NOI练习赛 解题报告

d646d58aa3d9c7b7b03a97885d5d5e8d

1 找不到

1.1 15分算法

对于$n \leq 1$的情况，如果这个矩阵是$\begin{bmatrix} 0 \end{bmatrix}$就输出$\begin{bmatrix} 1 \end{bmatrix}$，反之输出$\begin{bmatrix} 0 \end{bmatrix}$。

（其实根据数据范围你会发现，唯一的元素只有可能是1，那么答案就是0）

对于$n \leq 2$的情况，如果这个矩阵全是0或全是1，那么如上输出；否则输出$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$。

对于$n \leq 3$的情况，答案矩阵的大小显然不会超过2，各种判一判就好了。

总之纯乱搞的话是可以有15分的。

1.2 50分算法

注意到答案矩阵应该不会很大。规模为$m \times m$的0/1矩阵共有2^{m^2}个，而$n \times n$矩阵的规模为$m \times m$子矩阵共有$(n+1-m)^2$个，考虑旋转和

翻转也不会超过$8(n+1-m)^2$个。解这个不等关系，可以得到m大概在$O\left(\sqrt{\log n}\right)$级别。

那么，我们只需从小到大枚举m，然后在其中按字典序枚举所有可能的答案矩阵，暴力代入检验即可。易得其时间复杂度为$O\left(n^4\right)$。

由于数据的性质，这种做法的常数实际上非常小。根据常数期望得分20～80分。

1.3 100分算法

事实上，在枚举出m以后，可以将A矩阵中的所有$m\times m$子矩阵（包括旋转和翻转后的）放到一个字典树里，然后遍历字典树的每个叶子就能找到字典序最小的未出现矩阵，或得出所有$m\times m$矩阵均已出现过。

这种算法的时间复杂度为$O\left(n^2\log n\right)$，可以通过全部测试数据。

2 二进制通信

2.1 60分算法

首先$n!$枚举发送的数的顺序。

枚举出顺序以后，对于每个数，要么用方式0发送，要么用方式1，选取之前发送过的与当前数最接近的数发送。贪心选取较优的即可。

时间复杂度$O(n!n^2)$。期望得分：60。

2.2 100分算法

建图。

对于每个数建一个点，另外新建0号点，与每一个其他的点连边，边权为65（方式0的代价）。

对于每两个数连一条边，边权为16+6×它们不同的位数（方式1的代价），求最小生成树即可。

时间复杂度$O(n^2)$，期望得分100。

3 木乃伊危机

3.1 问题转化

显然的思路是搜索，然而本题的数据规模较大，这种非多项式算法必然不可行。

本题显然满足单调性。即如果a是问题的答案，则$a-1$也是问题的答案，故可以以一个log 的代价，将其转化为判定性问题：是否存在一种方案，使得经过a个回合之后，你不与任何一个木乃伊相遇。

在a个回合之后，每个木乃伊所能控制的范围是以其初始位置为中心，边长为$2a+1$的正方形；你所能到达的范围是以原点为中心，边长为$2a+1$的正方形，则："你的范围中存在一个格子，它不在任何一个木乃伊的范围中"，等价于"存在一种方案，你能够不与任何一个木乃伊相遇"。

正推的结论是显然的：你只需要向这个格子走去，假设中途你能与木乃伊相遇，则木乃伊就能够在a个回合到达这个格子，与前提矛盾。反推的结论也是显然的：你与每个木乃伊的欧几里得距离不会变大，所以你的最终位置一定在任何一个木乃伊的范围之外。

3.2 20分算法

根据上面的分析，我们只需要判断，你的边长为$2a+1$的正方形范围是否被n个这样的正方形完全覆盖。显然的思路是开二维数组模拟，暴力填充。注意地图的范围大约是$10^6 \times 10^6$的，所以必须先对坐标进行离散化，使坐标范围变为$O(n)$的。对于n个正方形中的每一个，我们都需要$O(n^2)$的时间进行填充，乘上二分答案的log，总的时间复杂度为$O(n^3 \log n)$。另外

一种模拟的思路是，枚举你的范围中的每一个格子，判断是否在任何一个木乃伊的范围当中。同理，这种算法的时间复杂度也是$O(n^3 \log n)$。

3.3 100分算法

考虑上面的模拟思路，它的实质是在二维数组上维护一个数据结构。修改：二维区间加上1；查询：求全局最小值。查询只有一次，且在全部修改之后。直接用二维线段树维护，平均情况下的复杂度是$O(n^2 \log^2 n)$，难以承受10^5的数据规模。然而，查询的特点能够使人立刻联想到降维处理。（或者说，扫描线）

将每个正方形的纵向边按照x坐标排序，而在y坐标方向开一维数组。从左向右依次扫描每条纵向边，如果当前扫到的是左边界，则在对应的y坐标上进行区间加1；扫到的是右边界，则在对应的y坐标上进行区间减1。只要当前的横坐标在你的正方形范围之中，且当前列的数组中最小值为0，就说明全局的最小值为0。

现在，问题转化为，在一维数组上维护一个数据结构，支持区间增减一个数和求全局最小值。暴力维护的时间复杂度为$O(n^2)$，用线段树维护的时间复杂度为$O(n \log n)$，乘上二分答案的log，总的时间复杂度为$O(n \log^2 n)$，足以通过测试数据。

本题的坐标是离散的方格，所以在细节上需注意+1或?1；用线段树的方法，也可以不写离散化，同样能够通过全部的测试数据。