

Algorithm To Add Two Numbers

1. Enter the input arguments one after the other, and store their values in a variables “a” and “b”.
2. Test both inputs to ensure that they are numbers, of not pause the program and ask the user to input numbers as input arguments.
3. Add the first argument to the second argument
4. Return $a + b$

Algorithm To Find The Factorial Of A Number

1. Enter the input argument and store its value in a variable.
2. Test that the input is a positive integer if not pause the program and ask the user to input a positive integer.
3. Initialize an empty array or list
4. Create a loop in the function to iterate for the number of times of the integer argument received by the function starting from 1.
5. For each iteration, store the value under iteration into a set, list or array starting from 1 to the number specified.
6. Create another loop to iterate over all the numbers in the newly populated array or list, and at every instance, multiply the number under iteration to the next number after it and store the result in a variable
7. After the iteration return the variable.

Algorithm To Generate The Fibonacci Sequence

1. Initialize a variable to store a set/array/list of the first two numbers of the Fibonacci sequence (0 and 1)
2. Initialize a loop, and for every time the loop iterates, add the last number in the set at that instant to the second-to-last number at that same instant and append the result as the new last item of the set of numbers
3. This process can be continued infinitely, or set to stop at a certain point based on a certain condition.

Algorithm To Sort A Given Set Of Numbers

1. Store the given set/list/array of numbers in a variable.
2. Let's take the numbers of items in the set to be n .
3. Initialize a loop to run for n times.
4. In this loop, nest another loop which is also to iterate for n times. (This causes and piece of code placed in the nested loop to run for n^2 times).
5. Declare a condition, that at every point in the inner iteration, if a number is greater than the number next to it (to the right) in the set, both numbers should be swapped, else their order should be maintained.
6. Return the set, after the operation has been carried out for the complete n^2 times.

Algorithm To Convert A Given Number From One Base To Another

The base converter program is divided into three sections;

1. 1 Initializing a function that takes two arguments; a number and its base and returns its base 10 equivalent
2. 2 Initializing a second function that takes two arguments; a base 10 number and the base it is to be converted to, and returns the result of the number in the base specified.
3. 3 A third function that takes in three arguments; the number to be converted, the base it is to be converted from, and the base it is to be converted to.

Breaking down the operations of these functions

1. The First function:

- i. In the first function, the both the number to be converted and the base of the number are tested to be positive integers, if they are not, the program is halted and the user is asked to input positive values as arguments.
- ii. After that, the number to be converted is splitted into an array of its constituent elements, and stored in a variable;
- iii. A variable " n " is also declared to hold the result of the operation about to be carried out and is initialized to 0
- iv. An iteration is carried out on the number to be converted, and for every digit in the number to be converted, the number at every instance added to " n " and then multiplied by the base of the number and the result is assigned back to the " n " variable.

- v. After carrying out this process for every digit in the input number "n" is returned

2. The Second Function

- i. In the second function both the number to be converted and the base it is to be converted to are tested to be positive integers, if not the program is paused and the user is asked to input positive integer values as arguments.
- ii. Initialize an array and make a copy of the number to be converted and store them in variables
- iii. While the copy of the input number to be converted is less than 1, divide the number by the base it is to be converted to, obtain the remainder, and remove any decimal points it may have, and then prepend it to the initialized empty array.
- iv. Still inside the while loop, divide the copy of the number to be converted by the base it is to be converted to and reassign the result back to the variable holding the copy of the number to be converted.
- v. Join all the numbers in the array to form one single number and return the result.

3. The Third Function

- i. Initialize a third function with all three arguments specified above, and test that all values received are positive integer values, if not the program is paused and the user is asked to input positive integer values as arguments.
- ii. Invoke the first function and pass to it the number to be converted and the base as arguments,
- iii. From the result of the operation above, pass it as first argument to the second function, and the base the number is to be converted to, as the second argument of the second function.
- iv. Return the result of the operation above

To test the algorithm invoke the third function and pass to it the 3 arguments specified above.