

(Adv.) Competitive Programming

Submit until 19.07.2019 13:30, via the [judge interface](#)



Problem: anartistwall (6 second timelimit)

This time you are to help a charismatic, old artist who is famous for his colourful paintings. Every time he has a creative phase, he nails a rectangular canvas on the big wall located behind his house and starts creating another colourful masterpiece. (Note that the canvas is always mounted level on the wall — this means that the upper and the lower side are parallel to the floor.)

Over the years many nails remained in the wall and sometimes he has to search a while before he finds an area big enough for the canvas he wants to use. But queer as artists sometimes are, he does not want to remove any nails from the wall.

Recently, a rich investment banker from Switzerland visited the old artist and asked him to create a painting for his new villa. He said that the canvas must be rectangular, but also made it clear that there are no further restrictions concerning the size of the painting.

At the end of their conversation they argued about payment and decided that the price should depend on the size of the painting.

Since the old artist needs every cent, he wants to create the biggest painting possible. Please help him to find the largest canvas size that he can fit on his wall without removing any of his beloved nails.

Input The first line of the input contains t ($1 \leq t \leq 100$), the number of test cases. Each test case starts with a line containing w, h ($0 \leq w, h \leq 500$), the width and height of the artist's wall, and n ($0 \leq n \leq (w + 1) \cdot (h + 1)$), the number of nails already located on the wall. Each of the next n lines contains x_i and y_i , the integer coordinates of the i -th nail ($0 \leq x_i \leq w, 0 \leq y_i \leq h$). For each test case, no two nails are located at the same coordinates.

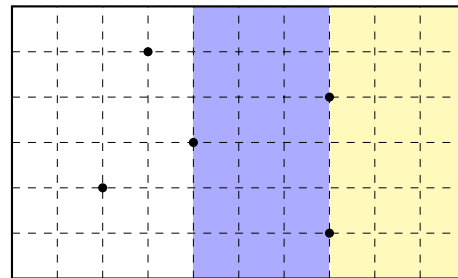
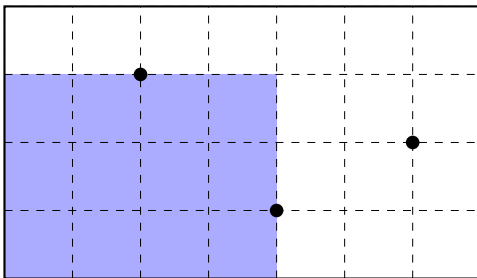
Output For each test case output a line containing a single integer, the area of the biggest canvas the artist can create on his wall.

Sample input

```
2
7 4 3
4 1
6 2
2 3
10 6 5
7 1
2 2
4 3
7 4
3 5
```

Sample output

```
12
18
```



Visualization of the two samples