分数约分

算法 1

暴力枚举分子分母分别删了哪些位再判断即可，复杂度 $\mathrm{O}(4^{\log n}\log n)$，期望得分 40。

算法 2

当 $a=b$ 时，答案显然为 a 的所有数位中最小的非零位，期望得分 20。

算法 3

考虑枚举 a 保留哪些位，可以简单算出 b 删除后的样子。

去 b 中暴力匹配需要的每一位，再看下 b 中没匹配上的位的多重集和 a 中删除的位的多重集是否一致即可。

一个细节是 b 删除后的样子可能存在前导零，解决方法是从低向高匹配如果 b 已经完全匹配上那么遇到 0 就不停匹配，这样 b 中剩余的 0 一定不多于 a 中的。又因为我们可以枚举 a 的每个零选不选所以一定能找到一种恰好与 b 中零数量相同的解。

复杂度 $\mathrm{O}(2^{\log n}\log n)$，期望得分 100 分。

共享单车

算法 1

考虑状压 dp，令 $f_{S,u}$ 表示已经走过了 S 中的点，目前在点 u 的情况下到达 n 还需要的最小时间。

转移的话如果 u 上没有单车那么应该有

$$f_{S,u} = \min_{x \notin S} w_{u,x} + f_{S \cup x,x}$$

如果 u 上有单车那么如果 u 上单车故障那么需要再找一个点，没有故障显然就会直接骑到终点，于是有

$$f_{S,u} = \min_{x \notin S} \left(\frac{p_u}{100}(w_{u,x} + f_{S \cup x,x}) + \left(1 - \frac{p_u}{100}\right) dis_{u,n} \right)$$

答案就是 $f_{\{1\},1}$。时间复杂度 $\mathrm{O}(2^n n^2)$，期望得分 40。

算法 2

跑出所有有车点之间的最短路，显然图的大小就变成了 k，跑算法 1 即可，时间复杂度 $\mathrm{O}(2^k k^2 + kn\log m)$，期望得分 100。

积木拼接

考虑从两侧向中间构造，令 $f_{l,r,x}$ 表示左边选到 s_l，右边选到 s_r，并且 s_r 有一个长度为 x 的前缀没有匹配上。

令 $g_{l,r,x}$ 表示左边选到 s_l，右边选到 s_r，并且 s_l 有一个长度为 x 的后缀没有匹配上。

考虑如何转移 $f_{l,r,x}$ 和 $g_{l,r,x}$，这里我们只讨论 $f_{l,r,x}$。

- 如果 $|s_r| - x \geq |s_l|$ 那么说明与 s_l 匹配的部分完全在 s_r 内，所以去掉 s_l 可以从 $f(i, r, x + |s_l|)(i < l)$ 转移来。
- 如果 $|s_r| - x < |s_l|$ 那么说明 s_r 匹配的部分完全在 s_l 内，所以去掉 s_r 可以从 $g(l, j, |s_r| - x)(j > r)$ 转移来。

显然上述转移可以用前缀和做到 O(1)，且可以通过 KMP 预处理每一对串的匹配情况即可 O(1) 判断转移合法。

有了 f, g 接下来要统计答案，考虑枚举中心，有如下几种情况：

- 中心位于两串之间，对 $f(l, r, 0)$ 求和即可。
- 中心位于一个串内部但不是这个串的中心，只需要枚举 $f(l, r, x)$ 和 $g(l, r, x)$ 并判断剩余的长度为 x 的部分是否回文即可，注意 x 不能为整个串的长度。
- 中心就是一个子串的中心，只统计 $f(l, r, |s_r|)$ 即可，需满足 s_r 回文。
- 单独就一个回文串的情况。

可以用 manacher 辅助判断回文。

科学研究

算法 1

考虑一个位置做若干次操作后会变为 $k_p^{k_{p-1}^{\cdots}}$，若从底层开始用扩展欧拉定理降幂则经过 $\mathrm{O}(\log m)$ 层后模数就变成了 1，于是我们只需要维护这个 log 层幂塔即可。

于是考虑我们对一个区间进行 $\mathrm{O}(\log m)$ 次操作区间就会完全相等，于是考虑势能分析，考虑将每组相邻元素之间的势能定义为两个幂塔的不同位置数量，那么每次区间操作会给区间内部的每个间隔势能 -1，同时区间两侧的间隔势能 $+\mathrm{O}(\log m)$，于是总势能为 $\mathrm{O}((n+q)\log m)$，这支持我们暴力修改来减少势能。

具体的维护可以用 set 维护相等连续段，线段树维护区间和，那么一次操作在 set 的一个连续段上相当于区间赋值。

若视 n, m, q 同阶那么时间复杂度 $\mathrm{O}(n\log^3 n)$，大概能得到 40 分。

算法 2

瓶颈在于算快速幂，但是由于幂塔只有 log 层且每层模数相等所以考虑预处理模数，具体的类似 BSGS 一样光速幂即可。

预处理复杂度 $\mathrm{O}\left(M^{3/2}+\left(\frac{M}{2}\right)^{3/2}+\dots\right)=\mathrm{O}(M^{3/2})$，查询 $\mathrm{O}(1)$。

总时间复杂度 $\mathrm{O}(n\log^2 n)$，期望得分 100。