组合计数

Appleblue17

2025.7.16

Overview

今天会讲什么:

- 组合恒等式
- 多项式和形式幂级数
- 生成函数和组合类基础
- Raney 引理和广义二项级数
- 容斥原理
- 多项式相关技巧(d-finite 级数, q-analog)
- 很多妙妙杂题 (?)

Overview

以后（可能）会讲什么：

- 多项式工业技巧
- 线性代数相关
  - 特征多项式和线性递推
  - 矩阵树定理，BEST 定理，LGV 引理
  - q-analog 矩阵计数，子空间反演
- FWT 和集合幂级数
- 数论函数和 DGF
- 概率论与 PGF
- 群论基础，Polya 定理，组合对象符号化
- 复合逆和拉格朗日反演
- 特殊数列（斯特拉数，伯努利数，分拆数等）
- Prüfer 序列，杨表

组合数

在 n 个元素中选择 m 个的方案数，记作 $\binom{n}{m}$，定义为：

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}$$

其中 n, m 为非负整数。
当 m 为非负整数时，可以拓展定义：

$$\binom{n}{m} = \frac{n^{\underline{m}}}{m!}$$

其中 n 为任意实数甚至复数。

组合恒等式

- 对称恒等式：$\binom{n}{m} = \binom{n}{n-m}$，其中 $n \geq 0$。
- 吸收/释放恒等式：$\frac{n}{m} = \frac{n}{m}\binom{n-1}{m-1}$。
- 加法公式：$\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$。
- 上指标反转：$\binom{n}{m} = (-1)^m\binom{n-m-1}{m}$。

组合恒等式

- 平行求和: $\sum_{k=0}^{m}\binom{n+k}{k}=\binom{n+m+1}{m}$, n 为非负整数。
- 二项式定理: $(1+x)^r=\sum_{k\geq 0}\binom{n}{k}x^k y^{n-k}$, 当 n 不为非负整数时需要 $|x|\leq 1$ 来保证收敛。
- 上指标求和: $\sum_{k=0}^{n}\binom{k}{m}=\binom{n+1}{m+1}$, n 为非负整数。这个式子两边乘上 m 就可以得到有限积分的结果。

范德蒙德卷积

$\sum_{k=0}^{n}\binom{a}{k}\binom{b}{n-k}=\binom{a+b}{n}$，$n$ 为非负整数。

组合证明：左边 a 个球，右边 b 个球，一共选 n 个球，相当于枚举左边选 i 个球，右边选 $n-i$ 个球。

代数证明：

$$
\begin{aligned}
LHS &= \sum_{i=0}^{n}([x^i](1+x)^a)([x^{n-i}](1+x)^b) \\
&= [x^n](1+x)^{a+b} \\
&= \binom{a+b}{n}
\end{aligned}
$$

反范德蒙德卷积

$\sum_{k=0}^{n}\binom{k}{a}\binom{n-k}{b}=\binom{n+1}{a+b+1}$，$n$ 为非负整数。

组合证明：共 $n+1$ 个球排成一列，选出 $a+b+1$ 个球，相当于枚举第 $a+1$ 个球在 $i+1$ 处。

代数证明：

$$\begin{aligned}
LHS &= \sum_{i=0}^{n}([x^i]\left(\frac{x^a}{(1-x)^{a+1}}\right))([x^{n-i}]\left(\frac{x^b}{(1-x)^{b+1}}\right)) \\
&= [x^n]\left(\frac{x^{a+b}}{(1-x)^{a+b+2}}\right) \\
&= \binom{n+1}{a+b+1}
\end{aligned}$$

例子

$$\sum_{k\le m}(-1)^k\binom{n}{k}=(-1)^m\binom{n-1}{m}$$

例子

$$\sum_{k\le m}(-1)^k\binom{n}{k}=(-1)^m\binom{n-1}{m}$$

上指标反转：

$$S=\sum_{k=0}^{m}\binom{k-n-1}{k}$$

使用平行求和：

$$S=\binom{m-n}{m}$$

再次上指标反转：

$$S=(-1)^m\binom{m-(m-n)-1}{m}=(-1)^m\binom{n-1}{m}$$

例子

$$\sum_{k=0}^{m} \frac{\binom{m}{k}}{\binom{n}{k}}, \text{ where } 0 \le m \le n$$

例子

$$\sum_{k=0}^{m} \frac{\binom{m}{k}}{\binom{n}{k}}, \text{ where } 0 \le m \le n$$

$$\begin{aligned} S &= \sum_{k=0}^{m} \frac{m!}{k!(m-k)!} \cdot \frac{k!(n-k)!}{n!} = \sum_{k=0}^{m} \frac{m!}{(m-k)!} \cdot \frac{(n-k)!}{n!} \\ &= \sum_{k=0}^{m} \binom{n-k}{m-k} \Big/ \binom{n}{m} = \binom{n}{m}^{-1} \sum_{k=0}^{m} \binom{n-m+k}{k} \\ &= \binom{n}{m}^{-1} \binom{n+1}{m} = \frac{n+1}{n+1-m} \end{aligned}$$

[THU Summer Bootcamp 2025 A] Acoustic String

给定长为 n 的 01 字符串 s，对其进行 $n-1$ 次以下操作：

- 对所有 $i=1,2,\cdots,n-1$，令 $s_i^* = s_i \oplus s_{i+1}$；
- 令 $s \leftarrow s^*$。

求经过 $n-1$ 次操作后的字符串 s.

数据范围：$1 \le n \le 10^6$。

考虑转化成格路，于是 s_i 的贡献就是 $\binom{n-1}{i} \bmod 2$。

$$Ans = \sum_{i=0}^{n-1} \binom{n-1}{i} s_i \bmod 2$$

用 Lucas 定理计算组合数 $\mathrm{mod} 2$ 即可。

[ARC201B] Binary Knapsack

n 个物品，第 i 个物品的重量为 2^{X_i}，价值为 Y_i。求容量为 W 的背包能装下的物品的最大价值。

数据范围：$T \le 10^5, \sum n \le 2 \times 10^5$，$1 \le w \le 10^18$，$0 \le X_i < 60$，$1 \le Y_i \le 10^9$。

[ARC201B] Binary Knapsack

n 个物品，第 i 个物品的重量为 2^{X_i}，价值为 Y_i。求容量为 W 的背包能装下的物品的最大价值。

数据范围：$T \le 10^5, \sum n \le 2 \times 10^5$，$1 \le w \le 10^18$，$0 \le X_i < 60$，$1 \le Y_i \le 10^9$。

- Hint 1：不是背包

[ARC201B] Binary Knapsack

n 个物品，第 i 个物品的重量为 2^{X_i}，价值为 Y_i。求容量为 W 的背包能装下的物品的最大价值。

数据范围：$T \le 10^5, \sum n \le 2 \times 10^5$，$1 \le w \le 10^18$，$0 \le X_i < 60$，$1 \le Y_i \le 10^9$。

- Hint 1：不是背包
- Hint 2：逐位考虑

假设再加入无限个重量为 1，价值为 0 的物品，那么就一定能装满背包。
从低位往高位考虑：

- 如果 $2 \mid W$，那么所有 $X_i = 0$ 的物品中必须选偶数个。将它们从大到小排序并两两分组，就可以将每一组 Y_i, Y_{i+1} 看成一个新的重量为 2，价值为 $Y_i + Y_{i+1}$ 的物品。
- 如果 $2 \nmid W$，那么就选出 $X_i = 0$ 中价值最大的那个，然后将剩下的按照上述方法两两分组。

一直递归做下去即可。时间复杂度 $O(n \log n)$.

[3rdUCup Stage13 D] And DNA

给定 N 和 M，求长度为 N 的整数序列 $A = (A_1, A_2, \ldots, A_N)$ 的数量，使得：

- $A_i \in [0, M]$；
- 对于所有 $i = 1, 2, \ldots, N$，有 $A_i + (A_{i-1} \& A_{i+1}) = M$，其中 $A_0 \triangleq A_N$ 且 $A_{N+1} \triangleq A_1$。

答案对 998244353 取模。

数据范围：$3 \le N \le 10^9$，$0 \le M \le 10^9$。

[3rdUCup Stage13 D] And DNA

给定 N 和 M，求长度为 N 的整数序列 $A = (A_1, A_2, \ldots, A_N)$ 的数量，使得：

- $A_i \in [0, M]$；
- 对于所有 $i = 1, 2, \ldots, N$，有 $A_i + (A_{i-1} \& A_{i+1}) = M$，其中 $A_0 \triangleq A_N$ 且 $A_{N+1} \triangleq A_1$。

答案对 998244353 取模。

数据范围：$3 \le N \le 10^9$，$0 \le M \le 10^9$。

- Hint 1：诈骗

[3rdUCup Stage13 D] And DNA

给定 N 和 M，求长度为 N 的整数序列 $A = (A_1, A_2, \ldots, A_N)$ 的数量，使得：

- $A_i \in [0, M]$；
- 对于所有 $i = 1, 2, \ldots, N$，有 $A_i + (A_{i-1} \& A_{i+1}) = M$，其中 $A_0 \triangleq A_N$ 且 $A_{N+1} \triangleq A_1$。

答案对 998244353 取模。

数据范围：$3 \le N \le 10^9$，$0 \le M \le 10^9$。

- Hint 1：诈骗
- Hint 2：逐位考虑

诈骗题。从低位往高位考虑，先考虑最低位：

- 如果 $2 \mid M$，那么只要存在 A_i 的最低位为 1，则 A_{i-1} 和 A_{i+1} 的最低位也必须为 1；由此推出所有 A_i 的最低位都为 1。也就是说，所有 A_i 的最低位都是一样的。
- 如果 $2 \nmid M$，它们对高位的影响就确定了。对于此层的要求即**不能有三个连续的** 1，直接矩阵快速幂优化 DP 即可。

最后，设 $f_{t,0/1}$ 表示当前第 t 位，低位有无进位的方案数，按照上述的转移即可。时间复杂度 $O(\log N + \log M)$.

