CF817F MEX Queries

离散化下来，现在有 01序列，支持：

区间染 1；区间染 0 ； 区间翻转；寻找第一个 0 的位置。

线段树维护即可：每个节点只用关心只否全0/全1，查询时线段树上二分即可。

CF1149C Tree Generator™

先想想知道括号序列后怎么计算两点的距离，其实就是把 out_u 到 in_v 这一段匹配的括号去掉之后，剩余的括号个数；当然如果 u, v 有祖先关系计算方式不太一样，但总的来说，可以发现答案一定等于：取某个区间，把匹配的括号去掉后，剩余括号数量最大值。把左括号看成 +1，右括号看成 -1 ，看成是取相邻两个连续段，让两段的和之差尽量大。

现在问题变成序列上的问题，可以线段树维护，不过维护的东西比较多。好消息是修改都是单点修改所以不需要再考虑 tag 要记啥。

CF1083C Mex Tree

令 a_i 是值为 i 的点编号。考虑查询是怎么做的：找到最小的 M 使得 $a_0, a_1, \ldots, a_M$ 不在一条路径上。由于我们只关心能不能找出这样一条路径，所以一些点的信息其实可以表示成：包含它们的最小路径。可以发现这个路径是唯一的。

我们要优化找 M 的过程，考虑建出线段树，每个节点维护区间内的 a_i 构成的最小路径，然后线段树二分即可。

但现在有一个比较难处理的事情：怎么合并信息，发现需要求两条路径的并，形成的最小路径。怎么求呢？[画图，分类讨论]

先预处理 dfn 的 ST 表，即可做到 $O(n \log n)$ 。

CF786B Legacy

可以发现直接连边边数太多，考虑怎么优化。

线段树优化建图。[画图]

P2824 排序

要点在于：我们只关心末状态中一个位置的值。

二分答案，变成维护 01 序列。容易了。

UOJ715 小明的树

我们说灯亮的点是黑色的。

尝试对一个状态找到一个易于计算的值，使得通过这个值能判断该状态是否合法。

建立一个新点 0 ，在 $0,1$ 间连边，然后可以发现：在合法状态下，每个黑色点组成的联通块，它们只会往外连一个白色点。

于是设定权值为： 黑白边个数减去黑色联通块个数，这个是一定 ≥ 0 的，且如果合法就等于 0 。黑色联通块又等于黑点减黑边。设 a_i 是点亮了 1 到 i 后状态的权值，一次删边加边的操作其实就是对 a 进行区间加，询问就是询问 0 的个数。线段树即可，维护 min 和 $cntmin$ 即可。

CF765F Souvenirs

支配对的思想：我们其实可以淘汰大部分 (i,j) 。在这个题里的话，可以发现如果存在 i',j' 满足 $i\leq i'\leq j'\leq j$ 且 $|a_{i'}-a_{j'}|\leq |a_i-a_j|$ ，那 (i,j) 就被淘汰了。

我们来寻找没被淘汰的数对。绝对值比较烦，先只看 $a_i\leq a_j$ 的。

枚举 i ，找到 j_0 满足 $i<j_0$ 且 $a_i\leq a_{j_0}$ ，且 j_0 最小。接下来如果存在 j_1 满足 (i,j_1) 可能成为最优的数对，那来考察一下 a_{j_1} 需要满足什么条件，(i,j_1) 才是有可能被取的数对。

初步的观察是 $a_{j_1}\leq a_{j_0}$ ，否则 $a_{j_1}-a_i$ 肯定比 $a_{j_0}-a_i$ 更大。再想一想：如果 $a_{j_1}-a_i\geq a_{j_0}-a_{j_1}$ ，那我们取 (j_0,j_1) 一定更优！

于是 a_{j_1} 一定在 $[a_i,\frac{a_i+a_{j_0}}{2}]$ 中，找到满足条件的 j_1 中最小者。同理也可以找到 $j_2,j_3,\dots$ 发现 a_i-a_j 每次会至少减半，于是可以得到：这样的数对只有 $O(n\log V)$ 种。

询问是容易的：树状数组即可。问题在于每次怎么找 j ：我们按照 a 的值从大到小的枚举 i ，现在相当于有单点修改以及询问，每次要找到 x 开始第一个满足 $b\leq r$ 的位置。

线段树二分即可。复杂度 $O(n\log n\log V)$ 。

qoj8672 排队

考虑枚举 r 然后维护序列 a ，其中 a_l 表示 $[l,r]$ 的答案。这样询问其实就是单点查询。

考察从 $r-1$ 到 r 时 a 的变化：如果 $a_i\in[L_r,R_r]$ 那 a_i 就会加 1 。

容易发现 a 永远单调不降，所以满足条件的 i 是一段区间，如果能找到这段区间，那只需要用差分处理，单点查询变成查前缀和，用树状数组即可。

怎么找到这段区间呢。相当于有 $Ask(x)$ 为最后一处前缀和 $\leq x$ 的位置。考虑树状数组二分。树状数组二分其实是一个倍增的过程。[画图]

P3203 [HNOI2010] 弹飞绵羊

可以 LCT。但注意到 $n \leq 2 * 10^5$ ，尝试分块。

分块后对于每一个点维护：只在这个块内跳，最终会跳到哪个点；会跳多少步。

发现查询，修改都是容易的。代码 700B

P4198 楼房重建

转化一下其实就是单点修改和查询整个序列前缀最大值个数。

单侧递归线段树。我们在每个节点维护前缀最大值个数和 max 值，发现合并的时候其实要解决：

Ask(p,z) 在 p 这个节点内的序列，开头接个 z 后前缀最大值的个数。

分类讨论：如果 $z \geq mx_{ls_p}$ 那答案就是 $Ask(rs_p, z)$ 。

否则答案是 $Ask(ls_p, z) + Ask(rs_p, mx_{ls_p}) = Ask(ls_p, z) + cnt_p - cnt_{ls_p}$ 。

于是单次 Ask 是 $\log n$ 的，而我们要进行 $O(q \log n)$ 次信息的合并，复杂度就是 $O(q \log^2 n)$ 。