

## 【北大集训 2021】经典游戏

### 【题目描述】

<https://uoj.ac/problem/719>。

Alice 和 Bob 在一棵有根树上博弈，树的每个点  $i$  上有  $a_i$  个棋子，两个人轮流选择一个非叶子节点上的棋子，将其移入其子树的一个节点上（这个节点不能是自己），并且 Bob 可以在游戏开始前往某一个节点上放一个棋子。需要支持两种操作：单点修改  $a_i$ 、查询点  $u$  的邻域中有多少个点为根时 Alice 必胜。

$n, q \leq 10^6$ 。

### 【题解】

首先，每个棋子是独立的，所以可以将所有每个棋子的 SG 函数异或起来得到总的异或值；而单个棋子的 SG 值就是其子树中离它最远的叶子节点。

这棵树暂时没有根，我们不妨给其找一个根；“离  $u$  最远”的限制使得这个根最适合放在直径的中点上（如果直径上点数为偶数，那么可以有两个根），可以推出一个性质：当“根”为  $u$  的祖先时， $SG(u)$  为其子树内的离其最远点；当根为  $u$  或者位于  $u$  的子树内时， $SG(u)$  为  $u$  到另一个直径端点的距离。记下这两个值分别为  $up_u$  和  $down_u$ 。

Alice 必胜，则意味着  $\oplus SG_u > up_{root}$ 。考虑快速统计  $root$  有多少个孩子满足如上条件，可以发现其所有孩子  $up$  都等于  $up_u + 1$ ，所以将所有孩子的 SG 值插进 Trie 里即可快速查询；而修改可以被转换为两个 dfn 序上的区间异或。拿一个树状数组维护异或的 tag 即可。

### 【提交记录】

<https://uoj.ac/submission/581557>。

## 【WC2014】时空穿梭

### 【题目描述】

<https://www.luogu.com.cn/problem/P4152>。

小 X 驾驶着他的飞船准备穿梭过一个  $n$  维空间，这个空间里每个点的坐标可以用  $n$  个实数来表示，即  $(x_1, x_2, \dots, x_n)$ 。为了穿过这个空间，小 X 需要在这个空间中选取  $c$  ( $c \geq 2$ ) 个点作为飞船停留的地方，而这些点需要满足以下三个条件：

1. 每个点的每一维坐标均为正整数，且第  $i$  维坐标不超过  $m_i$ 。

- 第  $i + 1$  ( $1 \leq i < c$ ) 个点的第  $j$  ( $1 \leq j \leq n$ ) 维坐标必须严格大于第  $i$  个点的第  $j$  维坐标。
  - 存在一条直线经过所选的所有点。在这个  $n$  维空间里，一条直线可以用  $2n$  个实数  $p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n$  表示。直线经过点  $(x_1, x_2, \dots, x_n)$ ，当且仅当存在实数  $t$ ，使得对  $i = 1 \dots n$  均满足  $x_i = p_i + tv_i$ 。
- 问方案数 (mod 998244353)。  $T \leq 100, n \leq 11, c \leq 20, m_i \leq 10^5$ 。

### 【题解】

我们枚举相邻两点间距离的最简比序列，可以发现填数的方案数只和总长有关，所以只需要枚举序列的总和，插板法计算序列的具体方案数，然后把“最简比”莫比乌斯反演掉。求方案数时，每维独立，对于每一维分别枚举其放大的倍数并将方案数求和（是一个等差数列求和），最后将每一维相乘即可。

预处理组合数，预处理  $c = 2 \sim 20$  的所有需要的数组，借助快速莫比乌斯反演可以将复杂度降低到勉强能过的程度。

### 【提交记录】

<https://www.luogu.com.cn/record/85925954>。

## 【POI2012】PRE-Prefixuffix

### 【题目描述】

<https://www.luogu.com.cn/problem/P3546>。

对于两个串  $S_1, S_2$ ，如果能够将  $S_1$  的一个后缀移动到开头后变成  $S_2$ ，就称  $S_1$  和  $S_2$  循环相同。例如串 ababba 和串 abbaab 是循环相同的。

