The world is in great danger!! **Mental's forces** have returned to Earth to eradicate humankind. Our last hope to stop this great evil is **Sam "Serious" Stone**. Equipped with various powerful weapons, Serious Sam starts his mission to destroy the forces of evil.

After fighting two days and three nights, Sam is now in front of the temple KOPTOS where Mental's general Ugh Zan III is waiting for him. But this time, he has a serious problem. He is in shortage of ammo and a lot of enemies crawling inside the temple waiting for him. After rounding the temple Sam finds that the temple is in rectangle shape and he has the locations of all enemies in the temple.
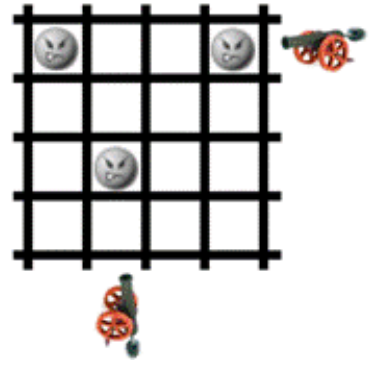
All of a sudden he realizes that he can kill the enemies without entering the temple using the great cannon ball which spits out a gigantic ball bigger than him killing anything it runs into and keeps on rolling until it finally explodes. But the cannonball can only shoot horizontally or vertically and all the enemies along the path of that cannon ball will be killed.

Now he wants to save as many cannon balls as possible for fighting with Mental. So, he wants to know the minimum number of cannon balls and the positions from which he can shoot the cannonballs to eliminate all enemies from outside that temple.

Input

The input file contains several test cases.

Here, the temple is defined as a $R \times C$ grid. The first line of each test case contains 3 integers: R ($0 < R < 1001$), C ($0 < C < 1001$) representing the grid of temple (R means number of row and C means number of column of the grid) and the number of enemies N ($0 < N < 1000001$) inside the temple. After that there are N lines each of which contains 2 integers representing the position of the enemies in that temple. Each test case is followed by a new line (except the last one). Input is terminated when $R = C = N = 0$.

Output

For each test case there will be one line output. First print the minimum number (m) of cannonballs needed to wipe out the enemies followed by a single space and then m positions from which he can shoot those cannonballs. For shooting horizontally print 'r' followed by the row number and for vertical shooting print 'c' followed by the column number. If there is more than one solution any one will do.

Sample Input

```
4 4 3
1 1
1 4
3 2

4 4 2
1 1
2 2

0 0 0
```

Sample Output

```
2 r1 r3
2 r1 r2
```