Algorithm to add two integer numbers

1. Start
2. Input an integer number A
3. Raise an error if non-integer number is entered by user
4. Input an integer number B
5. Raise an error if non-integer number is entered by user
6. Declare a variable, **Sum** to hold the output
7. Compute Raise an error if non-integer number is entered by user
8. Compute Sum = A+B
9. Print Sum
10. End

Algorithm to find factorial of positive integer number.

1. Start
2. Declare variable, **Number** to hold the number to entered
3. Enter a positive integer number
4. Raise an error if non-integer number is entered by user
5. Declare a variable, **Factorial** and initialize with 1 as value
6. Initiate loop
7. While Number is greater than zero, do the following:
   1. Multiply Factorial by Number and store result in Factorial
   2. Subtract 1 from Number
   3. End loop
8. Print Factorial
9. End

Algorithm to find the Fibonacci sequence

1. Start
2. Declare variables – **First\_Number, Second\_Numbe**r and **Term**
3. Initialize variables **First\_Number** and **Second\_Number** with numbers from user
4. Initialize variable **Term** with number of terms to be printed
5. Initiate a **counter** variable and set at zero
6. Initiate a loop
7. While **Counter** is greater than or equal to **Term,** do the following:
   1. Add **First\_Number , Second\_Number** and store in a variable called **Sum**
   2. Replace the value of **First\_Number** with **Second\_Number,** i.e First\_Number = Second\_Number
   3. Also replace the value of Second\_Number with value of Sum, i.e Second\_Number = Sum
   4. Increase Counter by 1 i.e Counter = Counter+1
8. End

Algorithm to convert from decimal base to base 2

1. Start
2. Enter number in decimal base and place in variable Number
3. Create a list with variable name *Converted* to store value
4. Declare variable Quotient and initialize value 1
5. Initiate loop
6. While Quotient is not equal to zero, do the following:
   1. Perform modular division, divide Number by 2 and place answer in Remainder variable ie. Remainder = Number%2
   2. Number by 2 and place result in Quotient ie. Quotient = Number/2
   3. Append Remainder in *Converted*
7. Reverse *Converted* and place in variable name *Re\_Converted*
8. Print *Re\_Converted*
9. Print *Re\_Converted*
10. End

Algorithm to Sort a given set of numbers (Bubble Sort)

1. Start
2. Determine the count/length of the series and place in variable name, N
3. Declare a variable Counter and initialize with Term i.e Counter = N
4. Initialize a loop
5. While (N-N)>(N-(N-1))>N-(N-2)…>N and Counter =N, do the following:

* If (N-N) > (N-(Counter-1)):

Place (N-(Counter-1)) in front of (N-N).

N=N-1

* Else

Place (N-N) in front of (N-(N-1))

N=N-1

1. Print Series
2. End