给出一个长度为  $n$  的串  $S$ ，求满足下面条件的最大的  $L$ ： $S$  的  $L$  前缀和  $S$  的  $L$  后缀是循环相同的。

$n \leq 10^6$ 。

### 【题解】

枚举  $S$  的 Border，记当前 Border 的长度为  $i$ ，那么我们就是要求  $S[i + 1, n - i]$  的最长 Border。记  $T$  的最长 Border  $f(T)$  考虑到  $f(S[i + 2, n - i - 1]) \geq f(S[i + 1, n - i]) - 2$ ，所以我们  $i$  从大到小计算，只需要 check  $\leq f(S[i + 2, n - i - 1]) + 2$  的即可。从大到小 check，不难发现均摊的复杂度是对的。

另一种理解是：将字符串变为  $s_1, s_n, s_2, s_{n-1}, s_3, \dots$ ，那么我们要找其一个回文前缀  $i$ ，和  $S[i+1, 2n]$  的最长回文前缀长度相加即可。两种方法是等价的，因为方法一其实本质上也是 manacher。

### 【提交记录】

<https://www.luogu.com.cn/record/85928701>。

## 【CF1178H】 Stock Exchange

### 【题目描述】

<https://www.luogu.com.cn/problem/CF1178H>。

股票交易所里有  $2n$  种股票，每种股票有两个属性  $a_i, b_i$ ，在时刻  $t \geq 0$ ，第  $i$  种股票的价格为  $a_i * [t] + b_i$ 。

每个时刻可以进行任意次股票交易，在时刻  $t$  时能够把股票  $i$  换成股票  $j$  当且仅当股票  $i$  在时刻  $t$  的价格不小于股票  $j$  在时刻  $t$  的价格。

现在你手上有 1 到  $n$  号股票各一张，现在要求的是把这些股票换成  $n+1$  到  $2n$  号股票各一张的最早时刻，以及在最早换完股票前提下的最少交易次数。

$1 \leq n \leq 2200, 0 \leq a_i, b_i \leq 10^9$ 。

### 【题解】

由一次函数的性质，得存在一种最优方案，使得只有第 0 天和第  $T$  天有交换股票。我们考虑二分 + 最大流判断，然后用费用流求最小代价。

二分 check 部分的优化：在第 0 天将所有股票换成其能换成的  $a_i T + b_i$  最大的股票，最后贪心 check。

费用流部分的优化：每个点直接以 1 的代价连向第 0 天排序结果的一个前缀，以 2 的代价连向它能连向的第  $T$  天排序结果的一个前缀，即可将边数将至  $O(n)$ 。

### 【提交记录】

<https://www.luogu.com.cn/record/85963059>。

## 【IOI2019】天桥

### 【题目描述】

<https://www.luogu.com.cn/problem/P5812>。

有  $n$  个高度分别为  $h_{1\sim n}$  的大楼分布在一条数轴上，有  $m$  座天桥从  $(x_1, y)$  横亘至  $(x_2, y)$ ，穿过了与其相交的大楼。问  $s$  大楼底部到  $t$  大楼底部的最短路（只能走大楼和天桥）。

$n, m \leq 10^5$ 。

### 【题解】

这张图的问题在于点数太多。记关键点为如下两类：

1. 天桥的左右两 endpoint。
2.  $s$  或者  $t$  到某一个天桥的前驱和后继。

我们只保留关键点及关键点下方第一个位于天桥上的点，并跑最短路即可。 $O(n \log_2(n))$ 。

### 【提交记录】

<https://www.luogu.com.cn/record/85969190>。

## Koishi Loves Construction

### 【题目描述】

<https://www.luogu.com.cn/problem/P3599>。

Koishi 决定走出幻想乡成为数学大师！

Flandre 听说她数学学的很好，就给 Koishi 出了这样一道构造题：

Task1：试判断能否构造并构造一个长度为  $n$  的  $1 \dots n$  的排列，满足其  $n$  个前缀和在模  $n$  的意义下互不相同。

Task2：试判断能否构造并构造一个长度为  $n$  的  $1 \dots n$  的排列，满足其  $n$  个前缀积在模  $n$  的意义下互不相同。

### 【题解】

对于第一问，若  $n \neq 1$  且  $n$  为奇数，则由等差数列求和的式子可以看出无解；否则我们这样构造： $0, -1, 2, -3, 4, \dots$ （其实也就是一个 Zig-zag?）。

