生生函数的运算与组合计数问题

April 2, 2020

生成函数是啥

- 想必大家已经很熟练了
- 对于数列 $\{a_i\}$，我们定义它的普通生成函数 (OGF) 和指数生成函数 (EGF) 分别为

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

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

生成函数是啥

- 想必大家已经很熟练了
- 对于数列 $\{a_i\}$，我们定义它的普通生成函数 (OGF) 和指数生成函数 (EGF) 分别为

$$A(x)=\sum_{i\geq 0}a_ix^i$$

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

- 普通生成函数解决无标号问题，指数生成函数解决有标号问题

生成函数是啥

- 两个普通生成函数的乘积 $A(x) \times B(x)$ 可以理解为，从 A 中选择一个元素，再从 B 中选取一个元素，然后再将它们拼接起来

生成函数是啥

- 两个普通生成函数的乘积 $A(x) \times B(x)$ 可以理解为，从 A 中选择一个元素，再从 B 中选取一个元素，然后再将它们拼接起来
- 比如说，有两个不相同的骰子，问有多少种方式使得两个骰子朝上的点数之和为 8

生成函数是啥

- 两个普通生成函数的乘积 $A(x) \times B(x)$ 可以理解为，从 A 中选择一个元素，再从 B 中选取一个元素，然后再将它们拼接起来
- 比如说，有两个不相同的骰子，问有多少种方式使得两个骰子朝上的点数之和为 8
- 答案的生成函数为 $(x + x^2 + x^3 + x^4 + x^5 + x^6)^2$

生成函数是啥

- 两个指数生成函数的乘积 $\hat{A}(x) \times \hat{B}(x)$ 可以理解为，$\hat{A}$ 中的元素内部已经分配好了编号，$\hat{B}$ 中的元素也已经分配好了编号。现在要将这两个序列中的元素组合起来，并且重新分配编号，且不能改变 $\hat{A}, \hat{B}$ 中编号大小的相对顺序

生成函数是啥

- 两个指数生成函数的乘积 $\hat{A}(x) \times \hat{B}(x)$ 可以理解为，$\hat{A}$ 中的元素内部已经分配好了编号，$\hat{B}$ 中的元素也已经分配好了编号。现在要将这两个序列中的元素组合起来，并且重新分配编号，且不能改变 $\hat{A}, \hat{B}$ 中编号大小的相对顺序

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

$$\frac{C_n}{n!} = \sum_{i+j=n} \frac{A_i}{i!} \times \frac{B_j}{j!}$$

$$C_n = \sum_{i+j=n} A_i \times B_j \times \binom{i+j}{i}$$

非常小的练习（我找不到题了）

- 为了保证大家都在线，这里有两个简单的小练习
- 设 a_n 表示将 n 个完全相同的球放入编号为 $1 \sim 5$ 的篮子的方案数，其中编号为 $1, 2$ 的篮子中的球数必须是偶数，剩下的三个篮子每个篮子中的球数必须介于 $3 \sim 5$ 之间，求 a_n 的 OGF
- 设 a_n 表示将 n 个不同的球放入 k 个不同的盒子的方案数，盒子可以是空的，求 a_n 的 EGF

非常重要的展开式

- 二项式定理
- $1+x+x^2+x^3+\cdots=\frac{1}{1-x}$
- $x+\frac{x^2}{2}+\frac{x^3}{3}+\cdots=-\ln(1-x)$
- $1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+\cdots=\exp(x)$

多项式求逆

问题

已知 $A(x) \times B(x) \equiv 1 \pmod{x^n}$，给出 $A(x)$，求 $B(x)$

保证 $A_0 \neq 0$

多项式求逆

问题

已知 $A(x) \times B(x) \equiv 1 \pmod{x^n}$，给出 $A(x)$，求 $B(x)$
保证 $A_0 \neq 0$

- 假设已经求出了 $B'(x)$ 满足 $A(x) \times B'(x) \equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}}$

多项式求逆

问题

已知 $A(x) \times B(x) \equiv 1 \pmod{x^n}$，给出 $A(x)$，求 $B(x)$

保证 $A_0 \neq 0$

- 假设已经求出了 $B'(x)$ 满足 $A(x) \times B'(x) \equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}}$

$$
\begin{aligned}
A(x) \times B(x) &\equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}} \\
\Rightarrow B(x) &\equiv B'(x) \pmod{x^{\lceil \frac{n}{2} \rceil}}
\end{aligned}
$$

多项式求逆

问题

已知 $A(x) \times B(x) \equiv 1 \pmod{x^n}$，给出 $A(x)$，求 $B(x)$
保证 $A_0 \neq 0$

- 假设已经求出了 $B'(x)$ 满足 $A(x) \times B'(x) \equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}}$

$$
\begin{aligned}
A(x) \times B(x) &\equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}} \\
\Rightarrow B(x) &\equiv B'(x) \pmod{x^{\lceil \frac{n}{2} \rceil}} \\
B(x)^2 - 2B(x)B'(x) + B'(x)^2 &\equiv 0 \pmod{x^n}
\end{aligned}
$$

多项式求逆

问题

已知 $A(x) \times B(x) \equiv 1 \pmod{x^n}$，给出 $A(x)$，求 $B(x)$
保证 $A_0 \neq 0$

- 假设已经求出了 $B'(x)$ 满足 $A(x) \times B'(x) \equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}}$

$$
\begin{aligned}
A(x) \times B(x) &\equiv 1 \pmod{x^{\lceil \frac{n}{2} \rceil}} \\
\Rightarrow B(x) &\equiv B'(x) \pmod{x^{\lceil \frac{n}{2} \rceil}} \\
B(x)^2 - 2B(x)B'(x) + B'(x)^2 &\equiv 0 \pmod{x^n} \\
B(x) - 2B'(x) + B'(x)^2 A(x) &\equiv 0 \pmod{x^n} \\
2B'(x) - B'(x)^2 A(x) &\equiv B(x) \pmod{x^n}
\end{aligned}
$$

多项式 $\ln$

问题

已知 $A(x) \equiv \ln B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

多项式 ln

问题

已知 $A(x) \equiv \ln B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

- 相信大家都没有忘记小学三年级教的求导和积分

多项式 ln

问题

已知 $A(x) \equiv \ln B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

- 相信大家都没有忘记小学三年级教的求导和积分

$$A'(x) = \frac{B'(x)}{B(x)}$$

多项式 ln

问题

已知 $A(x) \equiv \ln B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

- 相信大家都没有忘记小学三年级教的求导和积分

$$A'(x) = \frac{B'(x)}{B(x)}$$

- 对 $B(x)$ 求逆之后做一次 NTT，再积分回去即可科技为了你

关于泰勒展开

- 将多项式 $F(x)$ 展开为无穷级数的形式

关于泰勒展开

- 将多项式 $F(x)$ 展开为无穷级数的形式
- 每一个 $F(x)$，都有其唯一确定的导函数 $F'(x)$
- 比如 $F(x) = x^3 + 3$，我们可以得到 $F'(x) = 3x^2$

关于泰勒展开

- 将多项式 $F(x)$ 展开为无穷级数的形式
- 每一个 $F(x)$，都有其唯一确定的导函数 $F'(x)$
- 比如 $F(x) = x^3 + 3$，我们可以得到 $F'(x) = 3x^2$
- 但是我们不能逆用这个过程，因为有很多函数的导函数都是 $F'(x) = 3x^2$，具体来说，这类函数是 $F(x) = x^3 + C$

关于泰勒展开

- 将多项式 $F(x)$ 展开为无穷级数的形式
- 每一个 $F(x)$，都有其唯一确定的导函数 $F'(x)$
- 比如 $F(x)=x^3+3$，我们可以得到 $F'(x)=3x^2$
- 但是我们不能逆用这个过程，因为有很多函数的导函数都是 $F'(x)=3x^2$，具体来说，这类函数是 $F(x)=x^3+C$
- 有一句话说得很好：原函数的信息 $=$ 导函数的信息 $+$ 初始值信息

关于泰勒展开

- 将多项式 $F(x)$ 展开为无穷级数的形式
- 每一个 $F(x)$，都有其唯一确定的导函数 $F'(x)$
- 比如 $F(x)=x^3+3$，我们可以得到 $F'(x)=3x^2$
- 但是我们不能逆用这个过程，因为有很多函数的导函数都是 $F'(x)=3x^2$，具体来说，这类函数是 $F(x)=x^3+C$
- 有一句话说得很好：原函数的信息 $=$ 导函数的信息 $+$ 初始值信息

关于泰勒展开

- 我们选择一个点 x_0 来获取 $F(x)$ 的初始值信息，我们将 x_0 称为展开点

关于泰勒展开

- 我们选择一个点 x_0 来获取 $F(x)$ 的初始值信息，我们将 x_0 称为展开点
- 接下来考虑 $x_0 = 0$ 的情况

关于泰勒展开

- 我们选择一个点 x_0 来获取 $F(x)$ 的初始值信息，我们将 x_0 称为展开点
- 接下来考虑 $x_0 = 0$ 的情况

$$\begin{aligned} F(x) &= \int_0^x F'(x)\,\mathrm{d}x + F(0) \\ &= \int_0^x \left[\int_0^x F''(x)\,\mathrm{d}x + F'(0)\right] \mathrm{d}x + F(0) \\ &= \int_0^x \int_0^x F''(x)\,\mathrm{d}x\,\mathrm{d}x + \int_0^x F'(0)\,\mathrm{d}x + F(0) \end{aligned}$$

