

Rectangle Cutting

Input file: **standard input**
Output file: **standard output**
Time limit: 3 seconds
Memory limit: 512 megabytes

Yachiyo has a rectangle in the first quadrant of the Cartesian coordinate plane. The lower-left corner of the rectangle is at the origin $(0,0)$, and the upper-right corner is at (n,m) . In other words, the two sides of the rectangle are aligned with the x -axis and the y -axis, with lengths n and m along the x - and y -directions, respectively.

Yachiyo decides to make q cuts on this rectangle. Specifically, there are two kinds of cuts:

- Horizontal cut: given a positive integer k , cut along the line $y = k$.
- Vertical cut: given a positive integer k , cut along the line $x = k$.

Each cut goes through the entire region, further dividing the existing rectangles into more smaller rectangles.

Yachiyo wants to know, after each cut is completed, what the maximum area among all the currently divided small rectangles is.

Input

The first line contains three integers n, m, q ($1 \leq n, m \leq 10^9, 1 \leq q \leq 5 \times 10^5$), representing the horizontal length, vertical length, and the number of cuts of the initial rectangle, respectively.

The next q lines each contain two integers op, k , describing one cut operation:

- If $op = 1$, it means a vertical cut is performed, and the given integer k ($1 \leq k < n$) indicates cutting along the line $x = k$.
- If $op = 2$, it means a horizontal cut is performed, and the given integer k ($1 \leq k < m$) indicates cutting along the line $y = k$.

It is guaranteed that Yachiyo will never cut at the same position twice (that is, all cutting lines are distinct).

Output

Output q lines. The i -th line should contain one integer, representing the maximum area among all small rectangles after the i -th cut.

Example

standard input	standard output
5 3 3	12
1 1	9
1 4	6
2 2	

Note

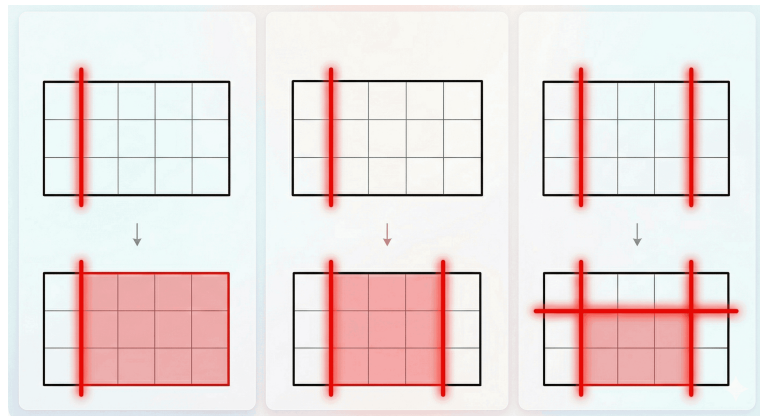


Рис. 1: *Sample explanation*