

# marshland

(*marshland.cpp/c/pas*)

## 【问题描述】

前方有一片沼泽地.

方便地, 我们用一个  $n \times n$  的网格图来描述它, 每一个格子代表着沼泽地的一小片区域. 其中  $(1, 1)$  代表网格图的左上角,  $(n, n)$  代表网格图的右下角. 若用  $X$  表示行数,  $Y$  表示列数, 那么  $X + Y$  为奇数的格子有一个危险度  $V_{X,Y}$ ,  $X + Y$  为偶数的格子的危险度为 0.

为了保障人们的安全, 你有  $m$  个长相怪异的大石头, 你可以选一些石头放在网格图上的某些格子上, 石头可以看成是一个 ‘L’ 形的块, 并且占三个格子, 它通过旋转有四种方式供放置, 仅会使得在拐角处的那个格子危险度减为 0.

网格图中还有  $k$  个位置是“禁止位置”, 石头的任何部位都不能位于这些格子上, 且这些位置的危险度一定为 0.

现在你需要知道放置一些石头后最小的危险度之和是多少. (石头可以不放完)

## 【输入格式】

从文件 *marshland.in* 中读入数据.

第一行三个整数  $n, m, k$ .

接下来  $n$  行每行  $n$  个整数, 表示每个格子的危险度, 保证  $X + Y$  为偶数的格子和禁止位置的格子的危险度为 0.

接下来  $k$  行每行 2 个整数  $X, Y$ , 表示禁止位置的坐标, 注意有可能会出现重复的禁止位置.

## 【输出格式】

输出到文件 *marshland.out* 中

输出一行一个整数代表最小的危险度之和.

## 【样例输入 1】

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

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

**【样例输出 1】**

3

**【样例输入 2】**

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

**【样例输出 2】**

9

**【样例输入 3】**

见选手目录下的 *marshland3.in*.

**【样例输出 3】**

见选手目录下的 *marshland3.out*.

**【数据规模和约定】**

对于 10% 的数据, 满足  $n \leq 4$ ,

对于 30% 的数据, 满足  $n \leq 10$ ,

对于 100% 的数据, 满足  $n \leq 50$ ,

对于所有的数据, 满足  $0 \leq m \leq \frac{n^2}{3}, 0 \leq k \leq n^2, 0 \leq V_{X,Y} \leq 10^6$ .