[Jilin 2023 H] Games on the Ads 2: Painting

给定一个 $n \times n$ 的网格，每行和每列各有一支画笔，共 $2n$ 支画笔。
每支画笔有一个 1 到 n 中的颜色，且所有行画笔和所有列画笔都分别是一个长为 n 的排列 $\{p\},\{q\}$。使用行画笔会将对应行的所有单元格涂成该画笔指定的颜色，使用列画笔同理。
给定最终的目标图案 $\{c_{i,j}\}_{1\le i,j\le n}$，求有多少种不同的画笔使用顺序，使得使用每个画笔恰好一次后，使得网格最终与给定的目标图案一致。
答案对 998244353 取模。

数据范围：$1 \le n \le 20$，p,q 是长为 n 的排列，$1 \le c_{i,j} \le n$

[Jilin 2023 H] Games on the Ads 2: Painting

给定一个 $n \times n$ 的网格，每行和每列各有一支画笔，共 $2n$ 支画笔。
每支画笔有一个 1 到 n 中的颜色，且所有行画笔和所有列画笔都分别是一个长为 n 的排列 $\{p\}, \{q\}$。使用行画笔会将对应行的所有单元格涂成该画笔指定的颜色，使用列画笔同理。
给定最终的目标图案 $\{c_{i,j}\}_{1 \le i,j \le n}$，求有多少种不同的画笔使用顺序，使得使用每个画笔恰好一次后，使得网格最终与给定的目标图案一致。
答案对 998244353 取模。

数据范围：$1 \le n \le 20$，p, q 是长为 n 的排列，$1 \le c_{i,j} \le n$

- Hint 1：图

[Jilin 2023 H] Games on the Ads 2: Painting

给定一个 $n \times n$ 的网格，每行和每列各有一支画笔，共 $2n$ 支画笔。
每支画笔有一个 1 到 n 中的颜色，且所有行画笔和所有列画笔都分别是一个长为 n 的排列 $\{p\},\{q\}$。使用行画笔会将对应行的所有单元格涂成该画笔指定的颜色，使用列画笔同理。
给定最终的目标图案 $\{c_{i,j}\}_{1\leq i,j\leq n}$，求有多少种不同的画笔使用顺序，使得使用每个画笔恰好一次后，使得网格最终与给定的目标图案一致。
答案对 998244353 取模。

数据范围：$1 \leq n \leq 20$，p, q 是长为 n 的排列，$1 \leq c_{i,j} \leq n$

- Hint 1：图
- Hint 2：几乎是完全二分图

显然每个格子都会恰好被行染一次，被列染一次。讨论一下情况：

- 如果 $c_{i,j} \neq p_i, c_{i,j} \neq q_j$，那么整个无解。
- 如果 $p_i = q_j$，那么无论怎么染这个格子都合法。
- 否则如果 $c_{i,j} = p_i$，那么必须先染列再染行；反之亦然。

于是问题转化为给定一个 $n + n$ 的二分图，且每个点度数均为 $n - 1$，求拓扑排序数量。

少了 n 条边让人很苦恼，直接大力枚举 2^n 种给边定向，这样就变成了完全二分图。

这意味着如果确定了列的染色顺序，每个行插入的位置是固定的。

设 S_i 表示行 i 被连向的点的集合（即必须在它之前染的列号）。将所有 S_i 按照集合大小排序，设为 $S_{i_1}, S_{i_2}, \cdots, S_{i_n}$。

- 如果存在 $S_{i_t} \not\subseteq S_{i_{t+1}}$，那么无解。
- 否则，设大小为 k 的集合有 c_k 个，那么方案数为 $\prod_{k=1}^{n} c_k!$。

时间复杂度 $O(n2^n)$.

[ICPC2023 Shanghai C] Equal Sums

给定 $\{l_i^{(x)}, r_i^{(x)}\}_{i=1}^n$ 和 $\{l_j^{(y)}, r_j^{(y)}\}_{j=1}^m$，求序列 $\{x_1, \ldots, x_n, y_1, \ldots, y_m\}$ 的数量，使得：

- 对于所有 $1 \le i \le n$，$l_i^{(x)} \le x_i \le r_i^{(x)}$；
- 对于所有 $1 \le j \le m$，$l_j^{(y)} \le y_j \le r_j^{(y)}$；
- $\sum_{i=1}^n x_i = \sum_{j=1}^m y_j$.

答案对 998244353 取模。

数据范围：$1 \le n, m \le 500$，$1 \le l_i^{(x)} \le r_i^{(x)} \le 500, 1 \le l_j^{(y)} \le r_j^{(y)} \le 500$。

[ICPC2023 Shanghai C] Equal Sums

给定 $\{l_i^{(x)}, r_i^{(x)}\}_{i=1}^n$ 和 $\{l_j^{(y)}, r_j^{(y)}\}_{j=1}^m$，求序列 $\{x_1, \dots, x_n, y_1, \dots, y_m\}$ 的数量，使得：

- 对于所有 $1 \le i \le n$，$l_i^{(x)} \le x_i \le r_i^{(x)}$；
- 对于所有 $1 \le j \le m$，$l_j^{(y)} \le y_j \le r_j^{(y)}$；
- $\sum_{i=1}^n x_i = \sum_{j=1}^m y_j$.

答案对 998244353 取模。

数据范围：$1 \le n, m \le 500$，$1 \le l_i^{(x)} \le r_i^{(x)} \le 500, 1 \le l_j^{(y)} \le r_j^{(y)} \le 500$。

- Hint 1：如何控制值域？

[ICPC2023 Shanghai C] Equal Sums

给定 $\{l_i^{(x)}, r_i^{(x)}\}_{i=1}^n$ 和 $\{l_j^{(y)}, r_j^{(y)}\}_{j=1}^m$，求序列 $\{x_1, \dots, x_n, y_1, \dots, y_m\}$ 的数量，使得：

- 对于所有 $1 \le i \le n$，$l_i^{(x)} \le x_i \le r_i^{(x)}$；
- 对于所有 $1 \le j \le m$，$l_j^{(y)} \le y_j \le r_j^{(y)}$；
- $\sum_{i=1}^n x_i = \sum_{j=1}^m y_j$.

答案对 998244353 取模。

数据范围：$1 \le n, m \le 500$，$1 \le l_i^{(x)} \le r_i^{(x)} \le 500, 1 \le l_j^{(y)} \le r_j^{(y)} \le 500$。

- Hint 1：如何控制值域？
- Hint 2：作差

记值域为 w 并认为 n, m 同阶，直接背包的话和的值域能够达到 $O(nw)$，统计每个答案的复杂度也是 $O(nw)$，于是总复杂度是 $O(n^3w)$。
注意到最后需要维护的信息仅仅是 $\sum x_i = \sum y_j \Leftrightarrow \sum x_i - \sum y_j = 0$，记这个差为 d，我们实际上能够通过调整两侧加入的顺序来将 d 控制在 $[-w, w]$。

具体而言，考虑若有 $\sum_{i=1}^{a} x_i - \sum_{j=1}^{b} y_j = 0$，可以按如下算法构造一条从 $(0,0)$ 到 (a,b) 的路径：

1. 设变量 $i=0, j=0$ 表示当前位置，$d=0$ 表示两侧和之差。
2. 若 $d \le 0$，令 $d \leftarrow d + x_{i+1}, i \leftarrow i+1$.
3. 若 $d > 0$，令 $d \leftarrow d - y_{j+1}, j \leftarrow j+1$.
4. 当 $i=a, j=b$ 时停止。可以发现 $i=a$ 时 i 不会再增加，$j=b$ 时 j 不会再增加，故一定能够停止。
5. 由此构造出三元组序列 $P=\{i_t, j_t, d_t\}$.

可以发现一个合法的 P 一定对应唯一的 $(\{x_i\},\{y_i\})$，故其构成双射。
整个过程中 d 都被控制在 $[-w,w]$ 内，于是设 $dp_{i,j,d}$ 表示上述算法进行到状态 (i,j,d) 的方案数，用一点前缀和优化转移。时间复杂度 $O(nmw)$.

[ECFinal 2019 B] Black and White

给定一个 $n \times m$ 的棋盘，当 $i \equiv j \pmod 2$ 时，格子 (i, j) 为白色，否则为黑色。
沿着格线行走，从左下角 $(0, 0)$ 走到右上角 (n, m)，每步只能向右或向上。最后路径划出左侧的格子中白色格子数减去黑色格子数定义为路径的得分。
求得分恰好为 k 的路径数量，答案对 998244353 取模。

数据范围：$1 \leq T \leq 100$，$1 \leq n \leq 10^5$，$1 \leq m \leq 10^5$，$-10^5 \leq k \leq 10^5$。

[ECFinal 2019 B] Black and White

给定一个 $n \times m$ 的棋盘，当 $i \equiv j \pmod 2$ 时，格子 (i, j) 为白色，否则为黑色。
沿着格线行走，从左下角 $(0, 0)$ 走到右上角 (n, m)，每步只能向右或向上。最后路径划出左侧的格子中白色格子数减去黑色格子数定义为路径的得分。
求得分恰好为 k 的路径数量，答案对 998244353 取模。

数据范围：$1 \le T \le 100$，$1 \le n \le 10^5$，$1 \le m \le 10^5$，$-10^5 \le k \le 10^5$。

- Hint 1：两步两步考虑

经过多番尝试之后，一步一步考虑基本上不可能很好地挖掘性质。一个好的想法是两步两步考虑。
首先对于考虑 $2 \nmid n+m$ 的情况，可以枚举第一步来转化为 $2 \mid n+m$。那么现在 $n+m$ 步被划分为了 $(n+m)/2$ 组。记向上走为 A，向右走为 B，考虑两种贡献计算方式：

1. 向右走时不计算贡献，向上走时计算左侧的贡献。容易发现 AA 和 BB 始终不产生贡献，而剩下两种：

$(x \bmod 2, y \bmod 2)$	BA	AB
$(0,0)$	1	0
$(1,1)$	0	-1

2. 向上走时不计算贡献，向右走时计算上方的贡献。这时贡献的计算与 n 奇偶性有关：

$(x \bmod 2, y \bmod 2)$	BA$(2 \mid n)$	AB$(2 \mid n)$	BA$(2 \nmid n)$	AB$(2 \nmid n)$
$(0,0)$	0	-1	1	0
$(1,1)$	1	0	0	-1

看起来没有规律很难处理，但是如果我们把两种计算方式加起来（算两次），就会发生神奇的事情：

$(x \bmod 2, y \bmod 2)$	$\mathsf{BA}(2 \mid n)$	$\mathsf{AB}(2 \mid n)$	$\mathsf{BA}(2 \nmid n)$	$\mathsf{AB}(2 \nmid n)$
$(0,0)$	1	-1	2	0
$(1,1)$	1	-1	0	-2

- 先考虑 $2 \mid n, m$，此时贡献已经跟位置没有关系了，直接枚举 BA 填了 t 个，AB 填了 $t-k$ 个，可以算出 AA 和 BB 的数量，然后直接可重排：

$$Ans = \sum_{t=\max\{0,2k\}}^{\min\{n,m\}/2+k} \binom{(n+m)/2}{t, t-2k, n/2-t+k, m/2-t+k}$$

- 对于 $2 \nmid n, m$，注意到如果给每个 $(0,0)$ 减一，给每个 $(1,1)$ 加一，就可以归到前一种情况。而 $(0,0)$ 一定比 $(1,1)$ 多一，因此先把 k 减一再照前一种情况做即可。

时间复杂度 $O(T\min\{n,m\})$.

