Ranked Choice Spoiling

$k=2$：

对于 `AB`，有 `CAB` 和 `ABC/ACB(等价)` 两种情况；对于 `BA`，有 `CBA` 和 `BAC` 两种情况（`BCA` 不行的原因：第一轮要淘汰 `B`，第二轮淘汰 `C`，那么 `BCA` 肯定是碍事的）。枚举 `CAB` 数量，得到一个关于 `CBA` 数量的不等式，判其有解性即可。

$k=3$：

枚举删字母的顺序，然后列出所有不等式，变成一个整数规划有解性的问题，我们相信这些不等式都能比较好的化简。

Useful Algorithm (ICPC 2022 Hangzhou B)

> 考虑两个 m 位二进制数 a,b 做加法的过程，记 $S(a,b)$ 为在第 i 位处进位了的 i 集合。 给定 $w_0\cdots w_{m-1}$，记 $D(a,b)=\max\{w_i|i\in S(a,b)\}$，$S(a,b)=\emptyset$ 时 $D(a,b)=0$。现在有 n 对 (a_i,b_i)，求 $\max_{i,j}\{D(a_i,a_j)\cdot(b_i+b_j)\}$，支持 q 次单点修改 (a_i,b_i)，强制在线。$n,q\le 10^5,m\le 16$。

首先，每一位是独立的。并且取 $\max$ 的问题可以线段树分治；考虑加入每个数的时候，分别考虑每一位是否有进位。也就是说：判断 $(a_i \bmod 2^k)+(a_j \bmod 2^k)\ge 2^k$；用线段树来维护。

考虑去掉线段树分治：另 $v_i=u_{2^k-i-1}$，那么问题就变成了动态维护 $\max\{a_i\max_{j\le i}\{a_j\}\}$。这个是经典的具有结合律的信息，线段树维护就好。

Shortest Path (ICPC 2022 Jinan I)

> 给定一张边带权无向图，令 $d(s,t,c)$ 为 s 到 t 恰好经过 c 条边的最短路（经过同一条边两次算两条，不存在则为 0），求 $\sum_{i=1}^{c}d(1,n,i)$。$n\le 2000,m\le 5000,c\le 10^9$。

首先肯定只会经过一个奇环，因为如果经过多个奇环的话，可以去掉偶数个奇环，变为在就经过的最短的边上来回徘徊（净增加 2 的倍数）。

枚举这条最短的边，然后设 $f_{i,0/1}$ 表示到 i 的奇偶的最短路（用 dij 可以求出），然后枚举一条边，求出经过这条边的 $2^2\cdot 3$ 种情况的边，然后合理地把这些射线的下凸包求出来就好了。

Caramel Clouds (CF833E)

> 有 n 朵云，第 i 朵会在 $[l_i,r_i]$ 时间内出现。对于每个时刻，如果天上有云则会挡住太阳，因此你可以花费 c_i 颗糖使得第 i 朵云不出现，但你只有 C 颗糖，且最多让两朵云不出现。你在花园里种下了树苗，树苗在有 k_i 个时刻受到阳光照射（没有被云挡住）后会长大。求树苗最早长大的时间。多组询问 k_i。
> $n,q\le 10^5$，$l_i,r_i,k_i\le 10^9$。

两朵云的贡献为：A 单独覆盖的长度 + B 单独覆盖的长度 + AB 一起覆盖的长度。

先不考虑第三项，那么易见每个是独立的；我们只要求出数组 f_x 表示前 x 时间能达成的最大贡献和次大贡献，增大 x 时依次增大每朵云的贡献，动态维护最大值和次大值。

考虑第三项：由于本质不同段数是 $O(n)$ 的，所以不同的第三项的数量也是 $O(n)$ 的；枚举每段 AB，然后将其贡献到后面的 A 和 B 的前缀最大值上；对于每朵云，分别维护每个连续段的前面能达到的时长的最大值，即可。