

洛谷 P7438

所求即为

$$[x^m y^q] e^{\left(\ln\left(\frac{1}{1-x}\right) - x\right)(1+y)}$$

满足

$$\frac{\partial F}{\partial x} = \frac{x(1+y)F}{1-x}$$

CF1510H

把线段的包含关系建成树，发现不好直接 DP。考虑把每条子线段不断往孩子移，这样每条子线段都覆盖了两个相邻孩子的间隔。容易发现这样操作后只需要满足每棵子树里的子线段数至少为子树大小，就能还原出合法方案。还容易证明选择的子线段端点一定是原来线段的某个端点，因此直接设 $f(u, x, b_l, b_r)$ 表示顶点 u 的子树内选择 x 条线段，左右是否额外选择了一个端点的最大长度（左边额外选择了一个右端点 x 则代价增加 x ，右边额外选择了一个左端点则增加 $-x$ ）。由于子树的状态数是 $O(\text{size}_u)$ 的，因此总复杂度为 $O(n^2)$ 。

CF1495F

容易发现任意两条第二种边要么不相交，要么相互包含，因此可以和上一题一样建成树。要求最短路，其实就是选择若干条互不相交的第二种边，若选择了 (l, r) ，则它的额外代价是 $\min\{0, b - \sum_{l \leq i < r} a_i\}$ 。先 DP 出每棵子树的最小代价，则询问 (x, y) 也就是求所有 $[x, y]$ 包含的极大的边的代价之和。每个点减掉所有儿子的代价容斥一下，就变成了一般的二维偏序，可以 $O((n+q) \log n)$ 解决。

CF1515H

假设询问重复的数算多次，那么可以直接用 Trie 维护，修改时找到对应的子树接到修改后的位置（这里要合并子树）并打标记，复杂度 $O((n+q) \log C)$ 。

现在要去掉重复的数，不妨直接考虑去掉重复的结点。考虑到 AND 和 OR 操作会合并一些结点，但是直接打标记并不能解决。但是我们容易知道是否有结点需要合并，也就是维护哪些深度的结点有两个孩子，如果这个深度还有 AND 或 OR 的标记，则有结点需要合并。我们可以以 $O(\log C)$ 的复杂度减少一个结点，而每次修改会增加 $O(\log C)$ 个结点，故总复杂度为 $O((n+q) \log^2 C)$ 。

CF1514E

任意一张竞赛图都有哈密顿路径。考虑以任意顺序加点，假设之前求出的哈密顿路径为 v_1, v_2, \dots, v_n ，则新加入顶点 u 后，如果 $u \rightarrow v_1$ 或 $v_n \rightarrow u$ 则将 u 加在路径最前或最后即可。否则，容易发现一定存在 $1 \leq i < n$ 使得 $v_i \rightarrow u \rightarrow v_{i+1}$ 。并且这个位置可以二分，即若目前有 $v_l \rightarrow u \rightarrow v_r$ ，对于 $l < m < r$ ，若 $u \rightarrow v_m$ ，则考虑 $v_l \rightarrow u \rightarrow v_m$ ，否则考虑 $v_m \rightarrow u \rightarrow v_r$ 。

求出哈密顿路径后，可以从前往后加点，用栈维护当前的强连通分量。新加入一个点 u 时，我们只需要找到它连向的最前的强连通分量，因此从长到短询问 u 是否向一个前缀连边，直到第一次发现没有连边，则除这个前缀外的所有强连通分量都会与 u 合并。

CF1503F

考虑从 x 往 $-x$ 一条边（ a 和 b 都连），那么最后得到的会是若干个大小为偶数的环，环上边的方向不一定是一样的，且 a 和 b 对应的边交替出现。容易发现，如果存在 $u \rightarrow v \rightarrow w$ ，则可以将这三个点合并成一个点。最后如果能缩成只有两个点 u, v 且 u 向 v 有两条边则构造出一组解。能这样构造的条件是 a （或 b ）在两个方向的边数恰好差 1。