[CCPC2024 Harbin H] Subsequence Counting

给定一个长为 m 的序列 $\{t\}$ 和一个长度为 L 的序列 $\{s_i\}_{i=0}^{L-1}$，其中 $\{s\}$ 由 n 个连续段组成，从左到右第 i 段包含 l_i 个数字 v_i.
给定 $k \in [0, L)$ 且保证 $\gcd(k, L) = 1$，构造序列 $\{s'\}$ 使得 $s'_{i\cdot k \bmod L} = s_i$.
求 $\{t\}$ 作为 $\{s'\}$ 的子序列出现的次数，答案对 998244353 取模。
也即，求 m 元组 $(i_1, i_2, \ldots, i_m)$ 的数量，使得
$0 \le i_1 < i_2 < \ldots < i_m < L$ 且 $t_j = s'_{i_j}$.

数据范围：$1 \le n \le 2 \times 10^3$，$1 \le m \le 10$，$1 \le k < L \le 10^9$，$\gcd(k, L) = 1$，$1 \le t_i \le 10^3$，$1 \le l_i \le 10^9$，$1 \le v_i \le 10^3$.

[CCPC2024 Harbin H] Subsequence Counting

给定一个长为 m 的序列 $\{t\}$ 和一个长度为 L 的序列 $\{s_i\}_{i=0}^{L-1}$，其中 $\{s\}$ 由 n 个连续段组成，从左到右第 i 段包含 l_i 个数字 v_i.
给定 $k \in [0, L)$ 且保证 $\gcd(k, L) = 1$，构造序列 $\{s'\}$ 使得 $s'_{i \cdot k \mod L} = s_i$.
求 $\{t\}$ 作为 $\{s'\}$ 的子序列出现的次数，答案对 998244353 取模。
也即，求 m 元组 $(i_1, i_2, \ldots, i_m)$ 的数量，使得
$0 \leq i_1 < i_2 < \ldots < i_m < L$ 且 $t_j = s'_{i_j}$.

数据范围：$1 \leq n \leq 2 \times 10^3$，$1 \leq m \leq 10$，$1 \leq k < L \leq 10^9$，$\gcd(k, L) = 1$，$1 \leq t_i \leq 10^3$，$1 \leq l_i \leq 10^9$，$1 \leq v_i \leq 10^3$.

- Hint 1：矩阵

[CCPC2024 Harbin H] Subsequence Counting

给定一个长为 m 的序列 $\{t\}$ 和一个长度为 L 的序列 $\{s_i\}_{i=0}^{L-1}$，其中 $\{s\}$ 由 n 个连续段组成，从左到右第 i 段包含 l_i 个数字 v_i.
给定 $k \in [0, L)$ 且保证 $\gcd(k, L) = 1$，构造序列 $\{s'\}$ 使得 $s'_{i \cdot k \bmod L} = s_i$.
求 $\{t\}$ 作为 $\{s'\}$ 的子序列出现的次数，答案对 998244353 取模。
也即，求 m 元组 $(i_1, i_2, \ldots, i_m)$ 的数量，使得
$0 \le i_1 < i_2 < \ldots < i_m < L$ 且 $t_j = s'_{i_j}$.

数据范围：$1 \le n \le 2 \times 10^3$，$1 \le m \le 10$，$1 \le k < L \le 10^9$，$\gcd(k, L) = 1$，$1 \le t_i \le 10^3$，$1 \le l_i \le 10^9$，$1 \le v_i \le 10^3$.

- Hint 1：矩阵
- Hint 2：欧几里德算法

首先把逆变换变成正的。求出 k 在模 L 意义下的逆元 d，则 $s'_i = s_{id \bmod k}$.
注意到 m 很小，这启发我们可以把 DP 写成矩阵形式，每个矩阵大小是 $m+1$. 那么问题变成：现有 n 段 L 个矩阵，求它们在经过变换重排后的乘积。如果记原本的矩阵序列为 $s_0, s_1, \cdots, s_{L-1}$，那么就是要求 (以下均省略下标的模 L)：

$$s_0 s_d s_{2d} s_{3d} \cdots s_{(L-1)d}$$

记这个问题为 $f(L, d, n)$，神奇的是，这竟然可以用类似欧几里德的方法进行递归。

考虑将原本序列每隔 d 个就换一次行，类似：

$$\begin{array}{ccccccc} s_0 & s_1 & s_2 & \cdots & s_{(L-1) \bmod d} & \cdots & s_{d-1} \\ s_d & s_{d+1} & s_{d+2} & \cdots & s_{(L-1) \bmod d+d} & \cdots & s_{2d-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ s_{kd} & s_{kd+1} & s_{kd+2} & \cdots & s_{L-1} & & \end{array}$$

那么，问题就变成了：从左到右对每一列求乘积，再按照跳跃顺序将这些列的乘积相乘，跳跃顺序为 $0, d-(L \bmod d), 2[d-(L \bmod d)], \cdots$.
而由于原本只有 n 段矩阵，所以乘积不一样的列只有至多 $n+1$ 个，并且可以在找出所有断点后用线段树维护。于是可以获得每一列的乘积。
这样就变成了一个子问题 $f(d, d-(L \bmod d), n+1)$，可以递归下去。

这里有一个小问题：如果 d 很接近 L，那么递归的子问题规模并没有减少太多。不过注意到此时我们可以令 d 变成 $L-d$，即：

$$s_0 s_{-d} s_{-2d} s_{-3d} \cdots s_{-(L-1)d} = s_0 s_{(L-1)d} s_{(L-2)d} s_{(L-3)d} \cdots s_d$$

也即把 s_0 后面的矩阵顺序翻转。这很容易实现，并且至多只会让段数增加 1.

于是这样，每两次递归后 d 至少会减半，递归深度至多 $O(\log L)$，并且矩阵增加的端数也是至多 $O(\log L)$。每次递归时需要用线段树维护区间乘矩阵，总复杂度是 $O(m^3 n \log n \log L)$.

Taylor 级数

Definition (Taylor Series)

如果函数 $f(x) : \mathbb{R} \to \mathbb{R}$ 在 x_0 处无限可微，则其 Taylor 级数为：

$$f(x) = \sum_{n=0}^{+\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n$$

如果 $f(x)$ 在 x_0 处的 Taylor 级数收敛到 $f(x)$，则称 $f(x)$ 在 x_0 处是可解析的。

初等函数的 Taylor 级数在其定义域内是可解析的。

多项式和形式幂级数

Definition (多项式)

定义在 $\mathbb{Q}$ 上的多项式 $P(x)$ 形如:

$$P(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

Definition (形式幂级数)

定义在 $\mathbb{Q}$ 上的形式幂级数 $F(x)$ 形如:

$$F(x) = \sum_{i=0}^{+\infty} a_i x^i$$

多项式和形式幂级数

我们可以通过维护函数在 0 处的 Taylor 展开（即 Maclaurin 级数）的 n 阶近似来对其进行运算和处理。也就是说，维护的多项式实际上是其级数的前 $n+1$ 项 $\pmod{x^{n+1}}$。

OI 中，如果只关心其有限项的性质，一般不需要特别关注形式幂级数的敛散性问题。除非需要进行无穷级数的求和，代入点值等操作时，才需要考虑其收敛性。即使不收敛，仍然可以定义形式微分和形式积分等操作。

常见函数的 Taylor 展开

$$\frac{1}{1-x} = \sum_{i\geq 0} x^i$$

广义二项式定理:

$$(1+x)^{-(k+1)} = \sum_{i\geq 0} \binom{-k}{i} (-x)^i = \sum_{i\geq 0} \binom{k+i}{i} (-x)^i$$

$$e^x = \sum_{i\geq 0} \frac{x^i}{i!}$$

$$\ln(1-x) = -\sum_{i\geq 1} \frac{x^i}{i}$$

生成函数

生成函数是一个形式幂级数，用于描述某个数列的性质。对于一个数列 $\{a_n\}$，其生成函数定义为：

$$A(x) = \sum_{n \geq 0} a_n x^n$$

生成函数通过将多项式的系数与数列的项对应起来，可以方便地描述递推，笛卡尔积等操作。它是描述组合对象和计数问题的强大工具。

组合类

组合对象是问题研究的某些可数的对象，组合类是组合对象组成的集合。组合类内的每个元素都定义了一个大小，记元素 $\alpha \in \mathcal{A}$，则其大小为 $|\alpha|$，需要满足：

- $|\alpha| \in N$
- $\forall k \in N$，集合 $\mathcal{A}_k = \{\alpha \in \mathcal{A} || \alpha| = k\}$ 是有限的。

如果记 $\mathcal{A}$ 中大小为 n 的元素有 a_n 个，那它的生成函数为：

$$A(x) = \sum_{i=1}^{n} a_i x^i$$

组合类的不交并

对于组合类 $\mathcal{A}$ 和 $\mathcal{B}$，如果 $\mathcal{A} \cap \mathcal{B} = \emptyset$，则它们的不交并 $\mathcal{C} = \mathcal{A} \cup \mathcal{B}$ 的生成函数为：

$$C(x) = A(x) + B(x)$$

组合类的笛卡尔积

对于组合类 $\mathcal{A}$ 和 $\mathcal{B}$，它们的笛卡尔积 $\mathcal{C} = \mathcal{A} \times \mathcal{B}$ 定义为：

$$\mathcal{C} = \{(a, b) | a \in \mathcal{A}, b \in \mathcal{B}\}$$

其生成函数为：

$$C(x) = A(x) \cdot B(x)$$

Sequence 构造

对于组合类 $\mathcal{A}$，其 Sequence 构造定义为：

$$\mathrm{SEQ}(\mathcal{A}) = \{(a_1, a_2, \ldots, a_n) | a_i \in \mathcal{A}\} = \mathcal{E} + \mathcal{A} + \mathcal{A} \times \mathcal{A} + \mathcal{A} \times \mathcal{A} \times \mathcal{A} + \cdots$$

意为：取出 $\mathcal{A}$ 中的任意多个元素拼接在一起后得到的新组合类。

$$\mathrm{SEQ}(A(x)) = 1 + A(x) + A^2(x) + \cdots = \frac{1}{1 - A(x)}$$

指数型生成函数

对于组合类 $\mathcal{A}$，其指数型生成函数定义为：

$$\hat{A}(x) = \sum_{i\geq 0} a_i \frac{x^i}{i!}$$

请将 EGF 的卷积理解为**可重排**。即，对于两个组合类 $\mathcal{A}$ 和 $\mathcal{B}$，如果它们的乘积不是直接拼接，而是可以互相重排（但是来自同一个组合类的元素不区分），那么应该如此计算：

$$c_n = \sum_{i=0}^{n} \binom{n}{i} a_i b_{n-i} = n! \sum_{i=0}^{n} \frac{a_i}{i!} \frac{b_{n-i}}{(n-i)!}$$

$$\hat{C}(x) = \hat{A}(x) \cdot \hat{B}(x)$$

Exp 的组合意义（Set 构造）

对于组合类 $\mathcal{A}$，其 Set 构造定义为：

$$\mathrm{SET}(\mathcal{A}) = \{e | e \in a_1 \times a_2 \times \ldots \times a_n\}$$

