

## 一、编程题

### ACM:分配土地

#### 题目描述：

从前有个村庄，村民们喜欢在各种田地上插上小旗子，旗子上标识了各种不同的数字。某天集体村民决定将覆盖相同数字的最小矩阵形的土地的分配给为村里做出巨大贡献的村民，请问，此次分配土地，做出贡献的村民中最大会分配多大面积？

#### 输入描述：

第一行输入  $m$  和  $n$ ， $m$  代表村子的土地的长， $n$  代表土地的宽

第二行开始输入地图上的具体标识

#### 输出描述：

输出需要分配的土地面积，即包含相同数字旗子的最小矩阵中的最大面积。

#### 补充说明：

#### 示例 1

##### 输入：

```
3 3
1 0 1
0 0 0
0 1 0
```

##### 输出：

```
9
```

##### 说明：

土地上的旗子为 1，其坐标分别为(0,0)，(2,1)以及(0,2)，为了覆盖所有旗子，矩阵需要覆盖的横坐标为 0 和 2，纵坐标为 0 和 2，所以面积为 9，即 $(2-0+1)*(2-0+1)=9$ 。

#### 示例 2

##### 输入：

```
3 3
1 0 2
0 0 0
0 3 4
```

##### 输出：

```
1
```

##### 说明：

由于不存在成对的小旗子，故而返回 1，即一块土地的面积。

#### 代码：

```
import sys
```

```
line = sys.stdin.readline().split()
```

```
m = int(line[0])
```

```
n = int(line[1])
```

```
tudi = {} # [x1, y1, x2, y2]
```

```

def change_point(x, y, point):
    if x < point[0]:
        point[0] = x
    elif x > point[2]:
        point[2] = x
    if y < point[1]:
        point[1] = y
    elif y > point[3]:
        point[3] = y

for i in range(m):
    line = sys.stdin.readline().split()
    line = [int(l) for l in line]
    for j in range(n):
        if line[j] == 0: continue
        if line[j] not in tudi:
            tudi[line[j]] = [j, i, j, i]
        else:
            change_point(j, i, tudi[line[j]])

max_area = 1
for key, value in tudi.items():
    max_area = max(max_area, (value[3] - value[1] + 1) * (value[2] - value[0] + 1))

print(max_area)

```