

玩具兵

【问题描述】

小明的爸爸给他买了一盒玩具兵，其中有 K 个步兵， K 个骑兵和一个天兵，个个高大威猛，形象逼真。盒子里还有一个 $M*N$ 棋盘，每个格子 (i,j) 都有一个高度 H_{ij} ，并且大得足以容纳所有的玩具兵。小明把所有的玩具兵都放到棋盘上去，突然想到了一种很有趣的玩法：任意挑选 T 个不同的格子，并给每个格子 i 规定一个重要值 R_i ，游戏的目标就是每次沿东南西北之一的方向把一个玩具兵移动到其相邻的格子中（但不能移动到棋盘外面去），最终使得每个挑选出的格子 i 上恰好有 R_i 个玩具兵。小明希望所有的玩具兵都在某个选定的格子中，因此他总是使选出的 T 个格子的重要值之和等于玩具兵的个数。为了增加难度，小明给玩具兵们的移动方式做了一些规定：

- 步兵只会往高处爬，因此如果两个格子 A 和 B 相邻，当且仅当格子 A 的高度小于或等于 B ，步兵才可以从 A 移动到 B 。
- 骑兵只会往低处跳，因此如果两个格子 A 和 B 相邻，当且仅当格子 A 的高度大于或等于 B ，骑兵才可以从 A 移动到 B 。
- 天兵技术全面，移动不受任何限制。

可是没玩几次，小明就发现这个游戏太难了，他常常玩了好半天也达不到目的。于是，他设计了一种“超能力”，每使用一次超能力的时候，虽然不能移动任何一个玩具兵，但可对它们进行任意多次交换操作，每次交换两个玩具兵。等这次超能力使用完后又可和平常一样继续移动这些玩具兵。借助强大的超能力，这个游戏是容易玩通的，但是怎样才能让使用超能力的次数最少呢？

【输入文件】

输入文件 `toy.in` 的第一行包含四个整数： M,N,K,T ($2 \leq M,N \leq 100$, $1 \leq K \leq 50$, $1 \leq T \leq 2K+1$)

第二行包括 $2K+1$ 个数对 (x_i, y_i) ，代表各个玩具兵的初始位置。前 K 个代表步兵，接下来的 K 个代表骑兵，最后一个代表天兵。

第三行包含 T 个三元组 (x_i, y_i, r_i) ，第 i 组代表第 i 个目标格的位置和重要值。

以下 M 行，每行 N 个整数。其中第 i 行第 j 个数为即格子的高度 H_{ij} 。高度是不超过 100 的正整数，注意：不同玩具兵的初始位置可能相同。输入数据保证无错，选定的 T 个格子的重要值之和保证等于 $2K+1$ 。

【输出文件】

输出文件 `toy.out` 仅包含一行，即使用超能力的最小次数 T 。

【输入输出样例】

toy.in
4 6 2 5
1 1 1 5 4 1 4 5 3 3
1 2 1 2 6 1 3 2 1 3 6 1 4 3 1
3 2 6 1 3 5
2 1 7 4 4 6
2 3 1 4 3 4
4 3 4 3 2 3

toy.out
1