意为：取出 $\mathcal{A}$ 中的任意多个元素，可重排在一起后得到的新组合类。

$$\mathrm{SET}(A(x)) = \sum_{k \geq 0} \frac{1}{k!} A(x)^k = \exp \hat{A}(x)$$

Exp 的组合意义（Set 构造）

Example

计数 n 个点的有标号连通图数量。

Exp 的组合意义（Set 构造）

Example

计数 n 个点的有标号连通图数量。

考虑任意一个图可以被唯一分解为若干个连通图的并，即可以理解为若干个连通图通过重标颜色组成的图。
设 a_n 为 n 个点的有标号连通图数量，b_n 为 n 个点的有标号图数量，则有：

$$\hat{B}(x) = \exp \hat{A}(x)$$

$$\hat{A}(x) = \ln \hat{B}(x)$$

[Zhejiang 2024 K] Sugar Sweet 3

三个人在玩摩尔投票，初始时分别持有 A, B, C 张不同类型的卡片。
游戏共进行 $A+B+C$ 轮。每轮**任意**一个玩家出牌，若牌堆为空或所有牌与当前牌类型相同，则将牌放入牌堆；否则，移除牌堆中的一张牌并丢弃当前牌。
设过程中（除了初始局面）时牌堆为空的时刻数量为 x，若最后牌堆为空，则获得 m^x 袋糖；否则没有糖。
求三个人所有可能的出牌顺序中，最终获得的糖袋数量的和，答案对 10^9+7 取模。

数据范围：$3s$，$1 \leq A, B, C \leq 1000$，$1 \leq A+B+C \leq 1000$，$1 \leq x \leq 10^9$.

[Zhejiang 2024 K] Sugar Sweet 3

三个人在玩摩尔投票，初始时分别持有 A, B, C 张不同类型的卡片。
游戏共进行 $A+B+C$ 轮。每轮**任意**一个玩家出牌，若牌堆为空或所有牌与当前牌类型相同，则将牌放入牌堆；否则，移除牌堆中的一张牌并丢弃当前牌。
设过程中（除了初始局面）时牌堆为空的时刻数量为 x，若最后牌堆为空，则获得 m^x 袋糖；否则没有糖。
求三个人所有可能的出牌顺序中，最终获得的糖袋数量的和，答案对 10^9+7 取模。

数据范围：$3s$，$1 \le A, B, C \le 1000$，$1 \le A+B+C \le 1000$，$1 \le x \le 10^9$.

- Hint 1：x 没有用

[Zhejiang 2024 K] Sugar Sweet 3

三个人在玩摩尔投票，初始时分别持有 A, B, C 张不同类型的卡片。
游戏共进行 $A+B+C$ 轮。每轮**任意**一个玩家出牌，若牌堆为空或所有牌与当前牌类型相同，则将牌放入牌堆；否则，移除牌堆中的一张牌并丢弃当前牌。
设过程中（除了初始局面）时牌堆为空的时刻数量为 x，若最后牌堆为空，则获得 m^x 袋糖；否则没有糖。
求三个人所有可能的出牌顺序中，最终获得的糖袋数量的和，答案对 10^9+7 取模。

数据范围：$3s$，$1 \le A, B, C \le 1000$，$1 \le A+B+C \le 1000$，$1 \le x \le 10^9$.

- Hint 1： x 没有用
- Hint 2： 分析结构

首先，这就是摩尔投票法，因此最后的局面只由 A, B, C 决定。最后牌堆里没有牌当且仅当 $2 \mid A+B+C$ 且 $2\max\{A, B, C\} \le A+B+C$。考虑按每次牌堆清空的节点将整个过程分成若干段。称每一段内第一次操作的颜色为该段的主色，那么该段内主色和非主色构成了一个括号序列。这就是整个问题的结构。

对于三种牌数量的限制 A, B, C，可以枚举每种颜色作为主色的数量 a, b, c，然后设每种主色对应的段中，非主色数量的分配具体为：

Color 1	a	$i_{(b)}$	$a - i_{(c)}$
Color 2	b	$j_{(c)}$	$b - j_{(a)}$
Color 3	c	$k_{(a)}$	$c - k_{(b)}$

$$\begin{cases} i - k = B - b - c \\ j - i = C - c - a \\ k - j = A - a - b \end{cases}$$

注意到由于 $a + b + c = A + B + C$，故只需要枚举 a, b；但是方程组不满秩，故还需要再枚举 i。枚举 (a, b, i) 后剩下的 (j, k) 可以直接求出。

接下来，枚举三种主色对应段的数量 p,q,r。那么以主色 1 为例，将非主色看成一种颜色，只需要求出将 $2a$ 个括号分为 p 段括号序列的方案数 $f(a,p)$，再乘上 $\binom{a}{i}$ 即可。最后再乘上 $\binom{p+q+r}{p,q,r}$ 后求和即得到答案。先考虑求出 $f(n,k)$ 的值，就是

$$f(n,k)=[x^n](\sum_{i\geq 1} Cat_i x^i)^k$$

可以直接 DP 求出；也可以在每个括号序列后加一个右括号，映射为 $(0,0)$ 走到 $(2n+k,-k)$ 的路径数：

$$f(n,k)=\binom{2n+k-1}{n}-\binom{2n+k-1}{n-1}$$

接下来，恰有 s 段的方案数为：

$$\begin{aligned} c_s &= \binom{a}{i}\binom{b}{j}\binom{c}{k}\sum_{p+q+r=s}\binom{s}{p,q,r}f(a,p)f(b,q)f(c,r) \\ &= \binom{a}{i}\binom{b}{j}\binom{c}{k}s!\sum_{p+q+r=s}\frac{f(a,p)}{p!}\frac{f(b,q)}{q!}\frac{f(c,r)}{r!} \\ &= \binom{a}{i}\binom{b}{j}\binom{c}{k}s![x^s]\Big(F_a(x)F_b(x)F_c(x)\Big) \end{aligned}$$

$$\text{where } F_a(x) = \sum_{p=0}^{a} f(a,p)\frac{x^p}{p!}$$

暴力多项式乘法十分爆炸，不过注意到我们实际上要求的是很多个这样的多项式乘积之和，故可以直接插值，这样就变成了点值乘法。

预先将所有 $F_a(x)$ 和点值处理出来，然后枚举 a, b, i 的值并累加点值，最后还原即可。总时间复杂度为 $O(n^3)$，常数小。

[CCPCFinal 2023 B] Periodic Sequence

对于正整数 n，设 $f(n)$ 为满足以下条件的字符串序列 $[S_1, S_2, \cdots, S_\ell]$ 的最大长度 ℓ：

- 每个字符串 S_i 为长度不超过 n 的非空小写字母字符串；
- 对于所有 $1 \le i < j \le \ell$，有 $S_i \ne S_j$；
- S_i 是 S_{i+1} 的周期。

求 $f(1), f(2), \cdots, f(N)$ 的值，答案对给定的质数 M 取模。

数据范围：$1 \le N \le 2 \times 10^5$，$5 \times 10^8 \le M \le 1.01 \times 10^9$，$M$ 是质数。

[CCPCFinal 2023 B] Periodic Sequence

对于正整数 n，设 $f(n)$ 为满足以下条件的字符串序列 $[S_1, S_2, \cdots, S_\ell]$ 的最大长度 ℓ：

- 每个字符串 S_i 为长度不超过 n 的非空小写字母字符串；
- 对于所有 $1 \le i < j \le \ell$，有 $S_i \neq S_j$；
- S_i 是 S_{i+1} 的周期。

求 $f(1), f(2), \cdots, f(N)$ 的值，答案对给定的质数 M 取模。

数据范围：$1 \le N \le 2 \times 10^5$，$5 \times 10^8 \le M \le 1.01 \times 10^9$，$M$ 是质数。

- Hint 1：上界是多少？

[CCPCFinal 2023 B] Periodic Sequence

对于正整数 n，设 $f(n)$ 为满足以下条件的字符串序列 $[S_1, S_2, \cdots, S_\ell]$ 的最大长度 ℓ：

- 每个字符串 S_i 为长度不超过 n 的非空小写字母字符串；
- 对于所有 $1 \le i < j \le \ell$，有 $S_i \ne S_j$；
- S_i 是 S_{i+1} 的周期。

求 $f(1), f(2), \cdots, f(N)$ 的值，答案对给定的质数 M 取模。

数据范围：$1 \le N \le 2 \times 10^5$，$5 \times 10^8 \le M \le 1.01 \times 10^9$，$M$ 是质数。

- Hint 1：上界是多少？
- Hint 2：根号

首先找上界。有一个显然的结论：对于任意一个可能出现的字符串 S_i，一定存在 $k \le |S_1|$ 使得 $S_i = S_1[1:k] + S_1[1:i_1] + \cdots + S_1[1:i_t]$，其中 $1 \le i_1, i_2, \cdots, i_t \le k$。容易归纳证明。
所以上界显然就是令 $S_1 = \text{'a'} + \text{'b'} \times (n-1)$ 时，所有这样的序列的个数，下面证明存在一个字符串序列能将它们全部串起来。

从 n 到 1 枚举 k。对于某个 k，把所有字符串连成一棵树，$S_i = S_1[1:k] + S_1[1:i_1] + \cdots + S_1[1:i_t]$ 即看作依次走边 $i_1, i_2, \cdots, i_t$ 到达的点。显然所有深度不超过 $l-k$ 的点与所有可能的字符串一一对应。那么假设现在字符串序列的末尾对应的节点是 u，添加一个新的串在树上对应：

1. 对于所有 u 的儿子 v，可以走到 v（对应新增一个前缀）。
2. 对于所有 u 的祖先 x，若 y 是 x 的某个更小的兄弟，则可以走到 x 或 y（对应删去一个后缀）。
3. 可能可以走到比儿子更深的节点，但这里不考虑。

按深度归纳证明如果初始时在根节点，那么可以遍历树上所有节点（并在某个节点停止）。假设根节点的儿子（按边权从小到大）依次为 $s_1, s_2, \cdots, s_k$，且 s_i 子树下按归纳假设构造的路径停止在 t_i 节点，那么直接按照以下方式构造：

$$\text{root} \to s_k \rightsquigarrow t_k \to s_{l-1} \rightsquigarrow t_{l-1} \to \cdots \to s_1 \rightsquigarrow t_1$$