多来几项

- 可以发现最终 $F(x)$ 被展开成了如下的形式

多来几项

- 可以发现最终 $F(x)$ 被展开成了如下的形式

$$
\begin{aligned}
F(x) &= F(0) + \int_0^x F'(0)\,\mathrm{d}x + \int_0^x \int_0^x F''(0)\,\mathrm{d}x\,\mathrm{d}x + \cdots \\
&= F(0) + \frac{F'(0)x}{1} + \frac{F''(0)x^2}{1 \times 2} + \cdots \\
&= \sum_{i \geq 0} \frac{F^{(i)}(0)x^i}{i!}
\end{aligned}
$$

更一般的情况

- 假如展开点不是 0，而是 a，该怎么办

更一般的情况

- 假如展开点不是 0，而是 a，该怎么办

$$
\begin{aligned}
F(x) &= \int_a^x F'(x)\,\mathrm{d}x + F(a) \\
&= F(a) + \int_a^x F'(a)\,\mathrm{d}x + \int_a^x \int_a^x F''(a)\,\mathrm{d}x\,\mathrm{d}x + \cdots
\end{aligned}
$$

更一般的情况

- 假如展开点不是 0，而是 a，该怎么办

$$
\begin{aligned}
F(x) &= \int_a^x F'(x)\,\mathrm{d}x + F(a) \\
&= F(a) + \int_a^x F'(a)\,\mathrm{d}x + \int_a^x \int_a^x F''(a)\,\mathrm{d}x\,\mathrm{d}x + \cdots
\end{aligned}
$$

- 令 $u = x - a$，应用换元积分

更一般的情况

- 假如展开点不是 0，而是 a，该怎么办

$$F(x) = \int_a^x F'(x)\,\mathrm{d}x + F(a)$$
$$= F(a) + \int_a^x F'(a)\,\mathrm{d}x + \int_a^x \int_a^x F''(a)\,\mathrm{d}x\,\mathrm{d}x + \cdots$$

- 令 $u = x - a$，应用换元积分

$$F(x) = F(a) + \int_0^u F'(a)\,\mathrm{d}u + \int_0^u \int_0^u F''(a)\,\mathrm{d}u\,\mathrm{d}u + \cdots$$
$$= \sum_{i\geq 0} \frac{F^{(i)}(a)u^i}{i!} = \sum_{i\geq 0} \frac{F^{(i)}(a)(x-a)^i}{i!}$$

牛顿迭代

问题

已知 $G[F(x)] \equiv 0 \pmod{x^n}$，给出 $G(x)$，求 $F(x)$

牛顿迭代

问题

已知 $G[F(x)] \equiv 0 \pmod{x^n}$，给出 $G(x)$，求 $F(x)$

- 记 $F_t(x)$ 表示经过 t 次迭代之后的 $F(x)$，满足

$$G[F_t(x)] \equiv 0 \pmod{x^{2^t}}$$

牛顿迭代

问题

已知 $G[F(x)] \equiv 0 \pmod{x^n}$，给出 $G(x)$，求 $F(x)$

- 记 $F_t(x)$ 表示经过 t 次迭代之后的 $F(x)$，满足

$$G[F_t(x)] \equiv 0 \pmod{x^{2^t}}$$

- 考虑泰勒展开公式，对于函数 $H(x)$，选择展开点 x_0，那么有

$$H(x) = \sum_{i \geq 0} \frac{H^{(i)}(x_0)}{i!}(x - x_0)^i$$

牛顿迭代

- 考虑将 $G[F_{t+1}(x)]$ 从 $F_t(x)$ 处展开

牛顿迭代

- 考虑将 $G[F_{t+1}(x)]$ 从 $F_t(x)$ 处展开

$$
\begin{aligned}
G[F_{t+1}(x)] &= \sum_{i\geq 0} \frac{G^{(i)}[F_t(x)]}{i!}[F_{t+1}(x) - F_t(x)]^i \\
&= G[F_t(x)] + G'[F_t(x)] \times [F_{t+1}(x) - F_t(x)] + \sum_{i\geq 2} \cdots
\end{aligned}
$$

牛顿迭代

- 考虑将 $G[F_{t+1}(x)]$ 从 $F_t(x)$ 处展开

$$
\begin{aligned}
G[F_{t+1}(x)] &= \sum_{i\geq 0} \frac{G^{(i)}[F_t(x)]}{i!}[F_{t+1}(x) - F_t(x)]^i \\
&= G[F_t(x)] + G'[F_t(x)] \times [F_{t+1}(x) - F_t(x)] + \sum_{i\geq 2} \cdots \\
0 &\equiv G[F_t(x)] + G'[F_t(x)] \times [F_{t+1}(x) - F_t(x)] \pmod{x^{2^{t+1}}} \\
F_{t+1}(x) &\equiv F_t(x) - \frac{G[F_t(x)]}{G'[F_t(x)]} \pmod{x^{2^{t+1}}}
\end{aligned}
$$

牛顿迭代

- 考虑将 $G[F_{t+1}(x)]$ 从 $F_t(x)$ 处展开

$$
\begin{aligned}
G[F_{t+1}(x)] &= \sum_{i\geq 0} \frac{G^{(i)}[F_t(x)]}{i!}[F_{t+1}(x) - F_t(x)]^i \\
&= G[F_t(x)] + G'[F_t(x)] \times [F_{t+1}(x) - F_t(x)] + \sum_{i\geq 2} \cdots \\
0 &\equiv G[F_t(x)] + G'[F_t(x)] \times [F_{t+1}(x) - F_t(x)] \pmod{x^{2^{t+1}}} \\
F_{t+1}(x) &\equiv F_t(x) - \frac{G[F_t(x)]}{G'[F_t(x)]} \pmod{x^{2^{t+1}}}
\end{aligned}
$$

- 注意这里的 $G'[F_t(x)]$ 是先求出 $G'(x)$，再代入 $F_t(x)$，并非对 $G[F_t(x)]$ 求导

多项式 exp

问题

已知 $A(x) \equiv \exp B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

多项式 exp

问题

已知 $A(x) \equiv \exp B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

- 构造函数 $G(x)$ 满足 $G[A(x)] = 0$

多项式 exp

问题

已知 $A(x) \equiv \exp B(x) \pmod{x^n}$，**给出** $B(x)$，**求** $A(x)$

- **构造函数** $G(x)$ **满足** $G[A(x)] = 0$
- $G[A(x)] = \ln A(x) - B(x)$

多项式 exp

问题

已知 $A(x) \equiv \exp B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

- 构造函数 $G(x)$ 满足 $G[A(x)] = 0$
- $G[A(x)] = \ln A(x) - B(x)$
- 应用牛顿迭代公式，可以得到

多项式 exp

问题

已知 $A(x) \equiv \exp B(x) \pmod{x^n}$，给出 $B(x)$，求 $A(x)$

- 构造函数 $G(x)$ 满足 $G[A(x)] = 0$
- $G[A(x)] = \ln A(x) - B(x)$
- 应用牛顿迭代公式，可以得到

$$G'[A(x)] = \frac{1}{A(x)}$$

$$\begin{aligned} F_{t+1}(x) &\equiv F_t(x) - \frac{G[F_t(x)]}{G'[F_t(x)]} \pmod{x^{2^{t+1}}} \\ &\equiv F_t(x)[1 - \ln F_t(x) + B(x)] \pmod{x^{2^{t+1}}} \end{aligned}$$

来 k 道题

经典问题

有若干种颜色不同的骨牌，其中大小为 $1 \times i$ 的骨牌共有 a_i 种，每种骨牌数量无限，问恰好填满 $1 \times n$ 的格子的方法。颜色相同的骨牌之间没有区别

$$a_i, n \le 10^5$$

经典问题

- 设 a 的 OGF 为 $A(x)$，可以看出答案的 OGF 为

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

- 非常良心

另一道经典问题

论文中的题

给定集合 S 和正整数 n，求有多少个 n 阶置换 p，满足 p 分解之和每一个轮换的大小都在 S 内

$$n \leq 10^5$$

论文中的题

- 置换与分解出的轮换一一对应

论文中的题

- 置换与分解出的轮换一一对应
- 假设原置换分解出的一个轮换长度为 n，那么这个轮换中的元素一共有 $(n-1)!$ 种排列方式

论文中的题

- 置换与分解出的轮换一一对应
- 假设原置换分解出的一个轮换长度为 n，那么这个轮换中的元素一共有 $(n-1)!$ 种排列方式
- 原置换由若干个这样的轮换组成，且轮换与轮换之间是无序的

论文中的题

- 置换与分解出的轮换一一对应
- 假设原置换分解出的一个轮换长度为 n，那么这个轮换中的元素一共有 $(n-1)!$ 种排列方式
- 原置换由若干个这样的轮换组成，且轮换与轮换之间是无序的
- 写出一个轮换的 EGF

论文中的题

- 置换与分解出的轮换一一对应
- 假设原置换分解出的一个轮换长度为 n，那么这个轮换中的元素一共有 $(n-1)!$ 种排列方式
- 原置换由若干个这样的轮换组成，且轮换与轮换之间是无序的
- 写出一个轮换的 EGF

$$\hat{F}(x)=\sum_{i\in S}(i-1)!\frac{x^i}{i!}$$

论文中的题

- 枚举最终的置换是由多少个轮换组成的，由于轮换是无序的，因此要除以一个阶乘

论文中的题

- 枚举最终的置换是由多少个轮换组成的，由于轮换是无序的，因此要除以一个阶乘

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

- 喜闻乐见的 exp

经典问题

论文中的题 *2

你有若干种不同的物品，其中体积为 i 的物品有 a_i 种，每种物品有无限个。求选取物品恰好装满容量为 n 的背包的方案数

$$n \leq 10^5$$

经典问题

- 设 $F(x)$ 是答案的 OGF，$G(x)$ 是 $\{a_i\}$ 的 OGF

经典问题

- 设 $F(x)$ 是答案的 OGF，$G(x)$ 是 $\{a_i\}$ 的 OGF

$$
\begin{aligned}
F(x) &= \prod_{i=1}^{n}(1+x^i+x^{2i}+x^{3i}+\cdots)^{a_i} \\
&= \prod_{i=1}^{n}(\frac{1}{1-x^i})^{a_i} \\
&= \exp\left(-\sum_{i=1}^{n} a_i \ln(1-x^i)\right) \\
&= \exp\left(\sum_{i=1}^{n} a_i \sum_{j\geq 1} \frac{x^{ij}}{j}\right) \\
&= \exp\left(\sum_{j\geq 1} \frac{G(x^j)}{j}\right)
\end{aligned}
$$

