A Solution

By Fun_S. (Original)

Subtask 1

你觉得呢？你觉得呢？你觉得呢？

初始就好了，输出 0。

Subtask 2

模拟！模拟！模拟！

还是很简单的，而且模拟过程中发现了结论（

Subtask 3

搜索！搜索！搜索！

随机生成，手玩发现大概率无解，果断输出 -1。

Subtask 4

结论是 n 为奇数是答案是一个二次式，偶数时无解。

Subtask 6

发现选相邻三个翻转，本质上就是交换第一个和第三个。

于是发现奇数位置的始终在奇数位置，偶数位置的始终在偶数位置。

所以奇数位置上必须全是奇数，偶数位置上全是偶数。

否则无解。如果有解，则按题意模拟即可。

Solution

发现题目其实就是把奇偶位置分开成两个序列看，然后一次操作可以交换任意一个序列中的相邻两个数。

这就是求两个数列的逆序对。

归并或者树状数组都行。

B Solution

By Qrj & Mt_F. (Unoriginal)

测试点1-2

对于每一次询问直接 `for` 循环遍历求 `gcd` 即可。

测试点3-4

分块，可以把整个矩阵分成若干个 $\sqrt{n} \times \sqrt{m}$ 的小矩阵，预处理求出每个小矩阵的 $\gcd$，对于每一次询问，整块直接 $O(1)$，散块一个一个遍历，遍历的散块大小在 $O(\sqrt{n} \times \sqrt{m})$ 级别，所以最终复杂度 $O(q\sqrt{n}\sqrt{m})$。

但考虑只这么做是假的。

如果一个散块有 $\sqrt{n}-1$ 行但是有 m 列，这样直接遍历时间复杂度是 $m\sqrt{n}$，所以对于散块不能直接一个一个枚举。所以还要将每一列每一行都预处理分块，相当于一个 $1 \times \sqrt{m}$ 或 $\sqrt{n} \times 1$ 的矩阵。这样就把原来 $O(m\sqrt{n})$ 的散块遍历再除了一个 $O(\sqrt{m})$，这样复杂度才对。

总结：1. 按$\sqrt{n} \times \sqrt{m}$二维分块。 2. 散块一维分块。

现在看来这个复杂度可以卡过 100 分，但是发现一般散块有四个或更多并且分块带有一定常数，~~所以AC 还是很难，除非精细实现加卡常。~~出题人常数大，相信各位写了是能过的。

测试点5-10

做法一

st表，预处理二维st表，对于每次询问，直接把询问的矩阵分成四个大块，这样就可以 $O(1)$ 查询。

复杂度瓶颈在预处理 $O(nmlognlogm)$。

做法二

二维线段树。

假做法三

虽然是假做法，但是出题人没有卡死，所以期望得分50。

考虑矩阵一大，最后得到 $gcd=1$ 的概率就很大，所以有些人会考虑全输出1，但期望得分0。

所以考虑在矩阵中随便选100个数来求gcd，最后这100个数的gcd有很大的概率等于整个矩阵的gcd.

期望得分随机（凭阳寿）。

虽然正确性一眼假，但是挺有意思的。况且出题人给了温馨提示，如果实在没做出来，建议放手一搏。

C Solution

By HOOC. (Unoriginal)

$10pts$:

随便暴力，直接 dfs 即可

$k=1$:

跑“最长路”或者 Dp，维护一下边权最大值

特殊性质：

此时原图构成一个 DAG （其实相当于正解的提示）

发现直接求路径找最大的第 k 长边是比较麻烦的，可以考虑二分边权然后判定是否可取。

将 $\geq$ 该值的边权记为 1，$<$ 的边权记为 0，假设当前二分到的值为 val，令 $f[v]$ 表示从 1 到点 v 的路径中边权值 $\geq val$ 的边数，则累加转换后的边权即为 $f[v]$ 的值。Dp 求出 $f[n]$ 后，如果大于等于 val 的边数 $\geq k$，即说明该边权可能是答案，继续二分找是否有更大的合法 val 即可。

跑 Dp 时可以用拓扑排序或者双端队列 bfs，复杂度 $O(m\log k)$

正解：

首先，如果你清楚 SCC 的定义及其性质的话，可以想到只需要对原图跑一遍 $Tarjan$ 缩点建图，就把原来有环的图变成了一个 DAG（由于可以重复经过一条边且它的贡献可以被多次计算，所以一条边的排名可以通过不停的在 SCC 里跑来改变），复杂度 $O(n+m)$。

然后就照着特殊性质的做法打一遍就可以了

当然，SCC 缩点时还可以存一下该 SCC 中的最大边权，如果其中的最大边权 $\geq val$，并且该 SCC 在 1 到 n 的一条路径上，那么可以直接判定该 val 是可取的

总复杂度为 $O(m\log k)$。

D Solution

By Sixy. (Original)

首先这个机器是一个树状数组，依照要求把树建出来，可知这个树上最多有 $m\log n$ 个节点，所以可以动态开点，压缩空间。

在这颗树上跑概率期望的背包问题。具体地，设 $f_{i,j}$ 代表第 i 个节点的和是 j 的概率，那递推式是
$f_{u,i+j-k}+=f_{u,i}\times f_{v,j}\times C(j,k)\times(pj)^{j-k}\times(1-pj)^k$
即对于子树和为 j，有 k 次上传失败的式子子。注意初始化，组合数和幂次都可以预处理。