Problem 1 – Mirror Numbers

You are given **n 4-digit numbers**. Write a program to find among these numbers all pairs of mirror numbers, such that **the reversed positions of digits in the first number are equal to the positions of digits of the second number**. Note that both numbers should be distinct (**a** ≠ **b**). Put the sign “<!>” between the numbers. For instance: 1234<!>4321, 6789<!>9876.

Input

The input comes from the console. The first line holds the **count n**. The next line holds **n 4-digit integer numbers**, separated by a space. The input numbers will be **distinct** (no duplicates are allowed).

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

Print at the console all **mirror numbers** {**a**, **b**} found in the input sequence in format "**a<!>b**" (without any spaces), each at a separate line. The **order** of the output lines **is not important**. Print "**No**" in case no stuck numbers exist among the input sequence of numbers.

Constraints

- The **count n** will be an integer number in the range [1…50].
- The input **numbers** will be **distinct** integers in the range [1111…9999].
- Time limit: 0.5 sec. Memory limit: 16 MB.

Examples

Input	Output
5 1234 4321 9876 1122 6789	1234<!>4321 9876<!>6789

Input	Output
3 1345 1243 2021	No

Input	Output
7 2222 1234 4321 1322 2231 5312 1231	1234<!>4321 1322<!>2231