Take an **array arr** of size **N** as input which represents a **large number**. **Add X** to this large number and print the resultant array.

eg:-for X = 2 and array is [4,2,3,6,5,8,7,1,5,3,9,6] In this case answer must be [4,2,3,6,5,8,7,1,5,3,9,8]

Note: The large integer does not contain any **leading 0's** in the array.

Input Format

First line contains an integer **N** representing the length of array.

Second line contains **N** integers representing the elements of array.

Third line contains an integer **X** .

Constraints

```
1 <= N <= 1000000
0 <= arr[i] <= 9
0 <= X <= 9
```

Output Format

print the **resultant** array.

Sample Input 0

```
10
1 8 7 5 2 2 9 3 7 4
9
```

Sample Output 0

```
1 8 7 5 2 2 9 3 8 3
```

Submitted Code

Language: Java 15

```
1  import java.io.*;
2  import java.util.*;
3
4  public class Solution {
5
6      public static void main(String[] args) {
7        Scanner sc = new Scanner(System.in);
8          int n = sc.nextInt();
9          int arr[] = new int[n];
10         for(int i =0;i<n;i++)arr[i]=sc.nextInt();
11         int x = sc.nextInt();
12         int res[]= calculateArray(arr,x);
13
14         for(int i=0;i<res.length;i++)System.out.print(res[i]+" ");
15     }
16     public static int[] calculateArray(int [] arr, int x){
17         for(int i=arr.length-1;i>=0;i--){
18             int val = arr[i]+x;
19             arr[i]=val%10;
20             x=val/10;
21         }
22
23          if(x <=0){
24            return arr;
25        }else{
26             int val =1;
27              int newArray[] = new int[arr.length+1];
28             newArray[0]=x;
29              for(int i=0;i<arr.length;i++){
30                  newArray[val]=arr[i];
31                  val++;
32              }
33              return newArray;
34          }
35
36
37     }
38 }
```


```
for (int i=0; i<n ; i++)
    for (int j=0 ; j<m ; j++)
```