

草莓列车(game)

60pts

此时 $n \leq 10^5, m \leq 10^6$ ，线段树即可。

80pts

只有 $l = 1$ 的操作，离线后扫一遍即可。

100pts

我们可以回忆一下 st 表，它可以做到 $\mathcal{O}(n \log n)$ 处理， $\mathcal{O}(1)$ 查询，但是此题的需求恰恰相反，于是我们可以反着做 st 表，对于一次操作 $l\ r\ x$ ，可以像正常的 st 表查询时一样把它拆成两个长度为 2^k 的区间，将 $st[l][k]$ 和 $st[r-(1 \ll k)+1][k]$ 打上标记，最后再做一遍 $\mathcal{O}(n \log n)$ 的处理，就得到了最终的序列。

时间复杂度 $\mathcal{O}(n \log n + m)$ 。

如果没有想到这个脑筋急转弯，但你拥有不错的数据结构基础，其实也是可以做出来的。

考虑分治，当我们分治到一个区间时处理恰好穿过 mid 的操作，可以发现这时变成了子任务 4 的情况。所以分治这部分总共是 $\mathcal{O}(n \log n)$ 。

现在的问题是怎么把一个形如 $l\ r\ x$ 的操作 $\mathcal{O}(1)$ 定位到分治区间上。如果我们把分治的区间的关系看成一棵树，可以发现 $l\ r\ x$ 对应的分治区间就是叶子 l 和叶子 r 的 LCA。我们可以先把 n 扩大成 2^k ，这时 $l \wedge r$ 的 highbit 就对应着 LCA 的深度。因为树高只有 $\log n$ ，所以可以预处理好 $\mathcal{O}(1)$ 查询。这个做法的复杂度也是 $\mathcal{O}(n \log n + m)$ 。

草莓路径 (path)

40pts

考虑两点间的最优路径怎么找。首先随便找一颗生成树，取出剩下的所有非树边对应的环的权值，发现两点之间的任意一条路径的权值都可以由它们树上的路径的权值再取一些环异或起来得到。所以枚举两点然后线性基即可。

100pts

现在的问题相当于从一个集合 A 里面取出两个数，再从另一个集合 B 里面取出任意多个数异或起来，要让得到的数最大。

用 B 中线性基的主元消掉 A 中元素的对应的位，可以发现现在从 A 中选出两个数之后，全选 B 中的主元一定是最优的。现在就转化成了普通的 trie 树问题。

草莓城市 (city)

容易发现坐标全部乘以 2 之后答案只可能是整数。

二分答案，然后每个放三角形的位置有 4 种放法，直接做的话是一个 4-sat 问题。

设现在直角三角形的斜边长为 $2mid$ ，每种放法的三角形都可以拆成两个腰长为 mid 的小等腰直角三角形，我们惊奇地发现所有 4 种放法对应的三角形都可以表示成左上、右下两个小三角形二选一和左下、右上两个小三角形二选一并起来。这样一个 4-sat 的状态就转化成了两个 2-sat 的状态。

草莓之歌 (easy)

40pts

对于一个确定的局面，考虑如何判断它是否能分为 k 段。首先，我们希望分出的段数尽量少，如果不足 k 段，我们可以再把它们拆开。考虑贪心划分，首先我们可以把第一个 B 之前的所有 A 划分到第一段，再给他们匹配 B，可以发现匹配开头的一段 B 是不劣的，然后把第一段的 AB 删除继续贪心。于是我们对于一个确定的局面可以 $\mathcal{O}(n)$ 地判断。

在这个贪心的基础上考虑 dp，我们从左往右地填，设出 dp 状态： $f_{i,j}$ 表示第 j 段的开头的 A 是第 i 个 A，前 $j-1$ 段的最小代价。我们每次枚举当前这一段包含多少个 A，从 $f_{x,j}$ 转移到 $f_{y,j+1}$ 那么就需要排名在 $[x, y-1]$ 的 A 都要出现在第 x 个 B 之前。记 pos_i 为第 i 个 A 之前有多少个 B，那么这一段的代价就是 $\sum_{i=x}^{y-1} \max(0, x - pos_i + 1)$ 。

从这个角度直接 dp，我们就获得了一个三方做法。

61pts

在此基础上，不难想到其答案关于 k 应该是一个凸函数，于是我们可以 wqs 二分，得到一个 $\mathcal{O}(n^2 \log n)$ 的做法。

100pts

再观察上面的转移，化一下式子发现可以斜率优化，于是使用单调队列维护凸包，即可复杂度 $\mathcal{O}(n \log n)$ 。