最后，对于不同的 k，也可以类似上面的路径从大到小把根串起来。于是证明了存在一个字符串序列能将它们全部串起来，即上限能达到。

接下来考虑计数，枚举 k，则有：

$$
\begin{aligned}
f(l) &= \sum_{k=1}^{l} [x^l] \frac{1}{1-x} \cdot x^k \cdot \mathsf{SEQ}[x + x^2 + \cdots + x^k] \\
&= [x^l] \sum_{k=1}^{l} \frac{x^k}{1 - 2x + x^{k+1}}
\end{aligned}
$$

一种暴力做法是对于某个 k，转成线性递推，可以 $O(n)$ 算出所有系数。

另一种做法是暴力拆开：

$$\begin{aligned}
[x^l]\frac{x^k}{1-2x+x^{k+1}} &= [x^{l-k}]\sum_{t\geq 0} x^t(2-x^k)^t \\
&= \sum_{t\geq 0}[x^{l-k-t}](2-x^k)^t \\
(\text{let } t=l-kr,) &= \sum_{r\geq 0}[x^{kr-k}](2-x^k)^{l-kr} \\
&= \sum_{r\geq 0}\binom{l-kr}{r-1}(-1)^{r-1}2^{l-(k+1)r+1}
\end{aligned}$$

注意这里出现了 kr，于是考虑根号分治。对于 $k < B$，用前面提到的暴力做法；对于 $k \ge B$，一定有 $r \le \lfloor n/B \rfloor \triangleq B_0$，然后枚举 r：

$$
\begin{aligned}
f_0(l) &= \sum_{k=B}^{l} \sum_{r \ge 0} \binom{l-kr}{r-1} (-1)^{r-1} 2^{l-(k+1)r+1} \\
&= \sum_{r=0}^{B_0} (-1)^{r-1} 2^{l-r+1} \sum_{k=B}^{l} \binom{l-kr}{r-1} 2^{-kr}
\end{aligned}
$$

$$g_{r,l} \triangleq \sum_{k\geq B} \binom{l-kr}{r-1} 2^{-kr}$$

$$\begin{aligned}
g_{r,l+r} &= \sum_{k\geq B} \binom{l+r-kr}{r-1} 2^{-kr} \\
&= \sum_{k\geq B-1} \binom{l-kr}{r-1} 2^{-(k+1)r} \\
&= 2^{-r}\left[g_{r,l} + 2^{-r(B-1)}\binom{l-r(B-1)}{r-1}\right]
\end{aligned}$$

令 $B=O(\sqrt{n})$，时间复杂度 $O(n\sqrt{n})$。

Raney 引理

Theorem

若序列 $a_1, \cdots, a_n \in \mathbb{Z}$ 满足 $\sum a_i = 1$，那么其 n 个循环移位产生的序列中有且仅有一个满足所有前缀和为正。

Proof.

设前缀和为 s_k，设 x 为满足 $s_k = \min_t s_t$ 的最大的 k，则显然以 x 开头的循环移位序列合法。考虑以 $y(y \neq x)$ 开头的循环移位序列：

- 若 $y < x$，则 $s_y \geq s_x$，不合法。
- 若 $y > x$，则 $s_y > s_x$，即 $s_y \geq s_x + 1 = s_{x+n}$，不合法。

$\square$

Catalan 数

求有多少个长为 $2n$ 的序列 $a_1, \cdots, a_{2n} \in \{-1, 1\}$ 满足 $\sum a_i = 0$，使得所有前缀和非负。

考虑补上 $a_0 = 1$，限制就变成每个前缀和为正。由 Raney 引理就有：

$$C_n = \frac{1}{2n+1}\binom{2n+1}{n} = \frac{1}{n+1}\binom{2n}{n}$$

此外，考虑最后一个 $s_x = 1$ 且 $x < 2n$ 的 x，那么 $[0, x]$ 与 $[x+1, n-1]$ 均为合法序列，故卡特兰数的生成函数满足：

$$C(x) = xC^2(x) + 1$$

Fuss-Catalan 数

Fuss-Catalan 数

求有多少个长为 mn 的序列 $a_1, \cdots, a_{nm} \in \{1-m, 1\}$ 满足 $\sum a_i = 0$，使得所有前缀和非负。

可以算出共 n 个 $(1-m)$ 和 $mn-n$ 个 1。同样设 $a_0 = 1$，再用 Raney 引理得到：

$$C_n^{(m)} = \frac{1}{mn+1}\binom{mn+1}{n} = \frac{1}{mn-n+1}\binom{mn}{n}$$

设 b_t 为最大的 k 使得 $s_k = t$ 且 $k < mn$（认为 $b_0 = -1$），那么 $[b_t+1, b_{t+1}](0 \le t < m)$ 都是合法序列，这构成一个双射。于是其生成函数 $F(x)$ 满足：

$$F(x) = xF^m(x) + 1$$

Raney 引理推广

Theorem

若序列 $a_1, \cdots, a_n \in (-\infty, 1] \bigcap \mathbb{Z}$ 满足 $\sum a_i = d$，那么其 n 个循环移位产生的序列中有且仅有 d 个满足所有前缀和为正。

Proof.

设前缀和为 s_k。设 b_t 为最大的 k 使得 $s_k = t$（认为 $b_0 = 0$），那么所有 $[b_t + 1, b_{t+1}](0 \le t < d)$ 将整个序列分成 d 段。每一段里的位置最大的最小值均满足条件；其它位置同理易证不合法。 $\square$

广义二项级数

对上述 Fuss-Catalan 数的生成函数 $F(x)$，如何求 $[x^n]F^d(x)$？
构造双射：

Bijection

求有多少个长为 $mn+d-1$ 的序列 $a_1,\cdots,a_{nm+d-1}\in\{1-m,1\}$ 满足 $\sum a_k=d-1(d>0)$，使得所有前缀和非负。

还是令 $a_0=1$，设 b_t 为最大的 k 使得 $s_k=t$（认为 $b_0=-1$），那么 $[b_t+1,b_{t+1}](0\le t<d)$ 都是合法序列。那么就相当于 d 个合法序列拼在一起了。
代入 Raney 定理的推广，答案为：

$$[x^n]F^d(x)=\frac{d}{mn+d}\binom{mn+d}{n}$$

广义二项级数

该级数被称为广义二项级数 $\mathcal{B}_m(x)$：

$$\mathcal{B}_m(x) = \sum_{k\geq 0} \binom{mk+1}{k} \frac{x^k}{mk+1} = \sum_{k\geq 0} (mk)^{\underline{k-1}} \frac{x^k}{k!}$$

$$\mathcal{B}_m^d(x) = \sum_{k\geq 0} \binom{mk+d}{k} \frac{x^k}{mk+d}$$

[Ptz Summer 2024 Day 1 F] Fruit Tea

对于长为 $n+2$ 的非负整数序列 $a=\{a_0,a_1,a_2,\dots,a_n,a_{n+1}\}$，其中 $a_0=a_{n+1}=0$，定义：

$$f(a)=\sum_{i=0}^{n}\max(0,a_{i+1}-a_i)$$

求满足 $f(a)=k$ 的不同序列 $\{a\}$ 的数量，答案对 998244353 取模。

数据范围：$1\le n\le 2\times 10^5$，$0\le k\le 2\times 10^5$.

[Ptz Summer 2024 Day 1 F] Fruit Tea

对于长为 $n+2$ **的非负整数**序列 $a=\{a_0,a_1,a_2,\ldots,a_n,a_{n+1}\}$，**其中** $a_0=a_{n+1}=0$，**定义**：

$$f(a)=\sum_{i=0}^{n}\max(0,a_{i+1}-a_i)$$

求满足 $f(a)=k$ **的不同序列** $\{a\}$ **的数量，答案对** 998244353 **取模。**

数据范围：$1\le n\le 2\times 10^5$，$0\le k\le 2\times 10^5$.

- Hint 1：Raney 引理

把 a_i 的跳跃拆成很多步 $+1$ 或 -1，这样就变成长为 $2k$ 的 Dyck Path。
如果枚举这个 Dyck Path 恰有 t 个峰（$t-1$ 个谷），设此方案数为 $N_{k,t}$，如何计算稍后再说。
那么现在已经确定了 $s=2t-1$ 个断点（因为峰和谷必须作为断点），还需要再在 $[0,2k]$ 的任意位置放 $n-s$ 个断点。这等于解方程 $y_0+y_1+\cdots+y_{n+1-t}=2k, y_i\geq 0$，方案数为 $\binom{2k+n-t}{n-t}$。

$$Ans=\sum_{t=1}^{k}N_{k,t}\binom{2k+n-t}{n-t}$$

接下来考虑如何计算 $N_{n,k}$，即恰有 k 个峰的 $2n$ 步 Dyck Path 的方案数。其名叫那罗延数（Narayana number）。

在最前面补上 $a_0=1$，那么整个序列和为 1。我们统计所有有 $n+1$ 个 1，n 个 -1，且（首尾接起来）恰有 $2k$ 个连续段的序列数量。

于是，枚举 1 段前面的断点，只需要将 n 个 1 和 n 个 -1 分别分成 k 段即可，这个方案数是 $\binom{n-1}{k-1}^2$。把每个方案循环移位旋转，可以得到 n 个方案；又每个真实的方案会被算 k 次（因为有 k 段 1 的连续段），所以上述序列数有：

$$\frac{n}{k}\binom{n-1}{k-1}^2$$

$$N_{n,k}=\frac{1}{n}\frac{n}{k}\binom{n-1}{k-1}^2=\frac{1}{k}\binom{n-1}{k-1}^2=\frac{1}{n}\binom{n}{k}\binom{n}{k-1}$$

总时间复杂度 $O(n)$.

