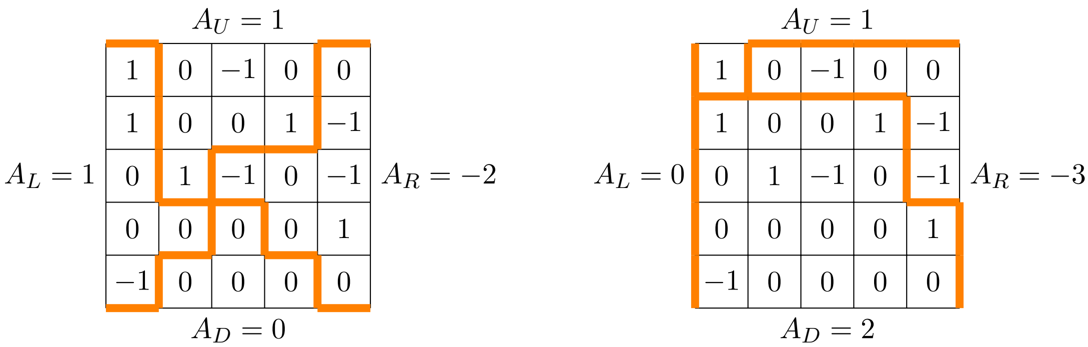
Division with Polylines

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

You are given a board of size $n \times n$, with each of its n^2 cells containing one of the numbers -1, 0, or 1. You are also given four integers A_L, A_D, A_R and A_U. We want to draw two polylines:

- one connecting the top-left corner of the board with the bottom-right corner; moving along the edges of the cells only to the right and down,
- and the other connecting the bottom-left corner of the board with the top-right corner, moving only to the right and up.

The two polylines divide the board into four (possibly empty) regions: left, down, right, and up. Check whether there exist two polylines, such that the cells in these four regions sum up to A_L, A_D, A_R and A_U, respectively. Print `TAK` (yes) or `NIE` (no) for each of the t independent test cases.



The figures show the same 5×5 board from the sample test.

The left figure shows a partition into regions with sums $(1, 0, -2, 1)$ (in the order A_L, A_D, A_R, A_U). The first polyline can be described by the word `RDDDRRDRDR`, and the second by `RURUURRUUR` (where `D`, `U`, and `R` denote moves along edges down, up, and right, respectively). For example, the right region contains 8 cells with a sum $A_R = 0 - 1 - 1 + 0 - 1 + 0 + 1 + 0 = -2$.

The right figure shows a partition into regions with sums $(0, 2, -3, 1)$. The left region is empty, so the sum of its cells is zero. The polylines can be described by the words `DRRRRDDRDD` and `UUUURURRRR`.

The same board cannot be partitioned into regions with sums $(-2, 0, 1, 1)$, i.e. $A_L = -2$, $A_D = 0$, $A_R = 1$, $A_U = 1$. Note that the same values, but in a different order, appear in the left figure – the order matters.

Input

The first line of input contains a single integer t ($1 \le t \le 100\,000$), denoting the number of test cases.

The first line of a test case contains five integers n, A_L, A_D, A_R, A_U ($1 \le n \le 2000$; $-n^2 \le A_L, A_D, A_R, A_U \le n^2$), denoting the side length of the board and the desired sums of the four regions.

Each of the next n lines of the test case describes the board; the i-th line contains n integers $a_{i1}, a_{i2}, \ldots, a_{in}$ ($a_{ij} \in \{-1, 0, 1\}$), describing the i-th row. The total sum of the board is equal to $A_L + A_D + A_R + A_U$.

The total sum of n^2 over all test cases does not exceed $4\,000\,000$.

Output

Output exactly t lines – for each test case output the word `TAK` (*yes*) if there exist polylines dividing the board into regions with the specified sums, or the word `NIE` (*no*) otherwise.

Example

standard input	standard output
5 5 1 0 -2 1 1 0 -1 0 0 1 0 0 1 -1 0 1 -1 0 -1 0 0 0 0 1 -1 0 0 0 0 5 0 2 -3 1 1 0 -1 0 0 1 0 0 1 -1 0 1 -1 0 -1 0 0 0 0 1 -1 0 0 0 0 5 -2 0 1 1 1 0 -1 0 0 1 0 0 1 -1 0 1 -1 0 -1 0 0 0 0 1 -1 0 0 0 0 2 1 -1 0 1 1 0 -1 1 1 0 0 0 0 0	TAK TAK NIE NIE TAK

Note

The first three test cases correspond to the examples shown in the statement. They contain the same board and queries for sums (A_L, A_D, A_R, A_U) in order:

1. $(1, 0, -2, 1)$, answer `TAK`
2. $(0, 2, -3, 1)$, answer `TAK`
3. $(-2, 0, 1, 1)$, answer `NIE`