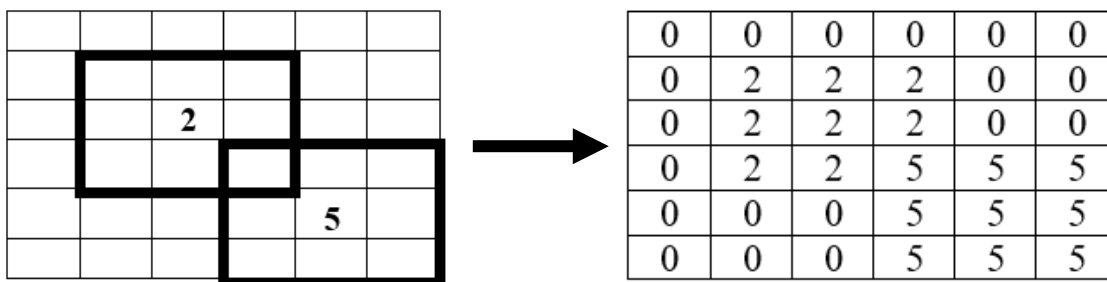


K	King Tide in a Cavern	
	Time Limit (seconds)	3
	Memory Limit (MB)	512

Deep within the cavern, dim light glints off the stone walls. Beyond lies the open sea, its level slowly rising. As water begins to push through narrow passages and hidden hollows, you must determine which parts of the cave will be claimed when the King Tide arrives.

You are given a rectangular cave viewed from above with integer **width W** and **height H** (a grid of $W \times H$ cells). The cave floor is defined by **R axis-aligned rectangles**. Each rectangle has a constant **height z** and covers an axis-aligned rectangular area of the grid. If rectangles overlap, the floor height at a cell is **the maximum z among all rectangles covering that cell** (as if heights are layered). Cells not covered by any rectangle have height 0.

For example, in a 6-by-6 cave below. If the upper-left rectangle has height 2, and the lower-right rectangle has height 5, the heights of each cell in this cave are as follows.



The area outside the cave is the sea. When the sea level is S , water enters the cave **from any outer boundary** and spreads to adjacent cells in the four cardinal directions (up, down, left, right). Water may move into a cell only if that cell's floor height $\leq S$. A cell is considered flooded if there exists a path from the outside boundary to that cell such that the maximum floor height along the path (the path's bottleneck) is $\leq S$.

Compute **the number of flooded cells** inside the cave for the given sea level S .

INPUT

The input consists of the following:

- The first line contains four integers W H R Q — the width and height of the cave, the number of rectangles R , and the number of queries Q .
- Each of the next R lines describes one rectangle with five integers $x1$ $y1$ $x2$ $y2$ z :
 - $(x1, y1)$ is the lower-left corner,
 - $(x2, y2)$ is the upper-right corner,
 - z is the height of the rectangle.
- The following Q lines each contain one integer S , the sea level for a query.

Coordinates are integers. Rectangles lie within the region $[0, W] \times [0, H]$ and may touch or overlap.

OUTPUT

For each query sea level S , output a single line with the total number of grid cells inside the cave that are flooded.

CONSTRAINTS

$$1 \leq R \leq 3 \times 10^5$$

$$1 \leq Q \leq 3 \times 10^5$$

$$1 \leq W, H \leq 10^9$$

$$0 \leq x1 < x2 \leq W$$

$$0 \leq y1 < y2 \leq H$$

$$0 \leq z \leq 10^9$$

$$0 \leq S \leq 10^9$$

Input	Output
5 4 2 3 0 0 3 2 1 2 1 5 4 2 0 1 2	6 11 20
6 3 2 3 0 0 2 3 1 4 0 6 3 3 0 2 3	6 12 18

EXPLANATION

For the sample case 1:

The cave is a grid of width 5 and height 4 → 20 cells total.

- **Rectangle 1:** from (0,0) to (3,2), height = 1.
- **Rectangle 2:** from (2,1) to (5,4), height = 2.
 - These rectangles overlap at cells (2,1) and (2,2), where the effective height = 2.
- All other cells not covered by any rectangle have height = 0.

Now check each query:

1. **S = 0:** all 0-height cells that touch the boundary flood → 6.
2. **S = 1:** 0-height + height-1 regions on the boundary flood → 6 + 5 = 11.
3. **S = 2:** max height is 2 everywhere reachable from edges → entire grid 20.

For the sample case 2:

The cave is $6 \times 3 = 18$ cells in total.

- **Rectangle 1:** from (0,0) to (2,3), height = 1.
- **Rectangle 2:** from (4,0) to (6,3), height = 3.
- No overlap — there's a gap of width 2 between the two regions.
- Cells not covered by rectangles have height = 0.

Queries:

1. **S = 0:** the middle strip touches the outer boundary → 6 flood.
2. **S = 2:** middle strip (0) + left block (1) both touch boundary → 6 + 6 = 12.
3. **S = 3:** all regions (including right block 3) are boundary-connected → 18 total.