Assume you have a square of size n that is divided into $n \times n$ positions just as a checkerboard. Two positions (x_1, y_1) and (x_2, y_2), where $1 \leq x_1, y_1, x_2, y_2 \leq n$, are called "independent" if they occupy different rows and different columns, that is, $x_1 \neq x_2$ and $y_1 \neq y_2$. More generally, n positions are called independent if they are pairwise independent. It follows that there are $n!$ different ways to choose n independent positions.

Assume further that a number is written in each position of such an $n \times n$ square. This square is called "homogeneous" if the sum of the numbers written in n independent positions is the same, no matter how the positions are chosen. Write a program to determine if a given square is homogeneous!

Input

The input contains several test cases.

The first line of each test case contains an integer n ($1 \leq n \leq 1000$). Each of the next n lines contains n numbers, separated by exactly one space character. Each number is an integer from the interval [-1000000,1000000].

The last test case is followed by a zero.

Output

For each test case output whether the specified square is homogeneous or not. Adhere to the format shown in the sample output.

Sample Input

```
2
1 2
3 4
3
1 3 4
8 6 -2
-3 4 0
0
```

Sample Output

```
homogeneous
not homogeneous
```