The lottery BWS is played annually. In this lottery N people bet choosing K numbers each. In a formal way, we can say that B_{ij} is the j-th value bet by the i-th person. Then the organizers choose K positive integers. The chosen numbers are called W_1, W_2, $\ldots$, W_K.

The winners are calculated as followed:

- A non-empty subset is chosen randomly from the N participants; in other words, some participants are luckily chosen.
- For each person in this subset the value S_1 is calculated, the sum of all the first numbers bet by them, that is, the sum of the B_{i1} where i is the index of each chosen person. In the same way the values S_2, $\ldots$, S_K are calculated.
- A parity test between W_j and S_j is performed; in other words, it is verified if the parity (if a number is pair or odd) matches between W_1 and S_1, W_2 and S_2, and so on until W_K and S_K.
- If all parities match, then the people in this subset are considered the winners!

The organizers want to know: is it possible to pick the numbers W_1, W_2, $\ldots$, W_K so that there is **no** subset of winning participants?

Input

The input contains several test cases. The first line of a test case contains the numbers N ($1 \leq N \leq 30000$) and K ($3 \leq K \leq 50$), which represent the number of participants and the quantity of numbers bet by each person, respectively. The participants bet with integer numbers between 1 and 10^9, inclusive. Each of the next N lines contains K numbers representing the bet of each person, one person per line.

Output

For each test case in the input you must output a single line, containing one letter: 'S' in case it is possible or 'N' otherwise.

Sample Input

```
2 3
1 2 3
5 6 7
3 3
3 2 1
6 5 4
4 4 4
4 3
9 4 7
4 4 4
2 7 2
2 2 1
```

Sample Output

```
S
S
N
```