1. Find the nth smallest element in any given array .

n Array   Output
2 [3,4,5,6] 4
3 [9 ,3 ,2,0] 3

2. Given any number n add the digits in the number until only a single digit remains . Print all the resulting numbers uptill the final number .

n   Output
11   2
199   19 10 1
255   12 3
5   5

3. Least frequent element in an array

Given an array, find the least frequent element in it. If there are multiple elements that appear least number of times, print any one of them.

Examples :

Input : arr[] = {1, 3, 2, 1, 2, 2, 3, 1}

Output : 3

4. Given a word remove all repeated characters.

Input   Output
Hello   Heo
World World
There Thr

5. Check if array contains contiguous integers with duplicates allowed

Given an array of n integers(duplicates allowed). Print "Yes" if it is a set of contiguous integers else print "No".

Examples:

Input : arr[] = {5, 2, 3, 6, 4, 4, 6, 6}

Output : Yes

The elements form a contiguous set of integers

which is {2, 3, 4, 5, 6}.

Input : arr[] = {10, 14, 10, 12, 12, 13, 15}
Output : No

6. Sort Without Moving:

Some people are standing in a row in a park. There are trees between them which cannot be moved. Your task is to rearrange the people by their heights in a non-descending order without moving the trees.

Example
For a = [-1, 150, 190, 170, -1, -1, 160, 180], the output should be
sortByHeight(a) = [-1, 150, 160, 170, -1, -1, 180, 190].
Input/Output
[input] array.integer a
If a[i] = -1, then the ithposition is occupied by a tree. Otherwise a[i] is the height of a person standing in the ithposition.
Guaranteed constraints:
5 ≤ a.length ≤ 15,
-1 ≤ a[i] ≤ 200.
[output] array.integer
Sorted array a with all the trees untouched

E1. Given an array containing integers, zero is considered an invalid number and rest all other numbers are valid. If two nearest valid numbers are equal then double the value of the first one and make the second number as 0. At last move all the valid numbers on the left.

Input:
The first line of input contains an integer **T** denoting the number of test cases. The first line of each test case consists of an integer **n**. The next line consists of **n** spaced integers.

Output:
Print the resultant array.

Constraints:
1<=T<=100
1<=N<=10000
1<=A[i]<=10000

Example:
Input:
1
12
2 4 5 0 0 5 4 8 6 0 6 8

Output:
2 4 10 4 8 12 8 0 0 0 0

Q7. We are given an array asteroids of integers representing asteroids in a row.

For each asteroid, the absolute value represents its size, and the sign represents its direction(positive meaning right, negative meaning left). Each asteroid moves at the same speed.

Find out the state of the asteroids after all collisions. If two asteroids meet, the smaller one will explode. If both are the same size, both will explode. Two asteroids moving in the same direction will never meet.

Input:
Array[of astroids]
Output:
Array[of remaining astroids after collision]

Constraints:
The length of asteroids will be at most 10000.
Each asteroid will be a non-zero integer in the range [-1000, 1000]

Test Case 1:
Input:
asteroids = [4, 15, -5]

Output: [4, 15]
Explanation:
The 15 and -5 collide resulting in 15. 4 and 15 never collide as they are moving in the same direction.

Test Case 2:
Input:
asteroids = [3, -3]

Output: []
Explanation:
The 3 and -3 collide exploding each other.
Test Case 3:
Input:
asteroids = [8, 3, -6,-2]

Output:[8]
Explanation:
The 3 and -6 collide resulting in -6. The 8 and -6 collide resulting in 8, and -2 and 8 collide resulting in 8.

Test Case 4:
Input:
asteroids = [-5, -10, 12, 2]

Output:[-5, -10, 12, 2]
Explanation:
The -5 and -10 are moving left, while the 12 and 2 are moving right.
Asteroids moving the same direction never meet, so no asteroids will meet each other.

Q8. You are given an array of integers representing coordinates of obstacles situated on a straight line.

Assume that you are jumping from the point with coordinate 0 to the right. You are allowed only to make jumps of the same length represented by some integer.

Find the minimal length of the jump enough to avoid all the obstacles.
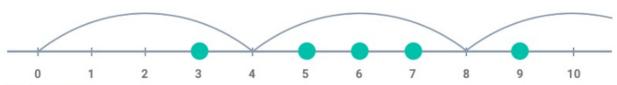
For inputArray = [5, 3, 6, 7, 9],

the output should be 4

Check out the image for explanation You are given an array of integers representing coordinates of obstacles situated on a straight line.

Assume that you are jumping from the point with coordinate 0 to the right. You are allowed only to make jumps of the same length represented by some integer.

Find the minimal length of the jump enough to avoid all the obstacles.

For inputArray = [5, 3, 6, 7, 9],
the output should be 4
Check out the image for explanation

Test Cases:
1.   inputArray: [2, 3]
      Output : 4
2.   inputArray: [1, 4, 10, 6, 2]
      Output: 7
3.   inputArray: [1000, 999]
      Ouput: 6
4.   inputArray: [19, 32, 11, 23]
      Output: 3
5.   inputArray: [5, 8, 9, 13, 14]
      Output: 6