# 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 **countn**. 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 **countn** 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** |  | **Input** | **Output** |
| 5  1234 4321 9876 1122 6789 | 1234<!>4321  9876<!>6789 | 3  1345 1243 2021 | No |

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

// Read the input sequence of n integers

Scanner str = **new** Scanner(System.in);

str.nextLine();

String[] numbers = str.nextLine().split(" ");

// Find and print all mirror numbers {a<!>b}

**boolean** validMatch = **false**;

**for** (**int** i = 0; i < numbers.length; i++) {

**for** (**int** j = i + 1; j < numbers.length; j++) {

String a = numbers[i];

String b = numbers[j];

**if** (a.charAt(0) == b.charAt(3) && a.charAt(1) == b.charAt(2) &&

a.charAt(2) == b.charAt(1) && a.charAt(3) == b.charAt(0)) {

System.out.printf("%s<!>%s%n", a, b);

validMatch = **true**;

}

}

}

**if** (!validMatch) {

System.out.println("No");

OR

// Read the input sequence of n integers

Scanner str = **new** Scanner(System.in);

**int** n = Integer.parseInt(str.nextLine());

**int**[] numbers = **new** **int**[n];

**for** (**int** i = 0; i < numbers.length; i++) {

numbers[i] = str.nextInt();

}

// Find and print all mirror numbers {a<!>b}

**boolean** validMatch = **false**;

**for** (**int** i = 0; i < numbers.length; i++) {

**for** (**int** j = i + 1; j < numbers.length; j++) {

String a = Integer.toString(numbers[i]);

String b = Integer.toString(numbers[j]);

**if** (a.charAt(0) == b.charAt(3) && a.charAt(1) == b.charAt(2) &&

a.charAt(2) == b.charAt(1) && a.charAt(3) == b.charAt(0)) {

System.out.printf("%s<!>%s%n", a, b);

validMatch = **true**;

}

}

}

**if** (!validMatch) {

System.out.println("No");