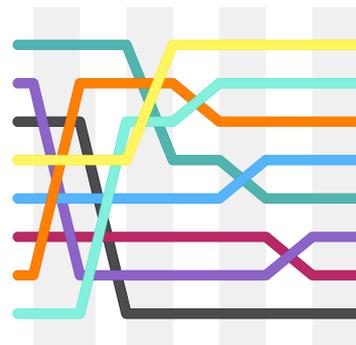


I In-place Sorting

Woe is you – for your algorithms class you have to write a sorting algorithm, but you missed the relevant lecture! The subject was in-place sorting algorithms, which you deduce must be algorithms that leave each input number in its place and yet somehow also sort the sequence.

Of course you cannot change any of the numbers either, then the result would just be a different sequence. But then it hits you: if you flip a 6 upside-down, it becomes a 9, and vice versa! Certainly no one can complain about this since you changed none of the digits! The deadline to hand in the exercise is in five hours. Try to implement this sorting algorithm before then!



CC BY-SA 4.0 by Balu Ertl on Wikimedia

Input

The input consists of:

- A line with an integer n ($2 \leq n \leq 10\,000$), the number of integers in the input sequence.
- n lines, the i th of which contains a positive integer x_i ($1 \leq x_i \leq 10^{18}$), the i th number of the sequence.

Output

If the sequence cannot be sorted in non-decreasing order by flipping some of the digits 6 or 9 in the input¹, output “impossible”. Otherwise, output “possible” followed by the sorted sequence – each number on its own line.

If there are multiple valid solutions, you may output any one of them.

Sample Input 1

4	possible 6 7 7 9
9	
7	
7	
9	

Sample Output 1

Sample Input 2

4	possible 67 69 99 190
97	
96	
66	
160	

Sample Output 2

¹Flipping any of the digits of n is not allowed.

Sample Input 3

3	impossible
80	
97	
79	

Sample Output 3**Sample Input 4**

2	possible
197	
166	

Sample Output 4

167
169