**Q:-1**

**Duplicate in array**

Send Feedback

Given an array of integers of size n which contains numbers from 0 to n - 2. Each number is present at least once. That is, if n = 5, numbers from 0 to 3 is present in the given array at least once and one number is present twice. You need to find and return that duplicate number present in the array.

Assume, duplicate number is always present in the array.

**Input format :**

Line 1 : Size of input array

Line 2 : Array elements (separated by space)

**Output Format :**

Duplicate element

**Constraints :**

1 <= n <= 10^6

**Sample Input:**

9

0 7 2 5 4 7 1 3 6

**Sample Output:**

7

**Q:2**

**Triplet sum**

Send Feedback

Given a random integer array and a number x. Find and print the triplets of elements in the array which sum to x.

While printing a triplet, print the smallest element first.

That is, if a valid triplet is (6, 5, 10) print "5 6 10". There is no constraint that out of 5 triplets which have to be printed on 1st line. You can print triplets in any order, just be careful about the order of elements in a triplet.

**Input format :**

Line 1 : Integer N (Array Size)

Line 2 : Array elements (separated by space)

Line 3 : Integer x

**Output format :**

Line 1 : Triplet 1 elements (separated by space)

Line 2 : Triplet 3 elements (separated by space)

Line 3 : and so on

**Constraints :**

1 <= N <= 1000

1 <= x <= 100

**Sample Input:**

7

1 2 3 4 5 6 7

12

**Sample Output ;**

1 4 7

1 5 6

2 3 7

2 4 6

3 4 5

**Q:-3**

**Rotate array**

Send Feedback

Given a random integer array of size n, write a function that rotates the given array by d elements (towards left)

Change in the input array itself. You don't need to return or print elements.

**Input format :**

Line 1 : Integer n (Array Size)

Line 2 : Array elements (separated by space)

Line 3 : Integer d

**Output Format :**

Updated array elements (separated by space)

**Constraints :**

1 <= N <= 1000

1 <= d <= N

**Sample Input :**

7

1 2 3 4 5 6 7

2

**Sample Output :**

3 4 5 6 7 1 2

**Q:-4**

**Sum me Up**

Send Feedback

There will be ‘t’ test cases having an integer. You have to sum up all the digits of this integer. For e.g. For 6754, the answer will be 6 + 7 + 5 + 4 = 22.

**Input Format:**

First line will have an integer ‘t’ denoting the number of test cases.

Next ‘t’ lines will have an integer ‘val’ each.

**Output format:**

Print ‘t’ lines of output denoting the sum of all the digits of the number in each test case.

**Constraints:**

1 <= t <= 10^5

0 <= val <= 10^18

**Sample Input:**

2

1547

45876

**Sample Output:**

17

30

**Explanation:**

1 + 5 + 4 + 7 = 17

4 + 5 + 8 + 7 + 6 = 30

**Q:-5**

**Longest Increasing Subsequence**

Send Feedback

You are given with an array of integers that contain numbers in random order. Write a program to find the longest possible sub sequence of consecutive numbers using the numbers from given array.

You need to return the output array which contains consecutive elements. Order of elements in the output is not important.

Best solution takes O(n) time.

If two arrays are of equal length return the array whose index of smallest element comes first.

**Input Format :**

Line 1 : Integer n, Size of array

Line 2 : Array elements (separated by space)

**Constraints :**

1 <= n <= 10^5

**Sample Input 1 :**

13

2 12 9 16 10 5 3 20 25 11 1 8 6

**Sample Output 1 :**

8

9

10

11

12

**Sample Input 2 :**

7

15 13 23 21 19 11 16

**Sample Output 2 :**

15

16