Time limit: 1.0s Memory limit: 256M

Canadian Computing Competition: 2006 Stage 1, Senior #4

In mathematics, a *group*, *G*, is an object that consists of a set of elements and an operator (which we will call \times) so that if *x* and *y* are in *G* so is $x \times y$. Operations also have the following properties:

- Associativity: For all x, y and z in G, $x \times (y \times z) = (x \times y) \times z$.
- Identity: the group contains an "identity element" (we can use *i*) so that for each x in G, $x \times i = x$ and $i \times x = x$.
- Inverse: for every element x there is an inverse element (we denote by x^{-1}) so that $x \times x^{-1} = i$ and $x^{-1} \times x = i$.

Groups have a wide variety of applications including the modeling of quantum states of an atom and the moves in solving a Rubik's cube puzzle. Clearly, the integers under addition from a group (0 is the identity, and the inverse of x is -x, and you can prove associativity as an exercise), though that group is infinite and this problem will deal only with finite groups.

One simple example of a finite group is the integers modulo 10 under the operation addition.

That is, the group consists of the integers 0, 1, ..., 9 and the operation is to add two keeping only the least significant digit. Here the identity is 0. This particular group has the property that $x \times y = y \times x$, but this is not always the case. Consider the group that consists of the elements a, b, c, d, e and i. The "multiplication table" below defines the operations. Note that each of the required properties is satisfied (associativity, identity and inverse) but, for example, $c \times d = a$ while $d \times c = b$.

×	i	a	b	c	d	e
i	i	a	b	с	d	e
a	a	i	d	e	b	c
b	b	e	i	d	c	a
c	c	d	e	i	a	b
d	d	c	a	b	e	i
e	e	b	c	a	i	d

Your task is to write a program which will read a sequence of multiplication tables and determine whether each structure defined is a group.

Input Specification

The input will consist of a number of test cases. Each test case begins with an integer n ($0 \le n \le 100$). If the test case begins with n = 0, the program terminates. To simplify the input, we will use the integers $1, \ldots, n$ to represent elements of the candidate group structure; the identity could be any of these (i.e., it is not necessarily the element 1). Following the number n in each test case are n lines of input, each containing integers in the range $[1, \ldots, n]$. The q^{th} integer on the p^{th} line of this sequence is the value $p \times q$.

Output Specification

If the object is a group, output yes (on its own line), otherwise output no (on its own line). You should not output anything for the test case where n = 0.

