

ABC176

E

题意：给定一个矩阵，并且标了一些点，现在你可以选择矩阵上的任何一行和一系列，摧毁这一行以及这一列上的所有目标点，问最多能摧毁多少个。（矩阵最大边长 $3e5$ 目标点不超过 $3e5$ 个）

题解：显然，最大值应该取在最多的那一行和最多的那一列，但是如果有相同的，我们要找交叉点不是目标点的那个组合（显然如果是目标点，那么ans-1可以直接暴力取最多的行和最多的列。看似是 n^2 复杂度，但是由于目标点最多只有 $3e5$ 个，所以最多找 $3e5$ 个组合就能找到一个交叉点不是目标点的组合。

```
#include <algorithm>
#include <cstdio>
#include <cstring>
#include <iostream>
#include <map>
#include <queue>
using namespace std;
const int MAX = 3e5 + 20;
typedef long long ll;
struct ppp{
    int val, id;
    bool operator < (ppp a) const {
        return val > a.val;
    }
};
ppp hang[MAX], lie[MAX];
map <int,int> pic[MAX];
int main()
{
    int h, w, m;
    cin >> h >> w >> m;
    for (int i = 1; i <= m; i++)
    {
        int x, y;
        scanf("%d%d", &x, &y);
        pic[x][y] = 1;
        hang[x].val++, lie[y].val++;
    }
    for (int i = 1; i <= max(h, w); i++)
        hang[i].id = i, lie[i].id = i;
    sort(hang + 1, hang + 1 + h);
    sort(lie + 1, lie + 1 + w);
    int ans = hang[1].val + lie[1].val - 1;
    {
        for (int i = 1; i <= h; i++)
        {
```

```
    if (i != 1 and hang[i].val != hang[i-1].val)
        break;
    for (int j = 1; j <= w; j++)
    {
        if (j!=1 and lie[j].val != lie[j-1].val)
            break;
        if(pic[hang[i].id].count(lie[j].id) == 0)
        {
            ans++;
            cout << ans << endl;
            return 0;
        }
    }
}
cout << ans << endl;
return 0;
}
```

From:

<https://wiki.cvbbacm.com/> - **CVBB ACM Team**

Permanent link:

https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:too_low:abc176cy



Last update: **2020/08/28 17:23**