考虑一个合法的序列，在 x 和 $-x$ 之间画一个半圆， a 在下面， b 在上面。因为是合法括号序列，每个连通块都是简单回路。考虑一个点沿着回路走一圈面对的方向越过 x 轴正方向的次数就可以知道上文所说的条件是必要的。

SRM793 Hard [TinyChessboardNim](#)

棋盘实际上是四元环。

归纳证明：若 x 是最大值，且与 x 相邻的两个点不相等则先手必胜。

若与 x 相邻的两个数是 a 和 b ($a > b$)，则先手可以将 x, a 各取走 $a - b$ ，变成 $[x - a + b, b, y, b]$ 。若此时是必败态则证毕。否则根据归纳假设，后手下一步只能将 $x - a + b$ 和 b 各取走 $x - a + b - y$ ，然而这个状态是先手可以直接到达的，因此先手必胜。

若当前状态为 $[x, a, y, a]$ ，若 $y \geq a$ 则先手必败，因为后手可以执行和先手相反的操作，最后一定是两个 a 先取完。否则，先手唯一可能的操作就是将 x 和 a 各取走 $x - y$ ，之后另一个 a 变成最大值。这样的操作可以写成 $(x, a, y) \rightarrow (a, y, y - (x - a))$ ，容易算出可以操作的步数，根据奇偶性即可判断胜负。

SRM789 Hard [FollowingNim](#)

我们把集合 S 称作特殊操作。

定义 1 一堆石子是**特殊必胜**的，如果在只使用特殊操作的情况下先手必胜。

根据定义 1，可以用 DP 求出所有特殊必胜的石子数。

引理 2 如果有一堆石子是特殊必胜的，则先手必胜。

证明 设这堆特殊必胜的石子数为 x ，考虑两种情况：

1. 如果存在某个 x' ($0 \leq x' < x$) 使得将 x 拿到剩 x' 的状态是必败态，则先手执行这个操作即可获胜。
2. 如果对所有 x' ($0 \leq x' < x$)，将 x 拿到剩 x' 的状态都是必胜态，若任意玩家对 x 这堆石子进行了非特殊操作，则另一位玩家得到了必胜态。因此，先手只需要对 x 进行特殊操作，则两位玩家只能交替进行特殊操作。同时，由于 x 是特殊必胜的，因此先手可以获胜。

引理 3 如果所有石子堆都不是特殊必胜的，则任意一位玩家都不会使得一堆石子变得特殊必胜，或执行特殊操作。

证明 根据引理 2 的证明，先手一定存在操作 x 这堆石子的必胜策略，因此无论进行的是否是特殊操作，都不会有一位玩家将一堆石子变得特殊必胜。同时根据定义 1，任意一堆非特殊必胜的石子，进行特殊操作后，都会变成特殊必胜，因此不会有玩家进行特殊操作。

根据引理 3，在所有石子堆都不是特殊必胜的情况下，每次操作都是独立的，因此可以用 SG 函数表示局面。

GYM103069E

考虑把格子黑白染色，每个连通块染一个颜色，选择的边两侧的格子颜色不同。注意到一个顶点是拐点当且仅当周围四个格子有 1 或 3 个黑色 (或颜色的异或为 1)，因此一个格子四个顶点拐点个数的奇偶性等于其左上、右上、左下和右下四个格子颜色的异或。此时，若 n 和 m 都是偶数，则能够唯一确定每个格子的颜色，只需检查是否合法以及计算代价。如果只有一维是偶数，不妨设 n 是偶数， m 是奇数。则能够确定的是所有偶数列格子的颜色，而剩下的格子每行只有两种方案。由于判断合法性以及计算代价只和相邻三行的格子有关，因此可以从上往下 DP，状态里记录前两行染色的方案即可。时间复杂度 $O(nm)$ 。