对于第二问，若  $n \neq 4$  且  $n$  不是素数，则因为会提早出现 0 而导致无解。否则，求出原根并用上面的构造即可（最后一位放 0）。

### 【提交记录】

<https://www.luogu.com.cn/record/85771488>。

## 【CF1523H】Hopping Around the Array

### 【题目描述】

<https://www.luogu.com.cn/problem/CF1523H>。

给定一个序列  $a_1, a_2, \dots, a_n$ ，在第  $i$  个点可以一步走到  $[i, i + a_i]$  中的任何一个点，多次询问，每一次从  $l$  出发，你可以删掉不超过  $k$  个点，将剩下的点重标号，使得到达  $r$  的步数最少。

$n \leq 2 \times 10^4$ ,  $k \leq 30$ 。

### 【题解】

题目的要求是  $i$  可以走向  $[i, i + a_i]$ ，这个  $i + a_i$  是非常关键的变量，我们一定是从  $i + a_i$  小的走向  $i + a_i$  大的；并且  $i$  可以走向  $i$ ，所以  $t$  秒钟能  $i$  走到的位置一定构成一个区间  $[i, j]$ 。

采取类似倍增的思路，设  $f_{i,j,k}$  表示从  $i$  出发走  $2^j$  秒用了  $k$  个删除，能走到的区间长度。那么转移时，需要维护一个  $f$  的 ST 表，预处理 ST 表的复杂度过大。

注意到，区间  $[i, i + f_{i,j-1,k_1})$  里最优的转移点，一定是  $i + a_i$  最大的点，我们只需要预处理出  $i + a_i$  序列的 ST 表即可。

### 【提交记录】

<https://www.luogu.com.cn/record/85766661>。

## 赵云八卦阵

### 【题目描述】

<https://uoj.ac/problem/703>。

给定一个序列  $a_1, \dots, a_n$ ，每次可以将某一项  $a_x$  ( $2 \leq x \leq n$ ) 变成  $a_x \oplus a_{x-1}$ ，求这个序列最长上升子序列长度的最大值。

$$n \leq 10^6, 1 \leq a_i < 2^{60}.$$

### 【题解】

首先,  $a_i$  之间相互独立, 且每个  $a_i$  可以成为  $a_i \oplus v$ , 其中  $v$  可以用  $a_{1 \sim i-1}$  线性表出。

注意到  $a_{1 \sim i}$  构成的线性基会至多分成 60 段, 我们找到这些段, 称段首为关键点。关键点的取值集合, 被它后面这一段的取值集合包含, 所以如果要将关键点算入答案, 那么后面这一段的所有非关键点都必然被算入答案 (否则可以调整)。进而得知选中的非关键点一定是一个后缀。

从后向前 dp, 记  $f_{i,j}$  表示后  $i$  段选择了  $j$  个关键点时, 最左边这个数的最大值, 那么借助在线性基上查排名、根据排名查数、查前驱, 即可转移。

线性基上排名和值互相转化: 求出线性基的最小表示, 将在基中的位提出来单独看, 则有排名 = 值, 于是便可以互相转化了。

线性基上查前驱: 即对于  $X$ , 求出最大的  $X \oplus v$ , 使得  $v \in B$  且  $X \oplus v < B$ 。从高到低位扫一遍, 若  $X \oplus B_i < X$  则  $X \leftarrow X \oplus B_i$ ; 再从高到低位扫一遍, 若  $X \oplus B_i < Y$ , 则  $X \leftarrow X \oplus B_i$ 。

### 【提交记录】

<https://uoj.ac/submission/581859> (最优解!)

