1. an array a[ ] of size N which contains elements from 0 to N-1 (length of the array), you need to find all the duplicates of the numbers inside the array. all the elements occurring more than once in the given array.

Example 1:

Input:

N = 4

a[] = [0,3,1,2]

Output: -1

Explanation: N=4 and all elements from 0

to (N-1 = 3) are present in the given

array. Therefore output is -1.

Example 2:

Input:

N = 5

a[] = [2,3,1,2,3]

Output: 2 3

Explanation: 2 and 3 occur more than once

in the given array.

Your Task:

Create the function duplicates() which takes array a[ ] and n as parameters and returns a list of elements that occur more than once in the given array and it is going to be sorted. If no such element is found, return list containing [-1].

Expected Time Complexity: O(n).

Expected Auxiliary Space: O(n).

1. You have an array A of positive numbers (integers). Your code is required to find the leaders in the array. An element of an array is considered to be leader if it is greater than or equal to all the elements to its right side. The right most element is always a leader.

Example 1:

Input:

n = 6

A[] = [16,17,4,3,5,2]

Output: 17 5 2

Explanation: The first leader is 17

as it is greater than all the elements

to its right. Similarly, the next

leader is 5. The right most element

is always a leader so it is also

included.

Example 2:

Input:

n = 5

A[] = [1,2,3,4,0]

Output: 4 0

Your Task:

You don't need to read input or print anything. You only have to call the function leader and display the leader integers

Expected Time Complexity: O(n)

Expected Auxiliary Space: O(n)

1. an array of size N-1 that only contains unique integers in the range of 1 to N. You are required to find the missing element.

Example 1:

Input:

N = 5

A[] = [1,2,3,5]

Output: 4

Example 2:

Input:

N = 10

A[] = [6,1,2,8,3,4,7,10,5]

Output: 9

Your Task :

You don't need to read input or print anything. Complete the function MissingNumber() that takes array and N as input parameters and returns the value of the missing number.

Expected Time Complexity: O(N)

Expected Auxiliary Space: O(1)

**Bonus Exercise: Write a JavaScript program to convert Binary number (positive) to Decimal using recursion.**