一道奇妙的题

一道来自纪中的题

- n 个格子排成一行，你可以进行 m 次操作。
- 在第 i 次操作中，你可以选择两个下标 $l, r(l \leq r)$，然后将第 l 个到第 r 个格子都染成颜色 i，要求这 m 次操作之后所有格子都被染上了颜色
- 问最后可能出现的颜色序列一共有多少种

$$n, m \leq 10^6$$

一道奇妙的题

- 首先如何判断一个序列是否合法

一道奇妙的题

- 首先如何判断一个序列是否合法
- 最后一次加入的颜色必定在序列中出现。如果删去最后一次加入的颜色，那么倒数第二种颜色要么不出现，要么一定是连续的

一道奇妙的题

- 首先如何判断一个序列是否合法
- 最后一次加入的颜色必定在序列中出现。如果删去最后一次加入的颜色，那么倒数第二种颜色要么不出现，要么一定是连续的
- 设 $dp[i][j]$ 表示考虑了前 i 种颜色，共有 j 个格子被染过色的方案数

一道奇妙的题

- 首先如何判断一个序列是否合法
- 最后一次加入的颜色必定在序列中出现。如果删去最后一次加入的颜色，那么倒数第二种颜色要么不出现，要么一定是连续的
- 设 $dp[i][j]$ 表示考虑了前 i 种颜色，共有 j 个格子被染过色的方案数

$$dp[i][j] = dp[i-1][j] + \sum_{k=0}^{j-1} dp[i-1][k] \times (k+1)$$

一道奇妙的题

- 看起来非常不可优化

一道奇妙的题

- 看起来非常不可优化
- 先把 $\sum$ 前面的那个当前颜色不染色的情况拿掉，即假设在最终的序列中每种颜色都出现了，最后乘一个组合数即可

一道奇妙的题

- 看起来非常不可优化
- 先把 $\sum$ 前面的那个当前颜色不染色的情况拿掉，即假设在最终的序列中每种颜色都出现了，最后乘一个组合数即可

$$
\begin{aligned}
dp[i][j] &= \sum_{k=0}^{j-1} dp[i-1][k] \times (k+1) \\
&= \sum_{k=0}^{j-2} dp[i-1][k] \times (k+1) + dp[i-1][j-1] \times j \\
&= dp[i][j-1] + dp[i-1][j-1] \times j
\end{aligned}
$$

一道奇妙的题

- 将 dp 看作是在 $n \times m$ 的格子上走，这个网格有 n 行 m 列

一道奇妙的题

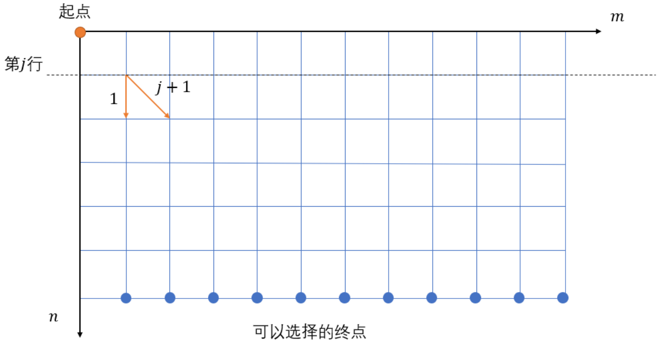
- 将 dp 看作是在 $n \times m$ 的格子上走，这个网格有 n 行 m 列
- 注意这里行相当于 dp 的第二维，列相当于 dp 的第一维

一道奇妙的题

- 将 dp 看作是在 $n \times m$ 的格子上走，这个网格有 n 行 m 列
- 注意这里行相当于 dp 的第二维，列相当于 dp 的第一维
- 那么相当于从 $(0,0)$ 出发，每次可以往下走一格，方案数为 1；或者往右下走一格，方案数为 $j+1$

一道奇妙的题

- 将 dp 看作是在 $n \times m$ 的格子上走，这个网格有 n 行 m 列
- 注意这里行相当于 dp 的第二维，列相当于 dp 的第一维
- 那么相当于从 $(0,0)$ 出发，每次可以往下走一格，方案数为 1；或者往右下走一格，方案数为 $j+1$



一道奇妙的题

- 我们需要分别求出，到达每个终点的所有路径的权值之和

一道奇妙的题

- 我们需要分别求出，到达每个终点的所有路径的权值之和
- 可以发现每一步都必然会导致所在的行 $+1$，因此我们考虑按行构造生成函数

一道奇妙的题

- 我们需要分别求出，到达每个终点的所有路径的权值之和
- 可以发现每一步都必然会导致所在的行 $+1$，因此我们考虑按行构造生成函数
- 我们用 x 的指数来代表当前所在的列数，每次我们要么使得当前的列不变，要么使得当前的列 $+1$

一道奇妙的题

- 我们需要分别求出，到达每个终点的所有路径的权值之和
- 可以发现每一步都必然会导致所在的行 $+1$，因此我们考虑按行构造生成函数
- 我们用 x 的指数来代表当前所在的列数，每次我们要么使得当前的列不变，要么使得当前的列 $+1$
- 因此 $F_i = dp[i][n]$ 的 OGF 为

$$F(x) = x\prod_{i=1}^{n-1}(1+(i+1)x)$$

- 注意第一步必定是由 $(0,0)$ 走到 $(1,1)$，因此这里要乘上 x

一道奇妙的题

- 很不幸，这个东西仍然不好求

一道奇妙的题

- 很不幸，这个东西仍然不好求
- 再将这个多项式转化一下

一道奇妙的题

- 很不幸，这个东西仍然不好求
- 再将这个多项式转化一下
- 由于总步数为 n，因此如果我们知道了向下走了多少步，那么我们也就知道了向右下走了多少步

一道奇妙的题

- 很不幸，这个东西仍然不好求
- 再将这个多项式转化一下
- 由于总步数为 n，因此如果我们知道了向下走了多少步，那么我们也就知道了向右下走了多少步
- 将向下走一步看作 x，那么最终多项式中 x^i 的系数就是原多项式中 x^{n-i} 的系数

一道奇妙的题

- 很不幸，这个东西仍然不好求
- 再将这个多项式转化一下
- 由于总步数为 n，因此如果我们知道了向下走了多少步，那么我们也就知道了向右下走了多少步
- 将向下走一步看作 x，那么最终多项式中 x^i 的系数就是原多项式中 x^{n-i} 的系数

$$F_1(x) = \prod_{i=1}^{n-1}(x+i+1)$$

这种奇形怪状的东西是可以求的

- 记 $G_t(x) = \prod_{i=1}^{2^t}(x+i+1)$

这种奇形怪状的东西是可以求的

- 记 $G_t(x) = \prod_{i=1}^{2^t}(x+i+1)$
- 如果我们能算出所有的 G，那么我们就可以用类似快速幂的方法算出原多项式

这种奇形怪状的东西是可以求的

- 记 $G_t(x) = \prod_{i=1}^{2^t}(x+i+1)$
- 如果我们能算出所有的 G，那么我们就可以用类似快速幂的方法算出原多项式

$$G_{t+1}(x) = G_t(x) \times G_t(x+2^t)$$

这种奇形怪状的东西是可以求的

- 记 $G_t(x) = \prod_{i=1}^{2^t}(x+i+1)$
- 如果我们能算出所有的 G，那么我们就可以用类似快速幂的方法算出原多项式

$$G_{t+1}(x) = G_t(x) \times G_t(x+2^t)$$

- 记 $a_i = [x^i]G_t(x), b = 2^t$

这种奇形怪状的东西是可以求的

- 记 $G_t(x) = \prod_{i=1}^{2^t}(x+i+1)$
- 如果我们能算出所有的 G，那么我们就可以用类似快速幂的方法算出原多项式

$$G_{t+1}(x) = G_t(x) \times G_t(x+2^t)$$

- 记 $a_i = [x^i]G_t(x), b = 2^t$

$$G_t(x+b) = \sum_{i\geq 0} a_i \times (x+b)^i$$

这种奇形怪状的东西是可以求的

- 记 $G_t(x) = \prod_{i=1}^{2^t}(x+i+1)$
- 如果我们能算出所有的 G，那么我们就可以用类似快速幂的方法算出原多项式

$$G_{t+1}(x) = G_t(x) \times G_t(x+2^t)$$

- 记 $a_i = [x^i]G_t(x), b = 2^t$

$$G_t(x+b) = \sum_{i\geq 0} a_i \times (x+b)^i$$

- 二项式定理展开

这一页是展开

$$
\begin{aligned}
G_t(x+b) &= \sum_{i\geq 0} a_i \times (x+b)^i \\
&= \sum_{i\geq 0} a_i \sum_{j=0}^{i} \binom{i}{j} x^j b^{i-j} \\
&= \sum_{j\geq 0} x^j \sum_{i\geq j} a_i b^{i-j} \frac{i!}{j!(i-j)!} \\
&= \sum_{j\geq 0} \frac{x^j}{j!} \sum_{i\geq j} (a_i i!) \times \frac{b^{i-j}}{(i-j)!}
\end{aligned}
$$