## 【UR23】地铁规划

### 【题目描述】

<https://uoj.ac/problem/693>。

有一个长度为  $m$  的边序列, 你需要使用交互库提供的可撤销并查集接口, 在线地对于每个  $l$  求出最大的  $r$ , 使得区间  $[l, r]$  内的边不形成环。

$$m \leq 2 \times 10^5.$$

### 【题解】

Baka's Trick: 即谢队法, 可以支持  $l$  和  $r$  分别递增的优化建图、不具有可减性的双指针等等, 需要用两个栈/前缀和式数据结构, 当左栈全部弹出后, 再将右栈全部倒出来插入左栈, 如此往复推进。

单栈模拟队列: 用一个栈维护上述的双栈, 栈内每个元素分别标记为 0 或者 1, 其中 0 表示左栈元素, 0 元素构成的子序列应自栈顶向栈底递增, 1 元素应当自栈顶向栈底递减。每次顺利右移  $l$  需要保证栈顶为 0 元素, 记栈内中 0 元素的个数为  $\text{cnt}$ , 那么

不断地弹出栈顶，直到弹出  $\text{lowbit}(cnt)$  个 0 元素为止，然后先把 1 元素插回去，再把 0 元素插回去； $cnt = 0$  时，我们重构整个栈使得全部变为 0 元素。

可以发现总复杂度是对的：对于 0 元素： $\sum_{i=1}^k \text{lowbit}(i) = \sum_{i=0}^{\log_2(k)} 2^i \lfloor \frac{n}{2^i} \rfloor - \lfloor \frac{n}{2^{i+1}} \rfloor = O(n \log_2(n))$ ；对于 1 元素：也一样可以观察到只会被遍历  $O(\log_2(n))$  次。

## 【提交记录】

<https://uoj.ac/submission/581960>。

## 【ZJOI2022】众数

### 【题目描述】

<https://uoj.ac/problem/741>。

可怜得到了一个序列  $a_1, a_2, \dots, a_n$ ，她可以对这个序列使用一次超能力：选择一个区间  $[l, r] (1 \leq l \leq r \leq n)$  和一个整数  $k \in [-10^9, 10^9]$ ，将区间内的所有数  $a_l, a_{l+1}, \dots, a_r$  加上  $k$ 。

九条可怜很喜欢长得比较一致的序列，因此她希望最终的序列众数的出现次数尽可能多。给出序列  $a$ ，你需要输出最终序列的众数出现次数的最大值，并输出这个众数的所有可能取值。注意对于一个序列，众数的取值可能不止一个。

$$\sum n \leq 5 \times 10^5。$$

### 【题解】

对于每个数分别算出其最多出现的次数。

根号分治，对于出现次数大于根号的数配上另一个数：把序列中其它的值分别遍历一遍，求出其前缀和的差的最大值，求两次，分别用来更新这两者的答案。

对于出现次数小于根号的数配上出现次数小于根号的数：对于每个  $l$ ，预处理出  $f_{l,k}$  表示最小的  $r$  使得  $[l, r]$  内的众数出现次数达到  $k$ ，那么  $k$  只需要到  $\sqrt{n}$  就好。我们用一个桶套桶维护双指针，离线下所有的询问区间众数的  $[l, r]$  的双指针，扫上  $\sqrt{n}$  遍就好了。

### 【提交记录】

<https://uoj.ac/submission/581619>。

## 【BalticOI 2022 Day1】Uplifting Excursion

### 【题目描述】

<https://www.luogu.com.cn/problem/P8392>。

有  $2m+1$  种物品，重量分别为  $-m, -m+1, \dots, m-1, m$ 。重量为  $i$  的物品有  $a_i$  个。你需要拿走若干物品，使得这些物品重量之和恰好为  $l$ 。在此基础上，你需要拿尽可能多的物品。

问在物品重量之和恰好为  $l$  的基础上，你最多能拿多少物品。

$1 \leq m \leq 300, -10^{18} \leq l \leq 10^{18}, 0 \leq a_i \leq 10^{12}$ 。

### 【题解】

我们先选上所有数，然后贪心地删值最大数使得重量小于等于  $l$ （或者反过来），然后试着去调整，发现调整的过程中一定不会使得值超过  $m^2$ ，所以只要维护一个大小为  $m^2$  的背包即可。

