Sweet Sugar IV

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

To celebrate the coming Children's Day, a sugar collection game will be hosted. The game is played on a grid map of $n \times m$ cells. The rows and columns of the grid are numbered from 1 to n and 1 to m, respectively.

There are d pieces of sugar on the map. The i-th piece of sugar is located at the cell (x_i, y_i), the sweetness of which is s_i. Every cell contains at most one piece of sugar.

There are k pawns on the map. Initially, all the pawns are located at the cell (S_x, S_y). In each operation, the player can move a pawn from (x, y) to $(x + 1, y)$ or $(x, y + 1)$. Note that a cell can contain multiple pawns. Whenever a pawn is located at a cell containing a piece of sugar, the player will take that sugar, and of course, each piece of sugar can not be taken more than once. The player can do legal operations for an arbitrary number of times. The final score of the player is equal to the sum of sweetness among all the pieces of taken sugar.

There are also p rectangular obstacles on the map. Assume the i-th rectangle is $[x_1, x_2] \times [y_1, y_2]$, then a cell (x, y) is considered to be dangerous if and only if $x_1 \le x \le x_2$ and $y_1 \le y \le y_2$. The player should never move the pawns into any dangerous cell. Fortunately:

- The edge of the map will never be dangerous (i.e. $2 \le x_1 \le x_2 \le n - 1$, $2 \le y_1 \le y_2 \le m - 1$).
- No two obstacles will overlap or touch each other. In other words, two cells (x, y) and (x', y') are considered to be adjacent if and only if $\max\{|x-x'|, |y-y'|\} \le 1$. If there are two adjacent dangerous cells, they must belong to the same obstacle.

You are now training for this sugar collection game. Given all the information about the game, try to find the maximum possible value of the final score that you can achieve. To become the master of this game, you will replay the game for q times, each time the location of the starting cell (S_x, S_y) and the number of pawns k may be changed.

Input

The first line contains five integers n, m, p, d and q ($1 \le n, m \le 10^5$, $n \times m \le 10^6$, $0 \le p \le 10^5$, $1 \le d \le 2 \cdot 10^5$, $1 \le q \le 3 \cdot 10^5$), denoting the size of the map, the number of obstacles, the number of pieces of sugar, and the number of replay times, respectively.

Each of the following p lines contains four integers x_1, y_1, x_2 and y_2 ($2 \le x_1 \le x_2 \le n - 1$, $2 \le y_1 \le y_2 \le m - 1$), describing each rectangular obstacle. It is guaranteed that no two obstacles will overlap or touch each other.

Each of the following d lines contains three integers x_i, y_i and s_i ($1 \le x_i \le n$, $1 \le y_i \le m$, $1 \le s_i \le 10^9$), describing each piece of sugar. It is guaranteed that every cell contains at most one piece of sugar, and no piece of sugar will be located at a dangerous cell.

Each of the following q lines contains three integers S_x, S_y and k ($1 \le S_x \le n$, $1 \le S_y \le m$, $1 \le k \le 5$), denoting the location of the starting cell and the number of pawns in a replay. It is guaranteed that (S_x, S_y) is safe.

Output

Output q lines, the i-th ($1 \le i \le q$) of which containing an integer, denoting the maximum value of the final score that you can achieve in the i-th replay.

Examples

standard input	standard output
3 3 1 4 5 2 2 2 2 1 1 1 3 3 1 1 2 3 2 1 5 1 1 1 1 1 2 1 3 2 1 2 2 2 1 1	7 10 1 4 6
5 7 1 3 4 2 3 3 4 3 1 1 2 2 10 5 4 100 1 2 1 1 2 2 1 1 2 2 7 1	110 110 111 0