

题解

题目算法摘要

T1 : 贪心

T2 : 最短路

T3 : 线段树/分块

T4 : DP

T1 : rush

一道很简单的题目。

30 分做法

指数级暴力。

60 分做法

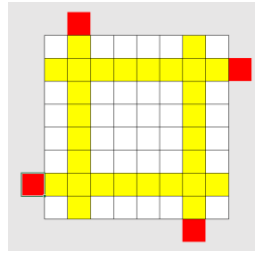
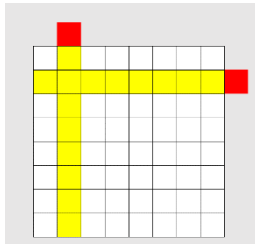
我们注意到，如果某个格子是空的，那么所在行列都不能放莱因哈特了。

而且一行只能放一个，一列也只能放一个。那么我们思考，相交的情况呢？

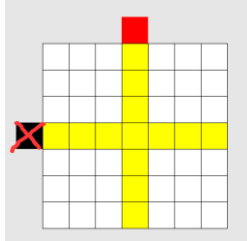
事实上，如果一行一列交叉，一定有一种方法可以使这行和这列各放一个，且不会影响其它行列。

比如：

哪怕是四个冲突也没有关系：



唯一需要特判的是 n 为奇数，正中心一行一列冲突，这种情况下答案减 1 就好。



如果暴力记录每个格子的情况，空间复杂度 $O(n^2)$ ，可以拿到 60 分。

100 分做法

我们注意到， n 与 m 同级，也就是说训练场的空地是稀疏的，那么我们想到，直接标记空格所在行列，空间复杂度就降到了 $O(n)$ ，也可以通过全部数据了。

T2 : bomb

这题题面写得很鬼，其实是很水的一道题目，难度比第二题低得多。一般对最短路问题比较熟练的同学很容易做出来。

0 分做法

输出 "Mission Failed"。

20~50 分做法

我也不知道，乱给的分数。拿到这种分数上来讲。

100 分做法

设 $d1[x], d2[x]$ 为城市 x 的到达时间，可进入时间；则 $\max(d1[x], d2[x])$ 为真实的进入时间； $d[x]$ 记录城市 x 被多少个城市保护。

每次堆中取出一个真实进入时间最小的城市，更新它所通往的城市的 $d1$ ，保护城市的 $d2$ ，保护城市的 $d--$ 。

若 $d=0$ ，则可入堆。

没什么难度。复杂度……Dij 复杂度。

T3 : noon

这道题比较难，但是暴力分给的很足，只要稍加思考就能拿到 50 分的暴力分，而 100 分有些难度，需要一定的思考。

30 分做法

首先我们分析题目，能看到的意思就是敌人所在的这一点的与原点连线的斜率严格大于之前任何一个点的斜率。那么我们可以直接用斜率作为该点的权值。则所求问题转化为，统计权值序列中严格大于之前所有权值的点个数。

在 30% 的数据中， n, m 较小，因此可以暴力修改暴力查询，即每次记录前缀最大值，如果当前值 $> \text{Max}$ ，则 $\text{ans}++$ ， $\text{Max} = \text{Now}$ 。复杂度 $O(nm)$ 。

50 分做法

在 20% 的数据中，修改的 x 单调递增。因此，前缀最大值可以保存下来不断重复使用，这样就只需要 $O(m)$ 的线性扫一遍就可以了。

结合 30 分做法可以拿到 50 分。

这两个暴力太好写了，因此本题暴力分只有 50。

100 分做法

方法 1

显然很容易发现，当我们修改位置 x 时， $[0, x-1]$ 中的敌人可视情况不会发生改变。也就是说，我们只要维护修改部分之后的可视情况即可。

原本维护的复杂度是 $O(n)$ 的，我们思考能不能将这个复杂度减小。

极容易想到一种做法——分块。

我们把所有敌人分成 \sqrt{n} 块，每个块内记录最大值和可视序列（显然这个序列是有序的），每次修改某个位置的时候，暴力在块内重建可视序列，然后从重构的块开始，利用之前块的前缀最大值，在块内二分查找到第一个大于前缀最大值的位置，然后将这个位置和块内后面的可视数加入答案，并更新前缀最大值。

总的复杂度 $O(m\sqrt{n}\log\sqrt{n})$ 。 10^5 的数据优化常数可以通过。

方法 2

上面的分块的复杂度如果常数写得大，很容易被卡掉。因此我们需要一个更优秀的复杂度。

显然，由于块内其实是不需要排序的（本来只要维护有序的那一部分），因此，分块能完成的事情，线段树也能相似的完成。

于是我们考虑记录这样两个东西—— \max, cnt ，分别表示区间最大值和区间内可视个数。那么答案就是根节点的 cnt 。

现在我们考虑怎么合并两个区间，显然左区间的 cnt 肯定计入父区间的 cnt ，现在唯一要解决的就是右区间有多少计入答案。

显然，如果右区间的 \max 小于左区间的 \max 那么就不用计入答案了，否则考虑右区间的两个子区间。

假设左子区间的 \max 和 cnt 分别为 Lm, Lt ，右子区间的 \max 和 cnt 分别为 Rm, Rt ，左区间的 \max 为 M ，右区间的 cnt 为 C ，分两种情况讨论：

1. 若 $Lm < M$ ，则对右子区间递归处理；
2. 若 $Lm > M$ ，则右子区间的可视部分会全部计入答案，即对左子区间递归处理后加上 $C - Lt$ 。

两种情况递归最多 $\log n$ 层，因此总的复杂度为 $O(m \log^2 n)$ ，完美通过所有数据，比分块要快些。

T4 : pool

这道题没什么难度，最直接的 DP。

100 分做法

用 $f[i][s] (i=1 \sim n, s=0,1)$ 表示耗时为 i ，最后一次为短点(0)或长点(1)。

很容易得到：

$$f[i][0] = f[i-1][0] + f[i-1][1];$$

$$f[i][1] = f[i-k][0];$$

最后的答案统计前缀和并输出即可。

迅捷PDF编辑器

迅捷PDF编辑器