### 【提交记录】

<https://www.luogu.com.cn/record/86035628>。

## 「Wdsr-2.7」八云蓝自动机

### 【题目描述】

<https://www.luogu.com.cn/problem/P7709>。

八云蓝自动机维护了一个长度为  $n$  的序列  $A$ ，每个元素都有一个初始值。同时自动机会支持以下三种操作：

- 1  $l\ r\ k$ ：将区间  $[l, r]$  内的所有数字全都变为  $k$ ，即  $A_l \leftarrow k, A_{l+1} \leftarrow k, \dots, A_r \leftarrow k$ 。
- 2  $x\ y$ ：交换  $A_x$  与  $A_y$  的值。
- 3  $x$ ：查询  $A_x$  的值。

为了测试八云蓝自动机的效率，紫需要进行非常非常多次的测试。为了生成每个测试的所有操作，紫构造出了一个长度为  $m$  的操作序列  $B$ ， $B$  中的元素就是八云蓝自动机可以执行的一个操作。

设  $\Upsilon(l, r)$  表示从初始状态开始，依次执行  $B_l, B_{l+1}, \dots, B_r$  操作后，所有操作 3 的结果之和。特别地，如果这些操作中没有操作 3，那么  $\Upsilon(l, r) = 0$ 。

紫会向八云蓝自动机发起  $q$  次询问，每次给出一组  $(l, r, p)$ ，八云蓝自动机需要计算出

$$\left( \sum_{i=l}^r \Upsilon(i, p) \right) \bmod 2^{32}$$

$$1 \leq n, m, q \leq 3 \times 10^5, 1 \leq a_i, k \leq 10^9; 1 \leq op \leq 3; 1 \leq x, y \leq n; x \neq y.$$

### 【题解】

我们考虑如何求出一个  $Ask(x)$  的值：找到它前面第一个涉及到它的修改，若其位置已经小于  $l$ ，那么直接返回  $A_x$ ；否则，若其为区间修改操作，则直接返回这个区间覆盖的值；若为交换操作，则返回递归返回  $Ask(y)$  的值。我们把所有的区间覆盖当作根节点，交换操作视作两个  $Ask()$  操作，把每个  $Ask$  连向它依赖的操作，形成了一个操作树。

在操作树上求  $Ask(x)$  的方法是，找到这个  $Ask()$  操作对应的节点，然后找到它上方的最浅的在给定区间内的节点  $grand$ ，然后对于那个点返回  $A_x$  或者返回区间覆盖的值。

我们考虑从小到大扫描一遍，固定  $p$ ，对每个  $l$  求出  $\sum_{i=l}^p \Upsilon(i, p)$ 。把这个拆成每个  $grand$  的贡献，贡献的形式一定是对区间  $(time(fat_{grand}), time(grand)]$  内分别贡献  $A_x$ 。

考虑根号重构，重构的部分用差分前缀和即可解决；散块的贡献，我们离线下来，用  $O(\sqrt{N}) - O(1)$  分块即可解决。复杂度  $O(n^{1.5})$ 。

### 【提交记录】

<https://www.luogu.com.cn/record/86045324>。

## 【CF859G】Circle of Numbers

### 【题目描述】

<https://www.luogu.com.cn/problem/CF859G>。

给定一个  $n$  次单位根构成的式子，问它是否等于 0。  $n \leq 10^5$ 。

### 【题解】

考虑其物理意义：等于 0 当且仅当其重心位于  $O$  点。直接用 double 当然是行不通的，我们多取几个质数，利用其单位根来判一下是否为 0 即可。

更为简洁的做法是考虑分圆多项式的性质，令  $a$  循环卷积卷上  $\prod_{p|n} (1 - x^p)$ ，若等于 0 则原式为 0。  $O(nw)$ 。

**【提交记录】**

<https://www.luogu.com.cn/record/86056257>。

**【CF1344E】 Train Tracks****【题目描述】**

<https://www.luogu.com.cn/problem/CF1344E>。

给定  $n$  个点的一棵树，以 1 为根，边有边权。

有  $m$  辆从 1 开始到  $s_i$  的火车，每个火车有一个初始时刻  $t_i$ ，在初始时刻时从根出发，向着目标  $s_i$  前进，当火车在  $x$  时刻到达一个点  $u$  时，假设下一个路径上的点是  $v$ ，两点间边权是  $d$ ，则在  $x + d$  时刻到达  $v$ ，火车在到达目标点后停止。