这一页是展开

$$
\begin{aligned}
G_t(x+b) &= \sum_{i\geq 0} a_i \times (x+b)^i \\
&= \sum_{i\geq 0} a_i \sum_{j=0}^{i} \binom{i}{j} x^j b^{i-j} \\
&= \sum_{j\geq 0} x^j \sum_{i\geq j} a_i b^{i-j} \frac{i!}{j!(i-j)!} \\
&= \sum_{j\geq 0} \frac{x^j}{j!} \sum_{i\geq j} (a_i i!) \times \frac{b^{i-j}}{(i-j)!}
\end{aligned}
$$

- 记 $A(x) = \sum_{i\geq 0} a_i i! x^{n-i}$, $B(x) = \sum_{i\geq 0} \frac{b^i}{i!} x^i$
- 第 j 项的 $\sum$ 为 $A(x) \times B(x)$ 的第 $n-j$ 项的系数

又一道题

一道不知道从哪里来的题

- 有一个离散随机变量 x，一开始它有 p_i 的概率是 i
- 现在要对 x 进行 m 次操作，每次会将 x 等概率地变为 $[0, x]$ 中的一个整数
- 问最后 x 变为 $0, 1, 2, \cdots n$ 的概率

$$n \leq 2.5 \times 10^5, m \leq 10^{18}$$

一道不知道从哪里来的题

- 设一开始 x 的概率生成函数为 $F(x)$，经过一次操作之后的概率生成函数为 $F_1(x)$

一道不知道从哪里来的题

- 设一开始 x 的概率生成函数为 $F(x)$，经过一次操作之后的概率生成函数为 $F_1(x)$

$$[x^i]F_1(x) = \sum_{j=i}^{n} \frac{[x^j]F(x)}{j+1}$$

$$\begin{aligned} F_1(x) &= \sum_{i=0}^{n} x^i \sum_{j=i}^{n} \frac{[x^j]F(x)}{j+1} \\ &= \sum_{j=0}^{n} \frac{[x^j]F(x)}{j+1} \sum_{i=0}^{j} x^i \\ &= \frac{1}{x-1} \sum_{j=0}^{n} [x^j]F(x) \frac{x^{j+1}-1}{j+1} \end{aligned}$$

一些神仙操作

- 有一件事情

一些神仙操作

- 有一件事情

$$\frac{x^{j+1}-1}{j+1}=\left[\frac{t^{j+1}}{j+1}\right]_1^x=\int_1^x t^j \,\mathrm{d}x$$

一些神仙操作

- 有一件事情

$$\frac{x^{j+1}-1}{j+1}=\left[\frac{t^{j+1}}{j+1}\right]_1^x=\int_1^x t^j\,\mathrm{d}x$$

- 嘿嘿嘿

$$\begin{aligned}F_1(x)&=\frac{1}{x-1}\sum_{j=0}^{n}[x^j]F(x)\int_1^x t^j\,\mathrm{d}x\\&=\frac{1}{x-1}\int_1^x\sum_{j=0}^{n}[x^j]F(x)t^j\,\mathrm{d}x\\&=\frac{1}{x-1}\int_1^x F(t)\,\mathrm{d}x\end{aligned}$$

转化

- 这个东西的分母是 $x-1$，而且积分是从 1 开始积的。$x-1$ 意味着这个分母不好消，从 1 开始积意味着我们需要算出 $F(1)$

转化

- 这个东西的分母是 $x-1$，而且积分是从 1 开始积的。$x-1$ 意味着这个分母不好消，从 1 开始积意味着我们需要算出 $F(1)$
- 设 $G(x)=F(x+1), G_1(x)=F_1(x+1)$

转化

- 这个东西的分母是 $x-1$，而且积分是从 1 开始积的。$x-1$ 意味着这个分母不好消，从 1 开始积意味着我们需要算出 $F(1)$
- 设 $G(x)=F(x+1), G_1(x)=F_1(x+1)$

$$G_1(x)=F_1(x+1)=\frac{1}{x}\int_1^{x+1}F(t)\,\mathrm{d}x=\frac{1}{x}\int_0^x G(t)\,\mathrm{d}x$$

- 反正我想不到

继续推式子

- 化简这个式子，把积分弄掉

继续推式子

- 化简这个式子，把积分弄掉

$$\begin{aligned}\frac{1}{x}\int_0^x G(t)\,\mathrm{d}t &= \frac{1}{x}\int_0^x \sum_{i=0}^{n} g_i t^i\,\mathrm{d}t\\ &= \frac{1}{x}\sum_{i=0}^{n}\int_0^x g_i t^i\,\mathrm{d}t\\ &= \frac{1}{x}\sum_{i=0}^{n}[\frac{g_i}{i+1}t^{i+1}]_0^x\\ &= \frac{1}{x}\sum_{i=0}^{n}\frac{g_i}{i+1}x^{i+1}\\ &= \sum_{i=0}^{n}\frac{g_i}{i+1}x^i\end{aligned}$$

式子推完了

- 这意味着我们从 $G(x)$ 迭代到 $G_1(x)$ 时，第 i 项的系数会除以 $i+1$
- 迭代 m 次之后会除以 $(i+1)^m$，因此我们可以快速算出 $G_m(x)$

式子推完了

- 这意味着我们从 $G(x)$ 迭代到 $G_1(x)$ 时，第 i 项的系数会除以 $i+1$
- 迭代 m 次之后会除以 $(i+1)^m$，因此我们可以快速算出 $G_m(x)$
- 现在唯一的问题就是怎么算出 $G(x)=F(x+1)$ 以及 $F_m(x)=G_m(x-1)$

式子推完了

- 这意味着我们从 $G(x)$ 迭代到 $G_1(x)$ 时，第 i 项的系数会除以 $i+1$
- 迭代 m 次之后会除以 $(i+1)^m$，因此我们可以快速算出 $G_m(x)$
- 现在唯一的问题就是怎么算出 $G(x) = F(x+1)$ 以及 $F_m(x) = G_m(x-1)$
- 二项式定理展开

式子推完了

- 这意味着我们从 $G(x)$ 迭代到 $G_1(x)$ 时，第 i 项的系数会除以 $i+1$
- 迭代 m 次之后会除以 $(i+1)^m$，因此我们可以快速算出 $G_m(x)$
- 现在唯一的问题就是怎么算出 $G(x)=F(x+1)$ 以及 $F_m(x)=G_m(x-1)$
- 二项式定理展开

又一道题

经典问题

有若干种颜色不同的珠子，其中重量为 i 的珠子一共有 a_i 种，每种珠子都可以无限量使用。用珠子串成一条重量为 n 的项链，共有多少种方法？项链允许旋转，但不允许翻转

$$a_i, n \leq 10^5$$

Polya!

- 枚举最终项链上有多少颗珠子，假设为 k 颗

Polya!

- 枚举最终项链上有多少颗珠子，假设为 k 颗
- 设 $\{a_i\}$ 的 OGF 为 $G(x)$，包含 k 颗珠子的项链的 OGF 为 $F(x)$

Polya!

- 枚举最终项链上有多少颗珠子，假设为 k 颗
- 设 $\{a_i\}$ 的 OGF 为 $G(x)$，包含 k 颗珠子的项链的 OGF 为 $F(x)$

$$\begin{aligned} F(x) &= \frac{1}{k}\sum_{i=1}^{k} G(x)^{\gcd(i,k)}[x^{\frac{n\times\gcd(i,k)}{k}}] \\ &= \frac{1}{k}\sum_{i=1}^{k} G(x^{\frac{k}{\gcd(i,k)}})^{\gcd(i,k)}[x^n] \end{aligned}$$

- 可以理解为每次在 $\frac{k}{\gcd(i,k)}$ 个位置放珠子

Polya!

- 记答案的 OGF 为 $A(x)$

Polya!

- 记答案的 OGF 为 $A(x)$

$$A(x) = \sum_{k\geq 1} \frac{1}{k} \sum_{i=1}^{k} G(x^{\frac{k}{\gcd(i,k)}})^{\gcd(i,k)}$$

Polya!

- 记答案的 OGF 为 $A(x)$

$$\begin{aligned} A(x) &= \sum_{k\geq 1} \frac{1}{k} \sum_{i=1}^{k} G\left(x^{\frac{k}{\gcd(i,k)}}\right)^{\gcd(i,k)} \\ &= \sum_{k\geq 1} \frac{1}{k} \sum_{i\mid k} \varphi(i)\, G(x^i)^{\frac{k}{i}} \end{aligned}$$

Polya!

- 记答案的 OGF 为 $A(x)$

$$
\begin{aligned}
A(x) &= \sum_{k\ge 1}\frac{1}{k}\sum_{i=1}^{k} G\left(x^{\frac{k}{\gcd(i,k)}}\right)^{\gcd(i,k)} \\
&= \sum_{k\ge 1}\frac{1}{k}\sum_{i\mid k}\varphi(i)\,G(x^i)^{\frac{k}{i}} \\
&= \sum_{i\ge 1}\varphi(i)\sum_{j\ge 1}\frac{G(x^i)^j}{ij}
\end{aligned}
$$

Polya!

- 记答案的 OGF 为 $A(x)$

$$
\begin{aligned}
A(x) &= \sum_{k\geq 1}\frac{1}{k}\sum_{i=1}^{k} G(x^{\frac{k}{\gcd(i,k)}})^{\gcd(i,k)} \\
&= \sum_{k\geq 1}\frac{1}{k}\sum_{i|k}\varphi(i)\,G(x^i)^{\frac{k}{i}} \\
&= \sum_{i\geq 1}\varphi(i)\sum_{j\geq 1}\frac{G(x^i)^j}{ij} \\
&= -\sum_{i\geq 1}\frac{\varphi(i)}{i}\ln(1-G(x^i))
\end{aligned}
$$

Polya!

