

Repeat and Missing Number Array

There are certain problems which are asked in the interview to also check how you take care of **overflows** in your problem.

This is one of those problems.

Please take extra care to make sure that you are type-casting your ints to long properly and at all places. **Try to verify if your solution works if number of elements is as large as 10^5**

Food for thought :

- Even though it might not be required in this problem, in some cases, you might be required to order the operations cleverly so that the numbers do not overflow. For example, if you need to calculate $n! / k!$ where $n!$ is factorial(n), one approach is to calculate factorial(n), factorial(k) and then divide them. Another approach is to only multiple numbers from $k + 1 \dots n$ to calculate the result. Obviously approach 1 is more susceptible to overflows.

You are given a read only array of n integers from 1 to n .

Each integer appears exactly once except A which appears twice and B which is missing.

Return A and B .

Note: Your algorithm should have a linear runtime complexity. Could you implement it without using extra memory?

Note that in your output A should precede B .

Example:

Input: [3 1 2 5 3]

Output: [3, 4]

$A = 3, B = 4$

Sample Input 2:

2

10

4 5 2 9 8 1 1 7 10 3

7

7 5 3 2 1 6 6

Sample Output 2:

6 1

4 6