written for it. Let us now create an algorithm to check whether a number is positive or negative.

1. Print “Give any number”
2. Read num
3. if (num==0) print “You entered 0”
4. if (num>0) print “You entered a positive number”
5. if (num<0) print “You entered a negative number”