[3rdUCup Stage40 D] Fern Market

对于长为 n 的序列 $\{a\}$，定义 $f(a_1, \ldots, a_n)$ 为以下问题的答案：

- 你初始拥有足够多的金钱。第 i 天的股票价格为 a_i，你可以选择买入一份股票，卖出一份股票，或不进行任何操作。若要求第 n 天结束时你手上不能有股票，求你能获得的最大收益。

对于所有 $n!$ 个 1 到 n 的排列 p，定义 $g(n)$ 为所有 $f(p)$ 的和。
给定 k 和质数 m，求出 $g(1), g(2), \cdots, g(k)$ 分别对 m 取模后的结果。

数据范围：$1 \leq k \leq 7000$，$10^8 + 7 \leq m \leq 10^9 + 7$，$m$ 是质数。

[3rdUCup Stage40 D] Fern Market

对于长为 n 的序列 $\{a\}$，定义 $f(a_1, \ldots, a_n)$ 为以下问题的答案：

- 你初始拥有足够多的金钱。第 i 天的股票价格为 a_i，你可以选择买入一份股票，卖出一份股票，或不进行任何操作。若要求第 n 天结束时你手上不能有股票，求你能获得的最大收益。

对于所有 $n!$ 个 1 到 n 的排列 p，定义 $g(n)$ 为所有 $f(p)$ 的和。
给定 k 和质数 m，求出 $g(1), g(2), \cdots, g(k)$ 分别对 m 取模后的结果。

数据范围：$1 \le k \le 7000$，$10^8 + 7 \le m \le 10^9 + 7$，$m$ 是质数。

- Hint 1：反悔贪心

[3rdUCup Stage40 D] Fern Market

对于长为 n 的序列 $\{a\}$，定义 $f(a_1, \ldots, a_n)$ 为以下问题的答案：

- 你初始拥有足够多的金钱。第 i 天的股票价格为 a_i，你可以选择买入一份股票，卖出一份股票，或不进行任何操作。若要求第 n 天结束时你手上不能有股票，求你能获得的最大收益。

对于所有 $n!$ 个 1 到 n 的排列 p，定义 $g(n)$ 为所有 $f(p)$ 的和。
给定 k 和质数 m，求出 $g(1), g(2), \cdots, g(k)$ 分别对 m 取模后的结果。

数据范围：$1 \leq k \leq 7000$，$10^8 + 7 \leq m \leq 10^9 + 7$，$m$ 是质数。

- Hint 1：反悔贪心
- Hint 2：扫描值域，转化成 01 序列

首先如果不考虑计数，这就是那个广为人知的经典反悔贪心题 CF865D Buy Low Sell High。做法是：

- 维护一个小根堆。从前往后考虑每一天，设价格为 p_i：
  - 如果堆内没有比 p_i 小的元素，则将 p_i 入堆。
  - 否则将堆顶元素 x 出堆，答案加上 $p_i - x$，然后将 p_i 入堆两次。

考虑枚举 t，然后统计 $[t,t+1)$ 被算进的贡献。将 $\le t$ 的元素标为 0，将 $>t$ 的元素标为 1，则共有 t 个 0 和 $n-t$ 个 1。会发现按照上述策略等价于对这个新的序列进行反悔贪心：

- 如果 $p_i=0$，那么无论什么情况最后都会往堆里放入一个 0。
- 如果 $p_i=1$，且栈内有 0，则出栈一个 0 并入栈两个 1；
- 如果 $p_i=1$，且栈内没有 0，则无论什么情况最后都会往堆里放入一个 1。

也就是说，一定能够取到最大值，即：

$$\max_{t=1}^{n-1}\{\min\{\#0_{\le t},\#1_{>t}\}\}$$

将 0 看为向右上走一步，将 1 看为向右下走一步，但已经在 0 时遇到 1 只能水平向右走一步，并将这种情况记作“损失”。那么对于一个序列，其最后的贡献就是 1 的数量减去“损失”发生的次数。

所有 1 的数量很好算，就是 $(n-t)\binom{n}{t}$。
对于“损失”，有结论：如果不管在 0 时遇到 1 只能水平向右走一步的限制，直接按照普通 Dyck Path 的规则，则“损失”数量等于最低点高度的倒数。
于是枚举 $r \le -1$，统计最低点至少低于 r 的 Dyck Path 数量，最后求和即可。
最后推出来的答案是一个上指标为 n 的组合数前缀和。故枚举 n，先预处理组合数前缀和，然后再枚举 t 统计答案即可。时间复杂度 $O(k^2)$.

容斥原理

容斥原理本质上是不同赋权方式的线性组合。
设计数对象集合为 S，有函数 $g: S \to \mathbb{R}$，我们希望计算：

$$g(S) = \sum_{s \in S} g(s)$$

如果能找到 n 个函数 $f_k: S \to \mathbb{R}$，使得对于任意 $s \in S$，都有：

$$g(s) = \sum_{k=1}^{n} c_k f_k(s)$$

则可以通过计算 $f_k(S)$ 来得到 $g(S)$：

$$g(S) = \sum_{k=1}^{n} c_k f_k(S)$$

例：基本容斥原理

Problem

设 $S_1, S_2, \ldots, S_n \subseteq S$ 为 n 个集合，求 $|\bigcup_{k=1}^{n} S_k|$ 的值。

$$g(s) = \begin{cases} 1, & \text{if } s \in \bigcup_{k=1}^{n} S_k \\ 0, & \text{otherwise} \end{cases}$$

对于 $T \subseteq [n], T \neq \varnothing$，将 $f_T(s)$ 设为：

$$f_T(s) = \begin{cases} 1, & \text{if } s \in \bigcap_{k \in T} S_k \\ 0, & \text{otherwise} \end{cases}$$

$$c_T = (-1)^{|T|}$$

例：基本容斥原理

这样，对于任意 $s \in S$，设 $T_s = \{k \mid s \in S_k\}$，则有：

$$\sum_{T\subseteq[n]} c_T f_T(s) = \sum_{T\subseteq T_s} c_T$$
$$= \sum_{T\subseteq T_s, T\neq\varnothing} (-1)^{|T|} = \sum_{k=1}^{|T_s|}(-1)^k\binom{|T_s|}{k}$$
$$= [|T| \geq 1] = g(s)$$

因此有：

$$|\bigcup_{k=1}^{n} S_k| = \sum_{T\subseteq[n]} c_T f_T(S) = \sum_{k=1}^{n}(-1)^{k-1}\binom{n}{k}|S_k|$$

例：二项式反演

Problem

设 $S_1, S_2, \ldots, S_n \subseteq S$ 为 n 个集合，设 $T_s = \{t \mid s \in S_t\}$，求满足 $|T_s| = k$ 的 $s \in S$ 的数量。

$$g(s) = \begin{cases} 1, & \text{if } |T_s| = k \\ 0, & \text{otherwise} \end{cases}$$

对于 $i \in [0, n]$，将 $f_i(s)$ 设为：

$$f_i(s) = \sum_{T \subseteq [n], |T| = i} [s \in \bigcap_{t \in T} S_t]$$

例：二项式反演

这样，对于任意 $s \in S$，设 $|T_s| = t$，则有：

$$g(s) = \sum_{i=t}^{n} c_i f_i(s) = \sum_{i=t}^{n} c_i \binom{i}{t} = [t = k]$$

即，系数 c_i 需要满足：

$$\sum_{i=t}^{n} c_i \binom{i}{t} = [t = k], \forall t \in [0, n]$$

二项式反演得到：

$$c_t = \sum_{i=t}^{n} (-1)^{i-t} \binom{i}{t} [i = k] = (-1)^{k-t} \binom{k}{t}$$

例：二项式反演

最后的结果即：

$$g_k(S) = \sum_{i=0}^{n} (-1)^{k-i} \binom{k}{i} f_i(S)$$

其中 $g_k(S)$ 表示恰好满足 k 个条件的方案数，而 $f_i(S)$ 表示钦定满足 i 个条件的方案数。

例：Mobius 反演

Problem

设正整数可重集 S，若知道 S 中 k 的倍数的元素的数量，求 S 中等于 n 的元素的数量。

$$g_n(s) = [s = n], f_k(s) = [k \mid s]$$

$$g_n(s) = \sum_{k \geq 1} c_s f_k(s) = \sum_{k \geq 1}^{n} c_s [k \mid s] = [s = n]$$

例：Mobius 反演

其中 c_k 满足：

$$\sum_{k|s} c_k = [s = n]$$

由 Mobius 反演得到：

$$c_s = \sum_{k|s} \mu(s/k)[k = n] = [n \mid s]\mu(s/n)$$

$$g_n(S) = \sum_{n|s} \mu(s/n) f_s(S)$$

[2ndUCup Stage21 E] Enumerating Substrings

给定一个大小为 k 的字母表，对于字符串 S（文本串）和字符串 P（模式串），定义 $F(S,P)$ 为 S 中可以取出的**不重叠**且等于 P 的子串的最大数量。
如果一个字符串 Q 中每个字母出现的次数不超过 2 次，则称 Q 是“美丽的”。
对于所有长度为 n 的字符串 S 和所有长度为 m 的“美丽”模式串 P，计算 $F(S,P)$ 的总和，答案对 10^9+7 取模。

数据范围：$1 \le n \le 10^6$，$1 \le m \le 2000$，$m \le n$，$1 \le k \le 10^9$.

[2ndUCup Stage21 E] Enumerating Substrings

给定一个大小为 k 的字母表，对于字符串 S（文本串）和字符串 P（模式串），定义 $F(S,P)$ 为 S 中可以取出的**不重叠**且等于 P 的子串的最大数量。
如果一个字符串 Q 中每个字母出现的次数不超过 2 次，则称 Q 是"美丽的"。
对于所有长度为 n 的字符串 S 和所有长度为 m 的"美丽"模式串 P，计算 $F(S,P)$ 的总和，答案对 10^9+7 取模。

数据范围：$1 \le n \le 10^6$，$1 \le m \le 2000$，$m \le n$，$1 \le k \le 10^9$.

- Hint 1：Q 的特殊性质有什么用？

[2ndUCup Stage21 E] Enumerating Substrings

给定一个大小为 k 的字母表，对于字符串 S（文本串）和字符串 P（模式串），定义 $F(S,P)$ 为 S 中可以取出的**不重叠**且等于 P 的子串的最大数量。
如果一个字符串 Q 中每个字母出现的次数不超过 2 次，则称 Q 是“美丽的”。
对于所有长度为 n 的字符串 S 和所有长度为 m 的“美丽”模式串 P，计算 $F(S,P)$ 的总和，答案对 10^9+7 取模。

数据范围：$1 \le n \le 10^6$，$1 \le m \le 2000$，$m \le n$，$1 \le k \le 10^9$.

- Hint 1：Q 的特殊性质有什么用？
- Hint 2：分析结构

P 中所有字符都至多出现两次，这意味着：

- P 不能有长超过一半的 Border，否则会有长小于一半的周期，必然会有字符出现超过两次；
- P 除了自身之外至多只有一个 Border，否则第一个字符会出现超过两次。
- 如果 P 有 Border，那么 Border 以外的部分和 Border 没有共用字符。

所以如果 P 出现的位置有重叠，它们只能以唯一的方式串起来，并且相邻两个 P 重叠的部分就是 Border。

我们把 k 个串起来的 P 看成一个新的计数对象 P^k。显然原问题可以拆解成每个位置上 P^k 出现多少次，想对每个 k 求出恰好 k 个 P 串起来并摆在任意位置的方案数 g_k。
考虑容斥，设 f_k 表示钦定某个位置有 k 个 P 串起来的方案数。则：

$$f_k = \sum_{i \geq k} (i - k + 1) g_i = g_k + \sum_{i > k} g_i$$

$$g_k = f_k - \sum_{i > k} g_i$$

接下来考虑如何计算 f_k。记 $S(n,m)$ 表示将 m 种字符放入 n 个位置，且每个字符出现次数均不超过 2 的方案数。

首先 $f_1 = S(m,k)(n-m+1)k^{n-m}$；对于 $k>1$，枚举 Border 长度 t，则：

$$g_k = \sum_{t=0}^{\lfloor m/2 \rfloor} k^t S(m-2t, k-t) \cdot (n-len+1)k^{n-len}, \text{ where } len = km-(k-1)t$$

对于 $S(n,m)$，枚举有 i 种字符出现 2 次，$n-2i$ 种字符出现 1 次，则：

$$S(n,m) = \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{m}{i}\binom{m-i}{n-2i} \cdot \frac{n!}{2^i}$$

注意 k 很大，所以最好先预处理出所有 $\binom{k-i}{j}$ 的值。时间复杂度 $O(n\log n + m^2)$.