- 记答案的 OGF 为 $A(x)$

$$\begin{aligned}
A(x) &= \sum_{k\ge 1}\frac{1}{k}\sum_{i=1}^{k} G\left(x^{\frac{k}{\gcd(i,k)}}\right)^{\gcd(i,k)} \\
&= \sum_{k\ge 1}\frac{1}{k}\sum_{i|k}\varphi(i)\, G(x^i)^{\frac{k}{i}} \\
&= \sum_{i\ge 1}\varphi(i)\sum_{j\ge 1}\frac{G(x^i)^j}{ij} \\
&= -\sum_{i\ge 1}\frac{\varphi(i)}{i}\ln(1-G(x^i))
\end{aligned}$$

- 只要算出了 $\ln(1-G(x))$，我们也就知道了 $\ln(1-G(x^i))$

原题放送

UOJ424 count

- 如果一个序列满足序列长度为 n，序列中的每个数都是 1 到 m 内的整数，且所有 1 到 m 内的整数都在序列中出现过，则称这是一个挺好序列。
- 对于一个序列 A，记 $f_A(l, r)$ 为 A 的第 l 个到第 r 个数中最大值的下标（如果有多个最大值，取下标最小的）。
- 两个序列 A 和 B 同构，当且仅当 A 和 B 长度相等，且对于任意 $i \le j$，均有 $f_A(i, j) = f_B(i, j)$。
- 给出 n, m，求有多少种不同构的挺好序列。

$$n, m \le 10^5$$

UOJ424 count

- 如果不考虑 1 到 m 均出现过这一条限制，那么两个序列不同构当且仅当它们的笛卡尔树不同

UOJ424 count

- 如果不考虑 1 到 m 均出现过这一条限制，那么两个序列不同构当且仅当它们的笛卡尔树不同
- 注意这里我们认为一个点的左儿子的权值小于这个点的权值，其右儿子的权值小于等于这个点的权值

UOJ424 count

- 如果不考虑 1 到 m 均出现过这一条限制，那么两个序列不同构当且仅当它们的笛卡尔树不同
- 注意这里我们认为一个点的左儿子的权值小于这个点的权值，其右儿子的权值小于等于这个点的权值
- 显然根到每个叶子往左走的次数不能超过 $m-1$

UOJ424 count

- 如果不考虑 1 到 m 均出现过这一条限制，那么两个序列不同构当且仅当它们的笛卡尔树不同
- 注意这里我们认为一个点的左儿子的权值小于这个点的权值，其右儿子的权值小于等于这个点的权值
- 显然根到每个叶子往左走的次数不能超过 $m-1$
- 事实上，当 $m \le n$ 时，我们并不需要保证 1 到 m 均出现过，因为此时满足最大值小于等于 m 的笛卡尔树一定能够通过调节节点的权值使得 1 到 m 均出现过

UOJ424 count

- 如果不考虑 1 到 m 均出现过这一条限制，那么两个序列不同构当且仅当它们的笛卡尔树不同
- 注意这里我们认为一个点的左儿子的权值小于这个点的权值，其右儿子的权值小于等于这个点的权值
- 显然根到每个叶子往左走的次数不能超过 $m-1$
- 事实上，当 $m \leq n$ 时，我们并不需要保证 1 到 m 均出现过，因为此时满足最大值小于等于 m 的笛卡尔树一定能够通过调节节点的权值使得 1 到 m 均出现过
- 也就是说，我们实际上要统计 n 个节点，根到每个节点的路径中往左走的次数不能超过 $m-1$ 的二叉树的数量

UOJ424 count

- 设 $f[i][j]$ 表示 j 个点，往左走的次数小于 i 的二叉树的数量，初值为 $f[0][0]=1$。这个东西写成生成函数就是

$$
\begin{aligned}
F_0(x)=1, F_i(x) &= xF_{i-1}(x)F_i(x)+1 \\
&= \frac{1}{1-xF_{i-1}(x)}
\end{aligned}
$$

UOJ424 count

- 设 $f[i][j]$ 表示 j 个点，往左走的次数小于 i 的二叉树的数量，初值为 $f[0][0]=1$。这个东西写成生成函数就是

$$
\begin{aligned}
F_0(x)=1, F_i(x) &= xF_{i-1}(x)F_i(x)+1 \\
&= \frac{1}{1-xF_{i-1}(x)}
\end{aligned}
$$

- 接着是非常神奇的一步，观察这个递推式，可以发现 $F_t(x)$ 一定可以被表示成 $\frac{A}{B}$ 的形式，这里 A, B 是两个多项式。对于 $F_0(x)$ 来说，$A=B=1$

UOJ424 count

$$
\begin{aligned}
\frac{A'}{B'} &= \frac{1}{1 - x\frac{A}{B}} \\
&= \frac{B}{B - xA} \Rightarrow A' = B, B' = B - xA
\end{aligned}
$$

UOJ424 count

$$\frac{A'}{B'} = \frac{1}{1 - x\frac{A}{B}} = \frac{B}{B - xA} \Rightarrow A' = B, B' = B - xA$$

- 模拟 FFT 的过程，代入 2^l 个点值，可以通过矩乘快速算出迭代 m 次之后的点值是啥，最后再做一次 IDFT 即可

组合恒等式

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

组合恒等式

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

$$\sum_{i=0}^{n} \binom{n}{i} = 2^n$$

组合恒等式

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

$$\sum_{i=0}^{n} \binom{n}{i} = 2^n$$

$$\sum_{i=0}^{n} \binom{n}{i} [2 \mid i] = \sum_{i=0}^{n} \binom{n}{i} [2 \nmid i] = 2^{n-1}$$

组合恒等式

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

$$\sum_{i=0}^{n} \binom{n}{i} = 2^n$$

$$\sum_{i=0}^{n} \binom{n}{i} [2 \mid i] = \sum_{i=0}^{n} \binom{n}{i} [2 \nmid i] = 2^{n-1}$$

$$\sum_{i=0}^{m} \binom{n+i}{n} = \binom{n+m+1}{m}$$

组合恒等式

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

$$\sum_{i=0}^{n} \binom{n}{i} = 2^n$$

$$\sum_{i=0}^{n} \binom{n}{i} [2 \mid i] = \sum_{i=0}^{n} \binom{n}{i} [2 \nmid i] = 2^{n-1}$$

$$\sum_{i=0}^{m} \binom{n+i}{n} = \binom{n+m+1}{m}$$

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

组合恒等式

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

$$\sum_{i=0}^{n} \binom{n}{i} = 2^n$$

$$\sum_{i=0}^{n} \binom{n}{i} [2 \mid i] = \sum_{i=0}^{n} \binom{n}{i} [2 \nmid i] = 2^{n-1}$$

$$\sum_{i=0}^{m} \binom{n+i}{n} = \binom{n+m+1}{m}$$

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

$$\sum_{i=0}^{k} \binom{n}{i}\binom{m}{k-i} = \binom{n+m}{k}$$

比较常用的转化

- 1. 将 n 个元素分为若干个集合，每种方案的权值为每个集合的大小的积
- 相当于先分为若干个集合，再从每个集合中选一个数

比较常用的转化

- 1. 将 n 个元素分为若干个集合，每种方案的权值为每个集合的大小的积
- 相当于先分为若干个集合，再从每个集合中选一个数
- 2.$X_1+X_2+\cdots+X_n=S$，对于每个 X_i 有 $X_i\geq 1$，计算每种方案的 $\prod X_i$ 之和
- 根据 1，相当于将 S 分为 n 个集合后，从每个集合内再选一个数的方案数

比较常用的转化

- 1. 将 n 个元素分为若干个集合，每种方案的权值为每个集合的大小的积
- 相当于先分为若干个集合，再从每个集合中选一个数
- 2.$X_1 + X_2 + \cdots + X_n = S$，对于每个 X_i 有 $X_i \geq 1$，计算每种方案的 $\prod X_i$ 之和
- 根据 1，相当于将 S 分为 n 个集合后，从每个集合内再选一个数的方案数
- 将每个变量拆成两个变量，分别代表这个集合所选数之前有多少个数，之后有多少个数
- $(X_1 + Y_1) + (X_2 + Y_2) + \cdots + (X_n + Y_n) = S - n$，每个变量 ≥ 1

新鲜出炉的例题

dict

- 给定一个 $1 \cdots n$ 的排列 $p_{1,2\cdots n}$，定义两个大小为 n 的不可重集合 A, B 的字典序比较方式为：
- 先比较 A 和 B 的第 p_1 消的元素，较小的那个字典序较小，否则比较第 p_2 小的元素，以此类推
- 现在给定 $p_{1,2\cdots n}$ 和一个大小为 n 的不可重集合 B，求有几个值在 $1 \cdots m$，大小为 n 的不可重集合 A 满足 A 的字典序比 B 小

$$n \le m \le 2 \times 10^5$$

dict

- 由于是比较字典序的大小，因此我们可以枚举它们的 lcp，即在哪一次比较中，A 的元素比 B 的对应元素小

dict

- 由于是比较字典序的大小，因此我们可以枚举它们的 lcp，即在哪一次比较中，A 的元素比 B 的对应元素小
- 在这一位之前，A 必须与 B 相同，在这一位之后，A 可以随便填

dict

- 由于是比较字典序的大小，因此我们可以枚举它们的 lcp，即在哪一次比较中，A 的元素比 B 的对应元素小
- 在这一位之前，A 必须与 B 相同，在这一位之后，A 可以随便填
- 假设此时比较到了第 i 位，这就意味着 $p_1, p_2, \cdots p_{i-1}$ 这些位置中的数已经确定了

dict

- 由于是比较字典序的大小，因此我们可以枚举它们的 lcp，即在哪一次比较中，A 的元素比 B 的对应元素小
- 在这一位之前，A 必须与 B 相同，在这一位之后，A 可以随便填
- 假设此时比较到了第 i 位，这就意味着 $p_1, p_2, \cdots p_{i-1}$ 这些位置中的数已经确定了
- 假设在这些位置之中，比 p_i 小的最大的位置为 p_j，比 p_i 大的最小的位置为 p_k