由于每个点可能可以到达多个点，每个点有且仅有一个当前时刻可以到的儿子，每秒钟只能切换某一个点可以到的儿子，切换比火车开动先进行，若一辆火车走向了非目标方向的点，则立刻自爆。每个点初始能到的儿子是确定的。求出能否不发生自爆，如果不能，第一次自爆最晚什么时候发生？在此基础上，至少切换多少次一个点可以到的儿子。

$n, m \leq 10^5$ 。

**【题解】**

我们可以把每辆火车拆成一些限制，然后用贪心来 check 这些性质能否同时被满足。但是限制有点多！

我们注意到这题中每个点恰有一个出边的性质和 LCT 是一样的，所以结合 access 的复杂度分析，限制的总量是  $O(n \log_2(n))$  的。

**【提交记录】**

<https://www.luogu.com.cn/record/86078364>。

**【CF765G】 Math, math everywhere****【题目描述】**

<https://www.luogu.com.cn/problem/CF765G>。

给定  $N$  的因式分解形式  $\prod_{i=1}^n p_i^{\alpha_i}$  和一个长为  $m$  的字符串  $s$ 。

求  $k$  的个数，使得  $0 \leq k < N$  且

$$\forall i \in [0, m), \gcd(k + i, N) = 1 \text{ if and only if } s_i = 1$$

模  $10^9 + 7$ 。  $m \leq 40$ ,  $n \leq 5 \times 10^5$ ,  $p_i, \alpha_i \leq 10^9$ 。

### 【题解】

首先我们发现  $s$  只和  $k$  模每个  $p_i$  的余数有关，所以  $\alpha_i$  是没有用的，所以直接把答案乘上  $p_i^{\alpha_i - 1}$  即可，然后每个素数其实就相当独立了，每个素数的作用是使得  $s_i$  中的一些位变成 0，于是就有了一个  $2^{40} \cdot n$  的暴力 dp。分为三段优化：

- $p \in [1, \frac{k}{2})$ : 直接暴力用 HashMap 和队列维护这个 dp。
- $p \in [\frac{k}{2}, m)$ : 我们发现中间几位的具体状态是没用的，我们直接记下中间几位中还有多少个数需要变成 0 即可。
- $p \geq m$ : 我们直接记下一共有多少个数需要变成 0 即可。  
及时在 dp 中去除无用信息，以使得状态总量可控。

### 【提交记录】

<https://www.luogu.com.cn/record/86093959>（在洛谷上仅次于 Bot）。

## 【CF1710E】Two Arrays

### 【题目描述】

<https://www.luogu.com.cn/problem/CF1710E>。

现有两个整数数组  $a_1, a_2, \dots, a_n$  和  $b_1, b_2, \dots, b_m$ 。Alice 和 Bob 将要玩一个游戏，Alice 先手，然后他们轮流进行操作。他们在一个  $n \times m$  的网格上进行游戏（网格有  $n$  排  $m$  列）。刚开始，有一个棋子放在网格的  $(1, 1)$  上。在 Alice 或 Bob 轮次中，玩家可以选择以下两个动作中的一个进行操作：

1. 将棋子移动到一个不同的格子上，该格子必须和棋子的原位置在同排或者同列上。玩家不能将棋子移动到已经被访问过 1000 次的格子上。
2. 以  $a_r + b_c$  的得分立刻结束游戏， $(r, c)$  表示棋子当前所在的单元格。

Bob 想要最大化自己的得分，Alice 则想要最小化自己的得分。如果他们都以最佳方式玩这个游戏，则最终的得分是多少？  $n, m \leq 10^3$ 。

### 【题解】

我们可以二分一个  $mid$ ，使得  $a_r + b_c \leq mid$  的状态为 Alice 的必胜态， $a_r + b_c > mid$  的状态为 Bob 的必胜态，双方都要避免走入对方的必胜态，这就变成了一个二分图博弈。

