Solution

NOIP2022 模拟赛

JKLover

2022 年 10 月 2 日

数组

算法一

若数组初始有 1，那么每次操作都可以将一个非 1 的数变成 1，需要的操作次数就是不为 1 的数的个数。
若初始没有 1，那么我们需要用最少的操作次数得到一个 1，然后再操作 $n-1$ 次。不难发现若一个区间内所有数的 gcd 为 1，就可以沿着同一个方向操作，以区间长度减一的操作次数得到一个 1。
枚举每个区间检查其 gcd 是否为 1 即可，时间复杂度 $O(n^2)$，期望得分 50 分。

数组

算法二

枚举区间时，如果我们固定左端点，当右端点向右移动时，区间 gcd 是不增的。于是可以枚举区间左端点，二分最小的右端点使得区间 gcd 为 1。查询区间 gcd 可以利用 ST 表实现，时间复杂度 $O(n \log n)$ ，期望得分 100 分。

矩阵

算法一

暴力搜索或 dp ，根据实现优劣程度，期望得分 20 到 50 分。

矩阵

算法二

可以先将所有的限制按照 max 值从小到大排序，每次处理所有 max 值相同的限制，算出它们的贡献后直接将它们所在的行列删去。因为之后的限制 max 值都比先前的大，被删掉的位置其实已经自动满足了后面的限制，就不用管了。

矩阵

算法二

假设有 r 个行限制与 c 个列限制，它们限定的最大值都是 mx，那么就是说这些位置的值不能超过 mx，并且每一行每一列至少有一个位置恰好取到 mx。

考虑用容斥原理计算这些限制的贡献，枚举 i 和 j 表示钦定其中有 i 行 j 列都没有取到 mx，那么贡献就是 $(-1)^{i+j}\binom{r}{i}\binom{c}{j}(mx-1)^t(mx)^{tot-t}$，其中 t 表示 i 行 j 列覆盖的所有位置数，而 tot 表示 r 行 c 列覆盖的所有位置数。

注意处理完一组限制后要将这些位置删去，即让 n 和 m 分别减去 r 与 c。

时间复杂度 $O(nm\log Mod)$，期望得分 100 分。

图论

以下分析均假定 n,m,q 同阶。

算法一

预处理出每个点能到哪些点，每次修改时依次暴力修改。时间复杂度 $O(n^2)$ ，期望得分 20 分。若没有操作 1 ，所有点权始终为 0 ，特判这类测试点，期望得分 30 分。

图论

算法二

考虑修改操作对之后每次询问的影响，不难发现，当询问 u 的点权时，往前找到最后一次能影响到 u 的赋值操作，再将这个赋值操作之后的所有能影响到 u 的取 min 操作拿出来，对修改权值一起求个最小值就是答案。

图论

算法二

于是可以先用 bitset 预处理传递闭包，将所有的修改操作存在栈里，询问时在栈中找出所有有影响的操作，计算出点权。

当栈的大小达到 $O(\sqrt{n})$ 时，将栈中所有操作依次暴力执行并将栈清空即可，时间复杂度 $O(\frac{n^2}{w}+n\sqrt{n})$ ，根据实现时占用的空间，期望得分 80 到 100 分。

图论

算法二

如果最后两个测试点空间不够，可以考虑拿时间换空间，第一次选出 $\frac{n}{k}$ 个点，只回答它们的询问，那么预处理传递闭包时也只需要记录每个点能否到达这 $\frac{n}{k}$ 个点，这样做 k 次，每次将 bitset 的空间拿来重复利用即可。时间复杂度乘了 k ，空间复杂度除以了 k 。k 的具体值可以根据自己的代码测试调整，对本题数据选 $k = 2, 3$ 是比较合适的。期望得分 100 分。

排序

算法一

暴力模拟，期望得分 20 到 40 分。

排序

算法二

考虑如何快速计算出答案。观察可以发现，每进行一次冒泡操作后，每个数前面的最大数如果比他大，就会被放到后面去。那么答案应该是 $\max\{f(i)\}$ ，其中 $f(i)$ 表示第 i 个数前面有多少个数大于他。

这个东西可以直接用树套树进行维护，时间复杂度 $O(n\log^2 n)$ ，期望得分 70 分。

排序

算法三

考虑 $f(i)=i-g(i)$ ，其中 $g(i)$ 表示第 $1\sim i$ 个数中 $\le a_i$ 的个数。而如果有两个数 $i<j, a_i\ge a_j$ ，显然 $f(i)<f(j)$ ，即 $f(i)$ 不可能成为答案。于是可以将 $g(i)$ 的定义改为所有数中 $\le a_i$ 的个数，这只会导致原本就不可能成为答案的某些 $f(i)$ 变小，不会影响答案 $\max\{f(i)\}$ 。

修改定义后的 $f(i)=i-g(i)$ 只需要用一棵权值线段树进行维护，时间复杂度 $O(n\log n)$ ，期望得分 100 分。