dict

- 由于是比较字典序的大小，因此我们可以枚举它们的 lcp，即在哪一次比较中，A 的元素比 B 的对应元素小
- 在这一位之前，A 必须与 B 相同，在这一位之后，A 可以随便填
- 假设此时比较到了第 i 位，这就意味着 $p_1, p_2, \cdots p_{i-1}$ 这些位置中的数已经确定了
- 假设在这些位置之中，比 p_i 小的最大的位置为 p_j，比 p_i 大的最小的位置为 p_k
- 显然 p_i 这个位置无论怎么填都不会影响到 p_j 之前、p_k 之后的位置，这些位置的方案数是确定的

dict

- 枚举 $A_{p_i} - A_{p_j}$，可以发现我们实际上要求的是一个类似这样的式子

dict

- 枚举 $A_{p_i} - A_{p_j}$，可以发现我们实际上要求的是一个类似这样的式子

$$\sum_{i=1}^{r} \binom{i-1}{a} \times \binom{t-i}{b}$$

dict

- 枚举 $A_{p_i} - A_{p_j}$，可以发现我们实际上要求的是一个类似这样的式子

$$\sum_{i=1}^{r} \binom{i-1}{a} \times \binom{t-i}{b}$$

- 当 a 能取到 t 的上界时，这就是一个简单的组合恒等式，它等于 $\binom{t}{a+b+1}$

dict

- 枚举 $A_{p_i} - A_{p_j}$，可以发现我们实际上要求的是一个类似这样的式子

$$\sum_{i=1}^{r} \binom{i-1}{a} \times \binom{t-i}{b}$$

- 当 a 能取到 t 的上界时，这就是一个简单的组合恒等式，它等于 $\binom{t}{a+b+1}$
- 如果 $t-a$ 比较小，我们可以暴力枚举 i，否则用总数减去不合法的情况

dict

- 枚举 $A_{p_i} - A_{p_j}$，可以发现我们实际上要求的是一个类似这样的式子

$$\sum_{i=1}^{r} \binom{i-1}{a} \times \binom{t-i}{b}$$

- 当 a 能取到 t 的上界时，这就是一个简单的组合恒等式，它等于 $\binom{t}{a+b+1}$
- 如果 $t-a$ 比较小，我们可以暴力枚举 i，否则用总数减去不合法的情况
- 这实际上是一个启发式分裂的过程，因此时间复杂度为 $O(n \log n)$

它 lei 了

青春猪头少年不会梦到兔女郎学姐

- 有 n 种球，第 i 种球有 a_i 个
- 对于一个序列，将其看作是首位相连的。一个序列的权值为每个极大相同颜色段的长度的乘积，求所有序列的权值之和
- $\sum a_i \leq 2 \times 10^5$

青春猪头少年不会梦到兔女郎学姐

- 我们先考虑这样一个问题：有 n 种球，第 i 种球有 a_i 个，规定相邻两个位置的球不能相同，问方案数

青春猪头少年不会梦到兔女郎学姐

- 我们先考虑这样一个问题：有 n 种球，第 i 种球有 a_i 个，规定相邻两个位置的球不能相同，问方案数
- 考虑第 i 种球，假设它们出现了 j 对相邻的情况，那么我们可以将这些相邻的球缩在一起。即从原来的 i 个球缩为了 $i-j$ 个球，同时乘上容斥系数 $(-1)^j\binom{i-1}{j}$，最后再将剩下的球可重排列一下

青春猪头少年不会梦到兔女郎学姐

- 我们先考虑这样一个问题：有 n 种球，第 i 种球有 a_i 个，规定相邻两个位置的球不能相同，问方案数
- 考虑第 i 种球，假设它们出现了 j 对相邻的情况，那么我们可以将这些相邻的球缩在一起。即从原来的 i 个球缩为了 $i-j$ 个球，同时乘上容斥系数 $(-1)^j\binom{i-1}{j}$，最后再将剩下的球可重排列一下
- 回到这道题，如果我们确定了每种颜色最后的分段情况是啥样的，那么就转化为了上面那个问题

青春猪头少年不会梦到兔女郎学姐

- 我们先考虑这样一个问题：有 n 种球，第 i 种球有 a_i 个，规定相邻两个位置的球不能相同，问方案数
- 考虑第 i 种球，假设它们出现了 j 对相邻的情况，那么我们可以将这些相邻的球缩在一起。即从原来的 i 个球缩为了 $i-j$ 个球，同时乘上容斥系数 $(-1)^j\binom{i-1}{j}$，最后再将剩下的球可重排列一下
- 回到这道题，如果我们确定了每种颜色最后的分段情况是啥样的，那么就转化为了上面那个问题
- 来点式子

青春猪头少年不会梦到兔女郎学姐

- 我们先只考虑链的情况，定义 $f(a,b)$ 表示将 a 个球分为 b 段，所有方案的权值之和

青春猪头少年不会梦到兔女郎学姐

- 我们先只考虑链的情况，定义 $f(a, b)$ 表示将 a 个球分为 b 段，所有方案的权值之和
- 那么一种球的 EGF 为

$$\begin{aligned}\hat{A}_i(x) &= \sum_{j=1}^{a_i} f(a_i, j) \sum_{k=1}^{j} \binom{j-1}{j-k} (-1)^{j-k} \frac{x^k}{k!} \\ &= \sum_{k=1}^{a_i} \frac{x^k}{k!} (-1)^k \sum_{j=k}^{a_i} f(a_i, j) (-1)^j \binom{j-1}{j-k}\end{aligned}$$

青春猪头少年不会梦到兔女郎学姐

- 有一个问题：$f(a, b)$ 咋求

青春猪头少年不会梦到兔女郎学姐

- 有一个问题：$f(a,b)$ 咋求
- 用之前提到的方法，可以发现 $f(a,b)=\binom{a-b-1}{2b-1}$，因此

青春猪头少年不会梦到兔女郎学姐

- 有一个问题：$f(a,b)$ 咋求
- 用之前提到的方法，可以发现 $f(a,b)=\binom{a-b-1}{2b-1}$，因此

$$\hat{A}_i(x)=\sum_{k=1}^{a_i}\frac{x^k}{k!}(-1)^k\sum_{j=k}^{a_i}f(a_i,j)(-1)^j\binom{j-1}{j-k}$$

青春猪头少年不会梦到兔女郎学姐

- 有一个问题：$f(a,b)$ 咋求
- 用之前提到的方法，可以发现 $f(a,b)=\binom{a-b-1}{2b-1}$，因此

$$\begin{aligned}\hat{A}_i(x) &= \sum_{k=1}^{a_i}\frac{x^k}{k!}(-1)^k\sum_{j=k}^{a_i} f(a_i,j)(-1)^j\binom{j-1}{j-k} \\ &= \sum_{k=1}^{a_i}\frac{x^k}{k!}(-1)^k\sum_{j=k}^{a_i}\binom{a_i-j-1}{2j-1}(-1^j)\frac{(j-1)!}{(j-k)!(k-1)!}\end{aligned}$$

青春猪头少年不会梦到兔女郎学姐

- 有一个问题：$f(a,b)$ 咋求
- 用之前提到的方法，可以发现 $f(a,b)=\binom{a-b-1}{2b-1}$，因此

$$
\begin{aligned}
\hat{A}_i(x) &= \sum_{k=1}^{a_i} \frac{x^k}{k!}(-1)^k \sum_{j=k}^{a_i} f(a_i,j)(-1)^j \binom{j-1}{j-k} \\
&= \sum_{k=1}^{a_i} \frac{x^k}{k!}(-1)^k \sum_{j=k}^{a_i} \binom{a_i-j-1}{2j-1}(-1^j)\frac{(j-1)!}{(j-k)!(k-1)!} \\
&= \sum_{k=1}^{a_i} \frac{x^k}{k!(k-1)!}(-1)^k \sum_{j=k}^{a_i} \binom{a_i-j-1}{2j-1}(-1)^j(j-1)! \times \frac{1}{(j-k)}
\end{aligned}
$$

青春猪头少年不会梦到兔女郎学姐

- 有一个问题：$f(a,b)$ 咋求
- 用之前提到的方法，可以发现 $f(a,b)=\binom{a-b-1}{2b-1}$，因此

$$\begin{aligned}\hat{A}_i(x)&=\sum_{k=1}^{a_i}\frac{x^k}{k!}(-1)^k\sum_{j=k}^{a_i}f(a_i,j)(-1)^j\binom{j-1}{j-k}\\&=\sum_{k=1}^{a_i}\frac{x^k}{k!}(-1)^k\sum_{j=k}^{a_i}\binom{a_i-j-1}{2j-1}(-1^j)\frac{(j-1)!}{(j-k)!(k-1)!}\\&=\sum_{k=1}^{a_i}\frac{x^k}{k!(k-1)!}(-1)^k\sum_{j=k}^{a_i}\binom{a_i-j-1}{2j-1}(-1)^j(j-1)!\times\frac{1}{(j-k)}\end{aligned}$$

- 由于 $\sum a_i=O(n)$，因此算出所有 $\hat{A}_i(x)$ 的复杂度为 $O(n\log n)$

青春猪头少年不会梦到兔女郎学姐

- 有环怎么办？

青春猪头少年不会梦到兔女郎学姐

- 有环怎么办？
- 我们钦定环的开头为第一种球，并且结尾不是第一种球，此时可以将环断为链，答案不变

青春猪头少年不会梦到兔女郎学姐

- 有环怎么办？
- 我们钦定环的开头为第一种球，并且结尾不是第一种球，此时可以将环断为链，答案不变
- 此时每一种方案都可以转 $m = \sum a_i$ 次，如果一种方案中第一种球被分为了 k 段，那么这种方案被计算了 k 次

