A、perm

要想求出每个前缀的答案，就要先考虑单次求答案。不妨思考哪些元素是一定不能被删去的，即对于一个 a_k ，不同时存在 $j < k, a_j > a_k$ 和 $l > k, a_l > a_k$。

发现 a 序列的前/后缀最大值满足要求，且如果一个 a_i 既不是前缀最大值，也不是后缀最大值，那么其前后一定都有比自己大的元素，它不满足要求，可以被删去。那么现在问题变为有 $|a|$ 个元素，a 的前缀最大值有 k_1 个，后缀最大值有 k_2 个，则有 $k_1 + k_2 - 1$ 个元素不能被删去（全局最大值同时是前缀最大值与后缀最大值，被重复统计），$|a| - k_1 - k_2 + 1$ 个元素可删可不删，则答案为 $2^{|a|+1-k_1-k_2}$，对每个前缀也是这样做，而 k_1, k_2 只需要用单调栈就能对每个前缀求出。$O(n)$。

B、array

梦梦的决策只有 $2n$ 种：选择一个 i，再选择放 a_i, b_i 的顺序，其余的都由熊熊决定，于是可以据此设计 dp：设 dp_i 表示以 i 结尾的最大子段和，那么 $dp_{2i-1} = \max\{0, \max(A_i, B_i) + dp_{2i-2}\}$，$dp_{2i} = \max\{0, A_i, B_i, A_i + B_i + dp_{2i-2}\}$，但转移到梦梦决策的 $2i$ 时需要特殊处理，这样就得到 $O(n^2)$ 的做法。

接下来考虑如何加速。梦梦的决策影响是很有限的，于是对于其不能决策时做一次上述 dp，再倒转序列求出 rdp_i 表示以 i 开头的最大子段和，那么梦梦如果操作了 $2i-1, 2i$，则新的最大子段和只需要分为完全在 $[1, 2i-2], [2i+1, 2n]$ 中，以及包含 $2i-1, 2i$ 中的恰一个或是包含两个（跨过），前者直接用 dp, rdp 查询，后者容易合并。$O(n)$。

C、string

对于单次对 T 的询问，设 f_i 表示 $T[1,i]$ 中以 `0` 结尾的合法子序列数量，g_i 表示 $T[1:i]$ 中以 `1` 结尾的合法子序列数量。则对于 T_{i+1} 分类讨论转移：

- T_{i+1} 为 `0`：$f_{i+1}=f_i+g_i$，$g_{i+1}=g_i$。
- T_{i+1} 为 `1`：$f_{i+1}=f_i$，$g_{i+1}=f_i+g_i+1$。
- T_{i+1} 为 `_`：$f_{i+1}=f_i$，$g_{i+1}=g_i$。

三种转移都容易用 3×3 的矩阵刻画，其中过程向量为 $[f_i,g_i,1]$。

令 $d=3$，用线段树维护区间转移矩阵乘积来加速转移，每次 1 操作暴力递归到没有变为 `_` 的叶子节点，可以做到 $O(qd^3\log n)$，可以通过本题。

D、 divisor

若 $x=\prod_{i=1}^{k} p_i^{e_i}$，则 x 的因数个数 $d(x)=\prod_{i=1}^{k}(e_i+1)$，则我们只关心 e 序列，则不妨令 $e_1 \geq e_2 \geq \cdots \geq e_k$，因为幂的顺序并不重要。题述操作等价于给某一项加减 1 或是添加一项 1，操作之后将 0 去掉并重新排序。对于 $x \leq 10^6$，可以找出只有 $m=289$ 个这样本质不同的 e 序列，所以计算出 m^2 对最短距离即可。

注意到对于 $x=2^{19}, y=2^2 3^6$ 时，将 x 乘 2 就可以使两者均有 21 个因子，但这样就会出现 $\{20\}$ 这样不在 289 个合法序列中的过程 e 序列。如果不考虑这样的情况，可以得到任意两个数最多需要 10 次操作，于是一个比较松的限制是最优操作中所有过程中的 e 满足 $\sum e_i \leq 29$，即初始有 $\sum e_i \leq 19$，经过至多 10 次操作后 $\sum e_i$ 不超过 29。再进一步发现目标的 $\sum e_i$ 也不超过 19，那么这个上界还能缩小，精确的上界可以验证得到为 22，提前从 289 个起点出发 bfs，每次询问记忆化地合并答案即可，可以在时限范围内通过。