注意到 1000 次是没有什么作用的，因为只要来回消耗一下，就能消成 1。而二分图最大匹配很难求，考虑转化为最大独立集。

我们枚举一个点  $(x, y)$ ，在其左上方全取 Alice 的必胜点，右下全取 Bob 的必胜点，假设我们能枚举所有  $(x, y)$  就可以求出最大独立集。而事实上， $x$  递增时最优的  $y$  也递增，满足凸性，所以直接双指针维护一下即可。求出有  $(1, 1)$  和没有  $(1, 1)$  的最大独立集，判断其是否相等即可。

### 【提交记录】

<https://www.luogu.com.cn/record/86105128>。

## 【POI2006】PAL-Palindromes

### 【题目描述】

<https://www.luogu.com.cn/problem/P3449>。

他写下了  $n$  个回文串，随后将这些串两两组合，合并成一个新串。容易看出，一共有  $n^2$  个新串。

两个串组合时顺序是任意的，即 ‘a’ 和 ‘b’ 可以组合成 ‘ab’ 和 ‘ba’，另外自己和自己组合也是允许的。

现在他想知道这些新串中有多少个回文串，你能帮帮他吗？ $\sum len \leq 2 \times 10^6$ 。

### 【题解】

两个回文串拼起来是回文串，当且仅当两个串的最小严格周期内的字符完全相等。

### 【提交记录】

<https://www.luogu.com.cn/record/86122697>。

## 【CF1693E】Outermost Maximums

### 【题目描述】

<https://www.luogu.com.cn/problem/CF1693E>。

有一个长度为  $n + 2$  的序列  $a$ ，其中  $a_0 = a_{n+1} = 0$ ，其余元素均给定。你可以进行下面两种操作任意次：

1. 设  $x$  表示序列  $a$  最靠左的最大值的位置，则令  $a_x \leftarrow \max_{i=0}^{x-1} a_i$ 。2. 设  $y$  表示序列  $a$  最靠右的最大值的位置，则令  $a_y \leftarrow \max_{i=y+1}^{n+1} a_i$ 。

你需要求出使序列  $a$  的所有元素均变成 0 所需的最少的操作总次数。  $n \leq 10^5$ 。

### 【题解】

首先有一个  $O(n^2)$  的贪心：每次把一个数变成其左边或者右边的最大值中比较小的一个。考虑优化。

一种是从贡献的角度去优化：我们考虑一个数的贡献次数，在数轴上开一个线段树，记它左边的数为黑点，右边的数为白点，那么它每次是变成一个距离它最近的白点或者距离它最近的黑点，然后问走到 0 的最优步数。我们直接用线段树维护区间矩阵连乘积，然后依次把黑点变成白点即可。

另一种优化是尝试去强硬的优化其过程：贪心时我们不立刻确定那个最大值的方向，而是把这些最大值标记为“不确定”。再去考虑次大值，设次大值最左边在  $l$ ，最右边在  $r$ ，则会将所有  $> l$  的“不确定”转化为“向右”，将所有  $< r$  的“不确定”转化为“向左”；将所有  $< r$  的“向右”计算上贡献并且改为不确定，再将所有  $> l$  的向左计算上贡献并且改为“不确定”。于是我们发现“向左”都是一个前缀，“向右是一个后缀”，“不确定”是中间的一段。直接用树状数组维护区间和，再用变量  $[ql, qr]$  维护“不确定”的两端点即可。

### 【提交记录】

<https://www.luogu.com.cn/record/86133353>。

## 色多项式

### 【题目描述】

<https://loj.ac/p/6787>。

对于一个无向图  $G = (V, E)$ ，色多项式  $P(x)$  是一个  $|V|$  次多项式，对于任何正整数  $k$ ， $P(k)$  为  $G$  的顶点的  $k$  染色的数量。

给定一个图  $G$ ，请你计算出  $G$  的色多项式系数。  $n \leq 21$ 。

### 【题解】

一种是  $2^n \cdot n^3$  集合幂级数做法，大力卡常！

我的办法是，用广义串并联图缩点，对于每条边维护其两端点相同和两端点不同时的色多项式；然后爆搜，从小到大搜颜色，用一个桶维护多项式的和，然后再乘上组合

数对应的多项式。实际上当然是可以卡掉的，但是出题人没卡，最优解！

### 【提交记录】

<https://loj.ac/s/1576105>。