青春猪头少年不会梦到兔女郎学姐

- 有环怎么办？
- 我们钦定环的开头为第一种球，并且结尾不是第一种球，此时可以将环断为链，答案不变
- 此时每一种方案都可以转 $m = \sum a_i$ 次，如果一种方案中第一种球被分为了 k 段，那么这种方案被计算了 k 次
- 我们将第一种球的 EGF 单独拿出来，如果我们钦定开头为第一种球，那么这个 EGF 里面的 $\frac{x^i}{i!}$ 应该变为 $\frac{x^{i-1}}{(i-1)!}$

青春猪头少年不会梦到兔女郎学姐

- 有环怎么办？
- 我们钦定环的开头为第一种球，并且结尾不是第一种球，此时可以将环断为链，答案不变
- 此时每一种方案都可以转 $m=\sum a_i$ 次，如果一种方案中第一种球被分为了 k 段，那么这种方案被计算了 k 次
- 我们将第一种球的 EGF 单独拿出来，如果我们钦定开头为第一种球，那么这个 EGF 里面的 $\frac{x^i}{i!}$ 应该变为 $\frac{x^{i-1}}{(i-1)!}$
- 如果我们钦定开头结尾都是第一种球，那么 $\frac{x^i}{i!}$ 应该变为 $\frac{x^{i-2}}{(i-2)!}$

青春猪头少年不会梦到兔女郎学姐

- 有环怎么办？
- 我们钦定环的开头为第一种球，并且结尾不是第一种球，此时可以将环断为链，答案不变
- 此时每一种方案都可以转 $m=\sum a_i$ 次，如果一种方案中第一种球被分为了 k 段，那么这种方案被计算了 k 次
- 我们将第一种球的 EGF 单独拿出来，如果我们钦定开头为第一种球，那么这个 EGF 里面的 $\frac{x^i}{i!}$ 应该变为 $\frac{x^{i-1}}{(i-1)!}$
- 如果我们钦定开头结尾都是第一种球，那么 $\frac{x^i}{i!}$ 应该变为 $\frac{x^{i-2}}{(i-2)!}$
- 注意要去掉多算的情况，因此要将含 j 的项除以 j

关于斯特林数

- 第一类斯特林数：将 n 个元素划分为 m 个圆排列的方案数

关于斯特林数

- 第一类斯特林数：将 n 个元素划分为 m 个圆排列的方案数
- 第二类斯特林数：将 n 个元素花费为 m 个集合的方案数

关于斯特林数

- 第一类斯特林数：将 n 个元素划分为 m 个圆排列的方案数
- 第二类斯特林数：将 n 个元素花费为 m 个集合的方案数

$$\begin{bmatrix} n \\ m \end{bmatrix} = \begin{bmatrix} n-1 \\ m-1 \end{bmatrix} + (n-1)\begin{bmatrix} n-1 \\ m \end{bmatrix}$$

$$\left\{ \begin{matrix} n \\ m \end{matrix} \right\} = \left\{ \begin{matrix} n-1 \\ m-1 \end{matrix} \right\} + m \left\{ \begin{matrix} n-1 \\ m \end{matrix} \right\}$$

$$\left\{ \begin{matrix} n \\ m \end{matrix} \right\} = \frac{1}{m!} \sum_{k=0}^{m} (-1)^k \binom{m}{k} (m-k)^n$$

一些引理以及推论

$$n^m = \sum_{i=0}^{m} \begin{Bmatrix} m \\ i \end{Bmatrix} n^{\underline{i}}$$

$$n^{\overline{m}} = \sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix} n^{i}$$

$$[n = m] = \sum_{k=m}^{n} (-1)^{n-k} \begin{bmatrix} n \\ k \end{bmatrix} \begin{Bmatrix} k \\ m \end{Bmatrix}$$

$$[n = m] = \sum_{k=m}^{n} (-1)^{n-k} \begin{Bmatrix} n \\ k \end{Bmatrix} \begin{bmatrix} k \\ m \end{bmatrix}$$

$$x^{\underline{n}} = (-1)^n (-x)^{\overline{n}}, x^{\overline{n}} = (-1)^n (-x)^{\underline{n}}$$

证明

$$n^m = \sum_{i=0}^{m} \left\{ {m \atop i} \right\} n^{\underline{i}}$$

证明

$$n^m = \sum_{i=0}^{m} \left\{ {m \atop i} \right\} n^{\underline{i}}$$

- 考虑 n^m 的组合意义，即有序地选 m 个 $1 \sim n$ 的数，可以相同的方案数

证明

$$n^m=\sum_{i=0}^{m}\left\{{m \atop i}\right\}n^{\underline{i}}$$

- 考虑 n^m 的组合意义，即有序地选 m 个 $1\sim n$ 的数，可以相同的方案数
- 枚举这 m 个数中有多少种不同的数，然后将相同的数缩在一起。此时一共有 i 个数，限制变为每个数都不能相同，因此一共有 $n^{\underline{i}}$ 种方案

证明

$$n^m=\sum_{i=0}^{m}\left\{{m \atop i}\right\}n^{\underline{i}}$$

- 考虑 n^m 的组合意义，即有序地选 m 个 $1\sim n$ 的数，可以相同的方案数
- 枚举这 m 个数中有多少种不同的数，然后将相同的数缩在一起。此时一共有 i 个数，限制变为每个数都不能相同，因此一共有 $n^{\underline{i}}$ 种方案
- 这个式子也可以写成

$$n^m=\sum_{i=0}^{m}\left\{{m \atop i}\right\}i!\binom{n}{i}$$

证明

$$n^{\underline{m}} = \sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix} n^{i}(-1)^{i-m}$$

$$n^{\overline{m}} = \sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix} n^{i}$$

证明

$$n^{\underline{m}} = \sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix} n^i (-1)^{i-m}$$

$$n^{\overline{m}} = \sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix} n^i$$

- 采用数学归纳法证明
- 这一页写不下了

证明

$$
\begin{aligned}
n^{\overline{m}} &= (n+m-1)n^{\overline{m-1}} \\
&= (n+m-1)\sum_{i=0}^{m-1}\begin{bmatrix}m-1\\i\end{bmatrix}n^i \\
&= \sum_{i=0}^{m-1}\begin{bmatrix}m-1\\i\end{bmatrix}n^{i+1}+\sum_{i=0}^{m-1}\begin{bmatrix}m-1\\i\end{bmatrix}n^i(m-1) \\
&= \sum_{i=1}^{m}\begin{bmatrix}m-1\\i-1\end{bmatrix}n^{i}+\sum_{i=0}^{m-1}\begin{bmatrix}m-1\\i\end{bmatrix}n^i(m-1) \\
&= \sum_{i=0}^{m}\left(\begin{bmatrix}m-1\\i-1\end{bmatrix}+(m-1)\begin{bmatrix}m-1\\i\end{bmatrix}\right)n^i \\
&= \sum_{i=0}^{m}\begin{bmatrix}m\\i\end{bmatrix}n^i
\end{aligned}
$$

反转公式

$$[n=m]=\sum_{k=m}^{n}(-1)^{n-k}\begin{bmatrix}n\\k\end{bmatrix}\begin{Bmatrix}k\\m\end{Bmatrix}$$

$$[n=m]=\sum_{k=m}^{n}(-1)^{n-k}\begin{Bmatrix}n\\k\end{Bmatrix}\begin{bmatrix}k\\m\end{bmatrix}$$

反转公式

$$[n=m]=\sum_{k=m}^{n}(-1)^{n-k}\begin{bmatrix}n\\k\end{bmatrix}\begin{Bmatrix}k\\m\end{Bmatrix}$$

$$[n=m]=\sum_{k=m}^{n}(-1)^{n-k}\begin{Bmatrix}n\\k\end{Bmatrix}\begin{bmatrix}k\\m\end{bmatrix}$$

- 第二个的证明是结合正常幂转下降幂、上升幂转正常幂
- 第一个的证明可以结合下降幂转正常幂、正常幂转下降幂

反转公式

$$[n=m]=\sum_{k=m}^{n}(-1)^{n-k}\begin{bmatrix}n\\k\end{bmatrix}\begin{Bmatrix}k\\m\end{Bmatrix}$$

$$[n=m]=\sum_{k=m}^{n}(-1)^{n-k}\begin{Bmatrix}n\\k\end{Bmatrix}\begin{bmatrix}k\\m\end{bmatrix}$$

- 第二个的证明是结合正常幂转下降幂、上升幂转正常幂
- 第一个的证明可以结合下降幂转正常幂、正常幂转下降幂
- 不出意外的话，这一页又写不下了

证明

$$n^m = \sum_{i=0}^{m} \left\{ {m \atop i} \right\} n^{\underline{i}}$$

证明

$$
\begin{aligned}
n^m &= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} n^{\underline{i}} \\
&= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} (-1)^i (-n)^{\overline{i}}
\end{aligned}
$$

证明

$$
\begin{aligned}
n^m &= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} n^{\underline{i}} \\
&= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} (-1)^i (-n)^{\overline{i}} \\
&= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} (-1)^i \sum_{j=0}^{i} \left[{i \atop j} \right] (-n)^j
\end{aligned}
$$

证明

$$
\begin{aligned}
n^m &= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} n^{\underline{i}} \\
&= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} (-1)^i (-n)^{\overline{i}} \\
&= \sum_{i=0}^{m} \left\{ {m \atop i} \right\} (-1)^i \sum_{j=0}^{i} \left[{i \atop j} \right] (-n)^j \\
&= \sum_{j=0}^{m} n^j \sum_{i=j}^{m} \left\{ {m \atop i} \right\} \left[{i \atop j} \right] (-1)^{i-j}
\end{aligned}
$$