[CCPC2023 Shenzhen B] Excuse

重复 n 次以下过程以生成序列 $a_1, a_2, \ldots, a_n$：

- 不断抛一个公平硬币，直到出现第一次反面。设此前正面出现的次数为 k，则将 k 添加到序列末尾。

求序列 $a_1, a_2, \ldots, a_n$ 的 mex 的期望值，答案对 998244353 取模。

数据范围：$1 \leq n \leq 10^5$。

[CCPC2023 Shenzhen B] Excuse

重复 n 次以下过程以生成序列 $a_1, a_2, \ldots, a_n$：

- 不断抛一个公平硬币，直到出现第一次反面。设此前正面出现的次数为 k，则将 k 添加到序列末尾。

求序列 $a_1, a_2, \ldots, a_n$ 的 mex 的期望值，答案对 998244353 取模。

数据范围：$1 \le n \le 10^5$。

- Hint 1：不要直接容斥

[CCPC2023 Shenzhen B] Excuse

重复 n 次以下过程以生成序列 $a_1, a_2, \ldots, a_n$：

- 不断抛一个公平硬币，直到出现第一次反面。设此前正面出现的次数为 k，则将 k 添加到序列末尾。

求序列 $a_1, a_2, \ldots, a_n$ 的 mex 的期望值，答案对 998244353 取模。

数据范围：$1 \le n \le 10^5$。

- Hint 1：不要直接容斥
- Hint 2：无限乘积

每个 a_i 都服从几何分布，并且 $P[a_i = k] = 2^{-k-1}$.
首先拆期望 $E[X] = \sum_{k\geq 1} P[X \geq k]$，而 mex $\geq k$ 就是 $0, 1, \cdots, k-1$ 都出现过。记 $X \geq k$ 的概率为 s_k.
“都出现过”的限制很容易让人想到容斥，但是尝试之后根本没法做。我们尝试用更根本和暴力的办法：直接写成指数型生成函数。

$$\hat{F}_t(x) = \sum_{i\geq 1} \frac{1}{i!}(x/2^{t+1})^i = e^{x/2^{t+1}} - 1, \text{ for } x = 0, 1, \cdots, k-1$$

$$\hat{F}_k(x) = \sum_{i\geq 0} \frac{1}{i!}(x/2^{k+1})^i = e^{x/2^{k+1}}$$

$$s_k = n![x^n] \prod_{t=0}^{k-1} \hat{F}_t(x) \cdot \hat{F}_k(x) = n![x^n] \prod_{t=0}^{k-1} (e^{x/2^{t+1}} - 1) \cdot e^{x/2^{k+1}}$$

注意到连乘号里面对于不同的 k 相当于求前缀积，这引发一个大胆的想法：尝试直接求出这个无限乘积，尽管现在我们无法确信它一定收敛。但是注意到里面每一项都没有常数项，所以至少先对每一项提出 x，得到：

$$s_k = n![x^n]\frac{x^k}{2^{k(k+1)/2}}e^{x/2^{k+1}}\prod_{t=0}^{k-1}(e^{x/2^{t+1}}-1) = n![x^n]\frac{x^k}{2^{k(k+1)/2}}e^{x/2^{k+1}}\frac{G(x)}{G(x/2^k)}$$

$$G(x) = \prod_{t=0}^{+\infty}(e^{x/2^{t+1}} - 1)$$

直接取对数。先求出 $H(x)=\ln\frac{e^x-1}{x}$ 的泰勒展开。展开形式和伯努利数有些联系，不过这里可以直接求多项式的 $\ln$。接下来：

$$\ln G(x)=\sum_{t=0}^{+\infty}\sum_{t\ge 1}H\left(\frac{x}{2^t}\right)=\sum_{i\ge 1}h_i\sum_{t\ge 1}\left(\frac{x}{2^t}\right)^i=\sum_{i\ge 1}\frac{2^{-i}}{1-2^{-i}}h_ix^i$$

所以 $G(x)$ 确实是收敛的，直接对上式求多项式 $\exp$ 即可得到 $G(x)$ 的多项式展开。

最后，计算答案：

$$Ans_n=\sum_{k\geq 1}s_k=n![x^n]G(x)\sum_{k\geq 1}\frac{x^k}{2^{k(k+1)/2}}\frac{e^{x/2^{k+1}}}{G(x/2^k)}$$

设 $P(x)=e^x/G(x)$，$Q(x)=\sum_{k\geq 1}\frac{x^k}{2^{k(k+1)/2}}P(x/2^k)$，则：

$$P(x/2^k)=\sum_{i\geq 0}p_i2^{-ki}x^i$$

$$q_t=\sum_{k=1}^{t}2^{-k(k+1)/2}p_{t-k}2^{-k(t-k)}$$

设 $d=t-k$，用 Bluestein Method：

$$2^{-k(k+1)/2}p_d2^{-kd}=(2^{-1})^{\binom{k+d}{2}}\left[\frac{1}{2^{k(k+1)/2}}2^{\binom{k}{2}}\right]\left[p_d2^{\binom{d}{2}}\right]$$

于是直接一次卷积即可获得所有 Ans_n 的值。时间复杂度 $O(n\log n)$.

Bonus

D-Finite 级数

称多项式 $F(x)$ 为 D-finite 形式幂级数当且仅当存在非负整数 r 与多项式 $a_0(x), a_1(x), \cdots, a_r(x)$ 满足：

$$\sum_{i=0}^{r} a_i(x) F^{(i)}(x) = 0$$

其中 r 为阶。

D-Finite 级数：常见例子

- 代数形式幂级数（包括有限次多项式，有限次分式）
- 指数函数 e^x，三角函数 $\sin x, \cos x$，对数函数 $\ln(1+x)\cdots\cdots$
- 一行组合数：$\sum_{i=0}^{n}\binom{n}{i} = (1+x)^n$
- 一列组合数：$\sum_{i}\binom{i}{n} = \frac{1}{(1-x)^{n+1}}$
- 斐波那契数 / 卡特兰数：带根号的封闭形式（二次方程根）
- 阶乘的生成函数，调和级数，错排
- 二维分式的一行：是 Algebraic 的

D-Finite 级数

若多项式 $F(x)$，$G(x)$ 均为 D-finite 的，

- 可加：$F(x)$ 与 $G(x)$ 的线性组合 $\alpha F(x) + \beta G(x)$ 也为 D-finite 的。
- 可乘：$F(x)$ 与 $G(x)$ 的卷积 $F(x)G(x)$ 与点积 $\sum_n f_n g_n x^n$ 也为 D-finite 的。此外 $F^n(x)$ 也为 D-finite 的。
- 可积分：$F(x)$ 的积分 $\int F(x)$ 也为 D-finite 的。
- 可复合（代数函数）：若 $a(x)$ 为 Algebraic 的，则 $F(a(x))$ 也为 D-finite 的（但 $a(F(x))$ 不一定）。

D-Finite 级数

对于阶为 r 的 D-finite 形式幂级数 $F(x)$，记录连续的 r 个系数可以 $O(r)$ 推出下一项系数。事实上 D-finite 形式幂级数具有整式递推的形式。

同时，若一列 D-finite 形式幂级数的相邻两项的比值为简单分式，可以由一项的系数快速推到下一项的系数，可以以曼哈顿距离的复杂度从一个位置的值转移到另一个位置（例如莫队形式的组合数前缀和）。

Q-Analog

q-analog 是对一般对象组合意义的拓展，一般令 $q=1$ 即是原对象。q 可以是任意对象，但下面仅认为 q 是 $\mathbb{F}_p$ 下的整数且满足 $\mathrm{ord}_q < n$。

Definition (q-整数)

q-整数 $[n]_q$ 定义为：

$$[n]_q = \frac{1-q^n}{1-q} = 1 + q + \cdots + q^{n-1}$$

当 $q=1$ 时对左式取极限得 $[n]_1 = n$，即一般正整数。

Definition (q-阶乘和 q-组合数)

定义 q-阶乘及 q-组合数为:

$$n!_q = \prod_{i=1}^{n} [i]_q$$

$$\binom{n}{m}_q = \frac{n!_q}{m!_q (n-m)!_q}$$

q-binomial 有许多与一般组合数类似的性质，如 q-binomial 均为非负整数，二项式定理，范德蒙德卷积，卢卡斯定理等。这里暂不详细展开。

Q-Analog 与多项式

考虑 $P(x)=\prod_{i=0}^{n}\frac{1}{1-q^ix}$。

$$P(qx)=\prod_{i=1}^{n}\frac{1}{1-q^ix}=\frac{1-x}{1-q^{n+1}x}P(x)$$

$$(1-q^{n+1}x)P(qx)=(1-x)P(x)$$

$$p_i(q^i-1)=p_{i-1}(q^{n+i}-1)$$

这个形式类似 $i\binom{n+i}{n}=(i+n)\binom{n+i-1}{n}$，有：

$$[x^i]P(x)=\binom{n+i}{n}_q$$

关于正负号，$P(x)$ 中的系数一定全为正的，而 $n!_q$ 也为正。

Q-Analog 与多项式

类似的，若设 $P(x)=\prod_{i=0}^{n}(1+q^ix)$，可以得到：

$$(1+q^{n+1}x)P(qx)=(1+x)P(x)$$

$$p_i(q^i-1)=p_{i-1}(q^{n+1}-q^{i-1})=p_{i-1}q^{i-1}(q^{n-i+2}-1)$$

同样类比 $i\binom{n+1}{i}=(n-i+2)\binom{n+1}{i-1}$，可得到：

$$[x^i]P(x)=q^{i(i-1)/2}\binom{n+1}{i}_q$$

[Nanchang 2025 H] Bingo Game

给定一个 $n \times n$ 的网格，选出尽可能多的格子，使得不存在任何一行或一列或正/反对角线上的格子均被勾选。
求在选出格子尽可能多的前提下，有多少种不同的勾选方案。答案对 998244353 取模。

