D7.T1 AtCoder - tokiomarine2020_e

题意：

> 给定 N 个两两不同的非负整数序列 A。求从中选择 $1 \sim K$ 个数，使得这些数的 `AND` 的值为 S，`OR` 的值为 T 的方案数。$A_i \le 2^{18}, N \le 50$。

题解：

首先大力状压 `DP` 加大力剪枝 (2^{36} 能过)。

其次，我们可以有一个很好的转化——显然只有 $S \wedge T$ 的位置上需要考虑——如果有一位都是 `1`，那么每一个选到的数这一位上全部都得是 `1`，不满足这位是 `1` 的直接排除；如果有一位都是 `0` 也同理，现在只需要考虑 $S \wedge T$ 位上的值了，也就是说，问题转化为了求 $S = 0, T = 2^r - 1$ 时的答案。

单独考虑两者中的一者，我们是会的；我们考虑把答案反演掉。

设 $f(k)$ 为恰好 k 处不合法的方案数，$g(k)$ 为指定 k 处不合法的方案数，那么二项式反演一下有：

$$f(k) = \sum_{i=k} (-1)^{i-k} \cdot \binom{i}{k} \cdot g(k)$$

考虑 $g(k)$ 怎么求：显然一个位置不合法，当且仅当两位同时为 0 或同时为 1，那么我们可以枚举哪几位相同，设 $h(U)$ 表示所有剩余的数中与 U 取且运算后相同的、大小 $\in [1, k]$ 的方案数，那么大力扫描数组就可以得出结果。于是：

$$Ans = f(0) = \sum_{U} (-1)^{\text{popcount}(U)} \cdot h(U)$$

时间复杂度：$2^r \cdot n$，评测链接：**AC Submission** (Faster ver.)，**AC Submission** (Slow ver.)。

D7.T2 AtCoder - agc047_d

题意：

> 有两棵点数同为 $2^H - 1$ 的满二叉树，第一棵树的第 i 个叶节点向第二棵树的第 P_i 个叶节点连了一条特殊边。 定义一个环的权值为它所经过的所有点的编号的乘积。求所有简单环且恰好经过两条特殊边的环的权值之和。

题解：

这个题很好地体现了一种优化枚举的思想，我们可以先枚举这个环在第一课树中的 `LCA`，然后暴力遍历这个 `LCA` 的子树，然后第二棵树中有一些叶子节点是被对应的，每个对应的叶子都打上标记，然后一起上传，总复杂度 $O(n \log^2 n)$。

D7.T3 AtCoder - keyence2020_f

题解：

> 有一个 H 行 W 列的矩阵，每个位置的颜色为白（`.`）或黑（`#`）。 你可以对其进行任意次操作，每次操作可以任选一行/列，将这一行/列的点全染成黑/白色。 问共能操作出多少种不同的矩阵。模 998244353，$H,W \leq 10$。