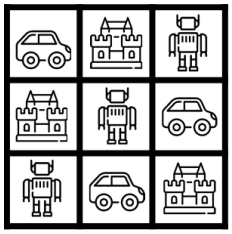
Latin Squares



A Latin Square is an n-by-n array filled with n different digits, each digit occurring exactly once in each row and once in each column. (The name "Latin Square" was inspired by the work of Leonhard Euler, who used Latin characters in his papers on the topic.)

A Latin Square is said to be in *reduced form* if both its top row and leftmost column are in their natural order. The natural order of a set of digits is by increasing value.

Your team is to write a program that will read an n-by-n array, and determine whether it is a Latin Square, and if so, whether it is in reduced form.

1 Input

The first line of input contains a single integer n ($2 \leq n \leq 36$). Each of the next n lines contains n digits in base n, with the normal digits '`0`' through '`9`' for digit values below 10 and uppercase letters '`A`' through '`Z`' representing digit values 10 through 35. All digits will be legal for base n; for instance, if n is 3, the only legal characters in the n input lines describing the square will be '`0`', '`1`', and '`2`'.

2 Output

If the given array is not a Latin Square, print "`No`" on a single line (without quotation marks). If it is a Latin Square, but not in reduced form, print "`Not Reduced`" on a single line (without quotation marks). If it is a Latin Square in reduced form, print "`Reduced`" on a single line (without quotation marks).

3 Sample Input and Output

3 012 120 201	Reduced
4 3210 0123 2301 1032	Not Reduced
11 0123458372A A9287346283 0285475A834 84738299A02 1947584037A 65848430002 038955873A8 947530200A8 93484721084 95539A92828 04553883568	No