数据范围：$2 \leq n \leq 10^5$，$1 \leq T \leq 10$。

当 $n>2$ 时，最多能选出 n^2-n 个格子，直接选 $(1,2),(2,3),\ldots,(n-1,n),(n,1)$ 即可。
设第 i 行不选的格子为 (i,p_i)，那么 $\{p_i\}$ 一定构成一个排列，只需要对这个排列计数。现在限制就变成了：不能全都 $p_i\neq i$（限制 A），也不能全都 $p_i\neq n+1-i$（限制 B）。
直接容斥，记 c_S 表示钦定不满足 S 内限制的方案数。$p_\varnothing=n!$，$p_A=p_B=D_n$ 即错排数。

对于 p_{AB}，即要求正反对角线都不能选。再次容斥，钦定正反对角线上一些格子必须选。由于中心格不一定存在，需要对 n 的奇偶性分类讨论，先考虑 n 为偶数的情况。

那么此时 $(k,k),(k,n+1-k),(n+1-k,k),(n+1-k,n+1-k)$ 四个格子会被分为一组，，考虑有多少种钦定方案会实际确定 t 个格子：

- $t=0$ 时，即四个都不钦定，1 种；
- $t=1$ 时，即任选一个钦定，4 种；
- $t=2$ 时，只能选对角的两个钦定，2 种。

所以写成生成函数就是：

$$F(x)=(1+4x+2x^2)^{n/2}$$

$$p_{AB}=\sum_{i=0}^{n}(n-i)!f_i$$

只需要求出 $F(x)$ 即可。设 $m=n/2$，注意到它是 d-finite 的，直接求导：

$$F'(x)=m(4x+4)(1+4x+2x^2)^{m-1}=m(4x+4)\frac{F(x)}{1+4x+2x^2}$$

$$(1+4x+2x^2)\sum_{k\geq 1}kf_kx^{k-1}=4m(x+1)\sum_{k\geq 0}f_kx^k$$

$$(k+1)f_{k+1}+4kf_k+2(k-1)f_{k-1}=4m(f_{k-1}+f_k)$$

$$(k+1-4m)f_{k+1}=4(m-k)f_k-2(k-1)f_{k-1}$$

对于 n 为奇数的情况，只需要将 $F(x)$ 再乘上 $1+x$ 即可。
总时间复杂度 $O(Tn)$.

[3rdUCup Stage39 F*] Fourier Coefficients

这是一道交互题。
给定一个整数 n，你需要猜一个函数 $f(x)$，满足以下形式：

$$f(x) = \sum_{k=0}^{n-1} A_k \cos(kx)$$

，其中 $A_0, A_1, \ldots, A_{n-1}$ 是整数且 $0 \le A_k < 998244353$。
你仅能进行一次交互：给出 n 对整数 $(X_1, Y_1), \ldots, (X_n, Y_n)$，满足 $0 \le X_i \le Y_i < 998244353$ 且 $Y_i \ne 0$。
交互库会返回 n 个整数 $Z_1, \ldots, Z_n$，其中：

$$Z_i = f(\arccos(X_i/Y_i)) \mod 998244353$$

数据范围：$8s$（保证交互库用时不超过 $1.3s$），$1 \le n \le 5 \times 10^5$.

[3rdUCup Stage39 F*] Fourier Coefficients

这是一道交互题。
给定一个整数 n，你需要猜一个函数 $f(x)$，满足以下形式：

$$f(x) = \sum_{k=0}^{n-1} A_k \cos(kx)$$

，其中 $A_0, A_1, \ldots, A_{n-1}$ 是整数且 $0 \le A_k < 998244353$。
你仅能进行一次交互：给出 n 对整数 $(X_1, Y_1), \ldots, (X_n, Y_n)$，满足 $0 \le X_i \le Y_i < 998244353$ 且 $Y_i \ne 0$。
交互库会返回 n 个整数 $Z_1, \ldots, Z_n$，其中：

$$Z_i = f(\arccos(X_i/Y_i)) \mod 998244353$$

数据范围：$8s$（保证交互库用时不超过 $1.3s$），$1 \le n \le 5 \times 10^5$.

- Hint 1：为什么是有理数？

[3rdUCup Stage39 F*] Fourier Coefficients

这是一道交互题。
给定一个整数 n，你需要猜一个函数 $f(x)$，满足以下形式：

$$f(x) = \sum_{k=0}^{n-1} A_k \cos(kx)$$

，其中 $A_0, A_1, \dots, A_{n-1}$ 是整数且 $0 \le A_k < 998244353$。
你仅能进行一次交互：给出 n 对整数 $(X_1, Y_1), \dots, (X_n, Y_n)$，满足 $0 \le X_i \le Y_i < 998244353$ 且 $Y_i \ne 0$。
交互库会返回 n 个整数 $Z_1, \dots, Z_n$，其中:

$$Z_i = f(\arccos(X_i / Y_i)) \mod 998244353$$

数据范围：$8s$（保证交互库用时不超过 $1.3s$），$1 \le n \le 5 \times 10^5$.

- Hint 1：为什么是有理数？
- Hint 2：不要代复数点值

首先注意到 $X_i/Y_i(X_i \le Y_i)$ 没有用，记 $x \equiv X_i/Y_i \pmod p$，可以构造 $X_i = 1, Y_i = x^{-1}$ 询问即可。
$\cos(nx) = T_n(\cos x)$ 是第一类 Chebyshev 多项式，故 $f(\arccos x)$ 确实是关于 x 的（至多）n 次有理多项式。但由于 Chebyshev 多项式很难描述，直接去做并不好。

考虑用复数形式去描述。记 $\theta = \arccos x$，$z = e^{i\theta} \in \mathbb{C}$，那么：

$$x = \frac{1}{2}(e^{i\theta} + e^{-i\theta}) = \frac{1}{2}(z + z^{-1})$$

$$\cos(k \arccos x) = \frac{1}{2}(e^{ki\theta} + e^{-ki\theta}) = \frac{1}{2}(z^k + z^{-k})$$

$$f(z) = \frac{1}{2}\sum_{k=0}^{n-1} A_k(z^k + z^{-k})$$

注意到这里 $\cos(k \arccos x)$ 是关于 x 的多项式（也就是上面所说的第一类 Chebyshev 多项式），**并且这个多项式成立只用到了 $z \cdot z^{-1} = 1$ 的性质**，故我们可以解除 $|z| = 1$ 的限制，并将定义域收缩到 $\mathbb{Q}$ 上。这意味着接下来我们可以将 z 代入为任意非零有理数，然后获取点值。

现在问题就变成了：可以指定 n 个点 $z_0, z_1, \cdots, z_{n-1}$，需要插值还原出 $f(z)$。由于 $f(z) = f(z^{-1})$，可以询问 $z_i = q^i$，其中 $q^i \neq 1 \pmod p$ for $i \geq 1$（可以直接令 q 为 p 的原根）。这样就可以获得 $q^{-n+1}, \cdots, q_{n-1}$ 共 $2n-1$ 个点值。
再令 $g(z) = z^{n-1} f(z)$ 为一个 $2n-1$ 次多项式，接下来就只需要对其插值即可。
对等比数列插值可以做到 $O(n \log n)$，直接推：

$$
\begin{aligned}
F(x) &= \sum_{i=0}^{n} y_i \prod_{j\neq i} \frac{x - x_j}{x_i - x_j} \\
&= \sum_{i=0}^{n} w_i \prod_{j\neq i} (x - x_j) \\
&= \prod_{i=0}^{n} (x - q^i) \cdot \sum_{i=0}^{n} \frac{w_i}{x - q^i} \\
&= \prod_{i=0}^{n} (x - q^i) \cdot \sum_{i=0}^{n} -q^{-i} \frac{w_i}{1 - q^{-i} x} \\
&= \prod_{i=0}^{n} (x - q^i) \cdot \sum_{j\geq 0} -x^j \sum_{i=0}^{n} w_i q^{-i(j+1)} \\
&= \prod_{i=0}^{n} (x - q^i) \cdot \sum_{j\geq 0} -x^j P(q^{-j})
\end{aligned}
$$

其中 $P(q^{-j})$ 的点值可以通过 Chirp Z 变换在 $O(n\log n)$ 内求出。

$$\begin{aligned}
\prod_{j\neq i}(q^i-q^j) &= \prod_{j=0}^{i-1}(q^i-q^j)\prod_{j=i+1}^{n}(q^i-q^j)\\
&= q^{\binom{i}{2}}\prod_{j=0}^{i-1}(q^{i-j}-1)\cdot q^{i(n-i)}\prod_{j=i+1}^{n}(1-q^{j-i})\\
&= q^{\binom{i}{2}+i(n-i)}(-1)^{n-i}s_is_{n-i}
\end{aligned}$$

$$\begin{aligned}
S_n(x) &= \prod_{i=0}^{n}(x-q^i)\\
&= (-1)^{n+1}q^{\binom{n+1}{2}}\prod_{i=0}^{n}(1-q^{-i}x)\\
&= (-1)^{n+1}q^{\binom{n+1}{2}}\sum_{i=0}^{n}(-1)^i q^{-\binom{i}{2}}\binom{n+1}{i}_q x^i
\end{aligned}$$

总时间复杂度 $O(n\log n)$.

[Ptz2024 Day3 B*] Festival Decorating

给定 n 个灯，第 i 个灯的坐标为 x_i，颜色为 c_i。
有 q 个查询，对于每个查询 d_i，找到最小的 u 使得存在另一个灯位于 $x_u + d_i$ 且颜色与 c_u 不同；如果不存在则输出 0。
只要求答案与正确答案的绝对或相对误差不超过 0.5.

数据范围：$1 \leq n, q \leq 2.5 \times 10^5$，$1 \leq x_i \leq 2.5 \times 10^5$，$1 \leq c_i \leq n$，$1 \leq d_i \leq 2.5 \times 10^5$。

[Ptz2024 Day3 B*] Festival Decorating

给定 n 个灯，第 i 个灯的坐标为 x_i，颜色为 c_i。
有 q 个查询，对于每个查询 d_i，找到最小的 u 使得存在另一个灯位于 $x_u + d_i$ 且颜色与 c_u 不同；如果不存在则输出 0。
只要求答案与正确答案的绝对或相对误差不超过 0.5.

数据范围：$1 \le n, q \le 2.5 \times 10^5$，$1 \le x_i \le 2.5 \times 10^5$，$1 \le c_i \le n$，$1 \le d_i \le 2.5 \times 10^5$。

- Hint 1：相对误差不超过 0.5

References

- command_block. 多项式计数杂谈. https://www.luogu.com.cn/article/oy8l7j3n
- Appleblue17. Re: 从零开始的生成函数. https://www.cnblogs.com/Appleblue17/p/14337965.html
- Appleblue17. 《具体数学》第五章二项式系数学习笔记（部分）. https://www.cnblogs.com/Appleblue17/p/16653961.html
- Wikipedia. Holonomic function. https://en.wikipedia.org/wiki/Holonomic_function
- OIWiki. 多项式与生成函数简介. https://oiwiki.33dai.wiki/math/poly/intro/