Sample Input

$ \begin{array}{cccccccccccccccccccccccccccccccccccc$		
1 2 2 1 6 1 1 2 3 4 6 1 5 6 1 2 3 4 3 1 5 4 3 6 1 2 4 5 6 1 3 4 2 3 6 1 2 3 7 7 7 7 1 1 1 1 3 1 1 1 4 1 1 1 5 1 1 1 6 1 1 1 7 1 1 1 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 2 3 3 1 2 3 1 2	2	
2 1 6 1 2 3 4 5 2 1 5 6 3 4 3 6 1 5 4 2 4 5 6 1 2 3 5 4 2 3 6 1 6 3 4 2 3 6 7 7 7 7 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 4 1 1 1 1 1 1 5 1 1 1 1 1 1 6 1 1 1 1 1 1 7 1 1 1 1 1 1 8 1 2 3 3 1 2 9 1 2 3 3 1 2	1 2	
6 1 2 3 4 5 6 2 1 5 6 3 4 3 6 1 5 4 2 4 5 6 1 2 3 5 4 2 3 6 1 6 3 4 2 1 5 7 1 2 3 4 5 6 7 2 1 1 1 1 1 3 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 3 1 2 3 3 1 2 0	2 1	
1 2 3 4 5 6 2 1 5 6 3 4 3 6 1 5 4 2 4 5 6 1 2 3 5 4 2 3 6 1 6 3 4 2 1 5 7 1 2 3 4 5 6 7 2 1 1 1 1 1 3 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 7 7 7 1 2 3 4 5 6 7 2 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 3 1 2 3 1 2 3 1 2 9	6	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	1 2 3 4 5 6	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$		
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	2 1 5 6 5 4	
4 5 6 1 2 3 5 4 2 3 6 1 6 3 4 2 1 5 7 7 7 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 4 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 3 1 2 3 3 1 2 3 1 2 3 4 2 4 6 1 2 3 4 4 4 4 4 4 4 4 4 4 4 4 4 <t< th=""><th>361542</th><th></th></t<>	361542	
5 4 2 3 6 1 6 3 4 2 1 5 7 7 7 7 1 1 1 2 1 1 1 1 1 3 1 1 1 1 1 4 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 3 1 2 3 3 1 3 1 2 3 1 2 8 1 2 3 1 2 9 1 2 1 1 1	4 5 6 1 2 3	
6 3 4 2 1 5 7 7 7 7 7 7 1 1 1 1 1 1 3 1 1 1 1 1 4 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 3 1 2 3 3 1 9	5 4 2 3 6 1	
7 1 2 3 4 5 6 7 2 1 1 1 1 1 3 1 1 1 1 1 4 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 8 1 2 3 1 2 3 1 2 3 1 2 9 1 2 1 1 1	6 3 4 2 1 5	
1 2 3 4 5 6 7 2 1 1 1 1 1 3 1 1 1 1 1 4 1 1 1 1 1 5 1 1 1 1 1 6 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 7 1 1 1 1 1 8 1 2 3 1 2 3 1 2 3 1 2 6 1 2 3 1 2 6 1 2 3 1 2 6 1 2 3 1 2 7 1 2 3	7	
2 1 1 1 1 3 1 1 1 1 4 1 1 1 1 5 1 1 1 1 6 1 1 1 1 7 1 1 1 1 3 1 1 1 1 3 1 1 1 1 3 1 2 3 1 3 1 2 1 1 4 2 3 1 2	1 2 3 4 5 6 7	
3 1 1 1 1 1 4 1 1 1 1 5 1 1 1 1 6 1 1 1 1 7 1 1 1 1 7 1 1 1 1 7 1 1 1 1 7 1 1 1 1 8 - - - 9 - - -	2 1 1 1 1 1 1	
4 1 1 1 1 5 1 1 1 1 6 1 1 1 1 7 1 1 1 1 7 1 1 1 1 7 1 1 1 1 7 1 1 1 1 3 1 1 1 1 3 1 2 3 1 3 1 2 1 1 0 1 1 1 1	3 1 1 1 1 1 1	
5 1 1 1 1 1 1 6 1 1 1 1 1 1 7 1 1 1 1 1 1 3 1 2 3 3 1 2 0	4 1 1 1 1 1 1	
6 1 1 1 1 1 1 7 1 1 1 1 1 3 1 2 3 3 1 2 3 1 2 0	5 1 1 1 1 1 1	
7 1 1 1 1 1 1 3 1 2 3 3 1 2 3 1 2 0	6 1 1 1 1 1 1	
3 1 2 3 3 1 2 3 1 2 0	7 1 1 1 1 1 1	
3 1 2 3 3 1 2 3 1 2 0		
1 2 3 3 1 2 0	3	
3 1 2 3 1 2 0	1 2 3	
3 1 2 0	3 1 2	
0	3 1 2	
	0	

Sample Output

yes			
yes			
no			
no			

Explanation

The first two collections of elements are in fact groups (that is, all properties are satisfied). For the third candidate, it is not a group, since $3 \times (2 \times 2) = 3 \times 1 = 3$ but $(3 \times 2) \times 2 = 1 \times 2 = 2$. In the last candidate, there is no identity,

since 1 is not the identity, since $2 \times 1 = 3$ (not 2), and 2 is not the identity, since $2 \times 1 = 3$ (not 1) and 3 is not the identity, since $1 \times 3 = 3$ (not 1).