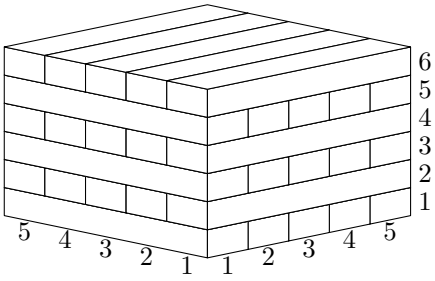
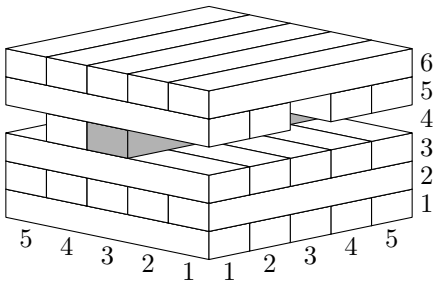
Problem J. Jenga Boom

Input file: `jenga.in`
Output file: `jenga.out`

Jane is a game designer and she designs the next version of Jenga Boom, where the blocks have dimensions of $1 \times w \times wn$ instead of the ordinary $1 \times 2 \times 6$. As usual, the initial tower is created at the game start. It consists of the blocks in levels of n blocks placed next to each other along their long sides and at a right angle to the previous level. Players remove blocks from the tower in turns, until the tower collapses.



Initial tower



The tower before collapse

Jane wants to build a game simulator that helps her to decide the best n and w. The simulator shall compute the moment when the tower collapses if blocks are removed in the specified order. Tower collapses if there is a cross-section between levels such that the center of the mass of the levels above it does not belong to or is at the edge of the convex hull of the previous level projection.

Input

The first line contains two integers n and w that define the dimensions of the block as described in the problem statement ($1 \le n, w \le 10\,000$). The second line also contains two integers: h — the number of levels in the tower and m — the number of removed blocks ($1 \le h, m \le 5\,000$).

The next m lines contain coordinates of the removed blocks with two integers each: l_i — the level of the block, counting from the bottom and k_i — the position of the block, counting from the edge nearest to the players ($1 \le l_i \le h$; $1 \le k_i \le n$). Blocks are removed one by one and no block is removed twice.

Output

In the first line output "`yes`" if the tower collapses, and "`no`" otherwise. In the former case, output the number of the block (starting from 1), that was removed just before the collapse, in the next line.

Examples

jenga.in	jenga.out
5 2 6 6 4 1 4 2 4 5 5 3 4 3 1 1	yes 5
3 3 10 1 10 3	no
2 2 2 1 1 1	yes 1