证明

$$
\begin{aligned}
n^m &= \sum_{i=0}^{m} \begin{Bmatrix} m \\ i \end{Bmatrix} n^{\underline{i}} \\
&= \sum_{i=0}^{m} \begin{Bmatrix} m \\ i \end{Bmatrix} (-1)^i (-n)^{\overline{i}} \\
&= \sum_{i=0}^{m} \begin{Bmatrix} m \\ i \end{Bmatrix} (-1)^i \sum_{j=0}^{i} \begin{bmatrix} i \\ j \end{bmatrix} (-n)^j \\
&= \sum_{j=0}^{m} n^j \sum_{i=j}^{m} \begin{Bmatrix} m \\ i \end{Bmatrix} \begin{bmatrix} i \\ j \end{bmatrix} (-1)^{i-j}
\end{aligned}
$$

- 第二个 $\sum$ 里面的项是与 n 无关的，但是 n 无论为何值的时候这个等式都成立，因此反转公式成立

斯特林反演

$$f(n)=\sum_{k=0}^{n}\left\{ {n \atop k} \right\} g(k) \Leftrightarrow g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left[{n \atop k} \right] f(k)$$

$$f(n)=\sum_{k=0}^{n}\left[{n \atop k} \right] g(k) \Leftrightarrow g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left\{ {n \atop k} \right\} f(k)$$

$$f(n)=\sum_{k\geq n}\left\{ {k \atop n} \right\} g(k) \Leftrightarrow g(n)=\sum_{k\geq n}(-1)^{k-n}\left[{k \atop n} \right] f(k)$$

斯特林反演

$$f(n)=\sum_{k=0}^{n}\left\{ {n \atop k} \right\} g(k) \Leftrightarrow g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left[{n \atop k} \right] f(k)$$

$$f(n)=\sum_{k=0}^{n}\left[{n \atop k} \right] g(k) \Leftrightarrow g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left\{ {n \atop k} \right\} f(k)$$

$$f(n)=\sum_{k\geq n}\left\{ {k \atop n} \right\} g(k) \Leftrightarrow g(n)=\sum_{k\geq n}(-1)^{k-n}\left[{k \atop n} \right] f(k)$$

- 我们接下来证明第一个，第二个的证明与之类似

斯特林反演

$$f(n)=\sum_{k=0}^{n}\left\{{n \atop k}\right\}g(k)\Leftrightarrow g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left[{n \atop k}\right]f(k)$$

$$f(n)=\sum_{k=0}^{n}\left[{n \atop k}\right]g(k)\Leftrightarrow g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left\{{n \atop k}\right\}f(k)$$

$$f(n)=\sum_{k\geq n}\left\{{k \atop n}\right\}g(k)\Leftrightarrow g(n)=\sum_{k\geq n}(-1)^{k-n}\left[{k \atop n}\right]f(k)$$

- 我们接下来证明第一个，第二个的证明与之类似
- 假设有

$$g(n)=\sum_{k=0}^{n}(-1)^{n-k}\left[{n \atop k}\right]f(k)$$

斯特林反演

$$f(n)=\sum_{i=0}^{n}[i=n]f(i)$$

斯特林反演

$$
\begin{aligned}
f(n) &= \sum_{i=0}^{n}[i=n]f(i) \\
&= \sum_{i=0}^{n}\sum_{j=i}^{n}\left\{ {n \atop j} \right\}\left[{j \atop i} \right](-1)^{j-i}f(i)
\end{aligned}
$$

斯特林反演

$$
\begin{aligned}
f(n) &= \sum_{i=0}^{n} [i = n] f(i) \\
&= \sum_{i=0}^{n} \sum_{j=i}^{n} \left\{ {n \atop j} \right\} \left[{j \atop i} \right] (-1)^{j-i} f(i) \\
&= \sum_{j=0}^{n} \left\{ {n \atop j} \right\} \sum_{i=0}^{j} \left[{j \atop i} \right] (-1)^{j-i} f(i)
\end{aligned}
$$

斯特林反演

$$
\begin{aligned}
f(n) &= \sum_{i=0}^{n} [i = n] f(i) \\
&= \sum_{i=0}^{n} \sum_{j=i}^{n} \left\{ {n \atop j} \right\} \left[{j \atop i} \right] (-1)^{j-i} f(i) \\
&= \sum_{j=0}^{n} \left\{ {n \atop j} \right\} \sum_{i=0}^{j} \left[{j \atop i} \right] (-1)^{j-i} f(i) \\
&= \sum_{j=0}^{n} \left\{ {n \atop j} \right\} g(j)
\end{aligned}
$$

再放送

CF1278F Cards

- 有 m 张牌，其中有一张是王牌，现在进行 n 次洗牌，每次洗牌之后查看第一张牌是什么
- 令 x 为洗牌后第一张牌为王牌的次数，求 x^k 的期望

$$n, m \leq 998244353, k \leq 5000$$

- 建议试试运用之前的公式爆推式子

CF1278F Cards

$$ans = \frac{1}{m^n} \sum_{i=1}^{n} \binom{n}{i} (m-1)^{n-i} i^k$$

CF1278F Cards

$$
\begin{aligned}
ans &= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}i^k \\
&= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}\sum_{j=0}^{k}\left\{ {k \atop j} \right\}\binom{i}{j}j!
\end{aligned}
$$

CF1278F Cards

$$
\begin{aligned}
ans &= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}i^k \\
&= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}\sum_{j=0}^{k}\begin{Bmatrix}k\\j\end{Bmatrix}\binom{i}{j}j! \\
&= \frac{1}{m^n}\sum_{j=0}^{k}\begin{Bmatrix}k\\j\end{Bmatrix}j!\sum_{i=1}^{n}\binom{n}{i}\binom{i}{j}(m-1)^{n-i}
\end{aligned}
$$

CF1278F Cards

$$\begin{aligned} ans &= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}i^k \\ &= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}\sum_{j=0}^{k}\left\{ {k \atop j} \right\}\binom{i}{j}j! \\ &= \frac{1}{m^n}\sum_{j=0}^{k}\left\{ {k \atop j} \right\}j!\sum_{i=1}^{n}\binom{n}{i}\binom{i}{j}(m-1)^{n-i} \\ &= \frac{1}{m^n}\sum_{j=0}^{k}\left\{ {k \atop j} \right\}j!\binom{n}{j}\sum_{i=1}^{n}\binom{n-j}{i-j}(m-1)^{n-i} \end{aligned}$$

CF1278F Cards

$$
\begin{aligned}
ans &= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}i^k \\
&= \frac{1}{m^n}\sum_{i=1}^{n}\binom{n}{i}(m-1)^{n-i}\sum_{j=0}^{k}\left\{{k \atop j}\right\}\binom{i}{j}j! \\
&= \frac{1}{m^n}\sum_{j=0}^{k}\left\{{k \atop j}\right\}j!\sum_{i=1}^{n}\binom{n}{i}\binom{i}{j}(m-1)^{n-i} \\
&= \frac{1}{m^n}\sum_{j=0}^{k}\left\{{k \atop j}\right\}j!\binom{n}{j}\sum_{i=1}^{n}\binom{n-j}{i-j}(m-1)^{n-i} \\
&= \frac{1}{m^n}\sum_{j=0}^{k}\left\{{k \atop j}\right\}j!\binom{n}{j}m^{n-j}
\end{aligned}
$$

一道例题

异或图

给定 S 个点集大小为 N 的图，问有多少个图的集合异或结果为一个连通图

$$N \leq 10, S \leq 60$$

异或图

- 设 $g(n)$ 表示将所有点恰好分为 n 个连通块的方案数

异或图

- 设 $g(n)$ 表示将所有点恰好分为 n 个连通块的方案数
- $f(n)$ 表示将所有点划分为 n 个集合，不同集合直接一定不连通，集合内部可以连通，也可以不连通的方案数

异或图

- 设 $g(n)$ 表示将所有点恰好分为 n 个连通块的方案数
- $f(n)$ 表示将所有点划分为 n 个集合，不同集合直接一定不连通，集合内部可以连通，也可以不连通的方案数
- 枚举 n 的集合划分，问题转化为有若干个异或方程，求解的个数，可以上线性基

异或图

- 设 $g(n)$ 表示将所有点恰好分为 n 个连通块的方案数
- $f(n)$ 表示将所有点划分为 n 个集合，不同集合直接一定不连通，集合内部可以连通，也可以不连通的方案数
- 枚举 n 的集合划分，问题转化为有若干个异或方程，求解的个数，可以上线性基

$$f(n) = \sum_{i \geq n} \left\{ {i \atop n} \right\} g(i)$$

$$g(n) = \sum_{i \geq n} \left[{i \atop n} \right] f(i)(-1)^{i-n}$$

$$g(1) = \sum_{i \geq 1} (-1)^{i-1}(n-1)! f(i)$$

另一道题

戴老师 ppt 里的

求 n 个点的带标号无向图的连通块数的 k 次方幂之和

$$n \leq 10^5, k \leq 15$$

另一道题

- n 个点带标号连通图的数量很好求，我们记它的 EGF 为 $\hat{G}(x)$

另一道题

- n 个点带标号连通图的数量很好求，我们记它的 EGF 为 $\hat{G}(x)$
- 设 n 个点带标号无向图的 EGF 为 $\hat{F}(x)$

另一道题

- n 个点带标号连通图的数量很好求，我们记它的 EGF 为 $\hat{G}(x)$
- 设 n 个点带标号无向图的 EGF 为 $\hat{F}(x)$

$$
\begin{aligned}
\hat{F}(x) &= \sum_{i \geq 0} \frac{\hat{G}(x)^i}{i!} \\
&= \exp \hat{G}(x) \\
\hat{G}(x) &= \ln \hat{F}(x)
\end{aligned}
$$

另一道题

- 先将 k 次方幂拆开

$$m^k = \sum_{i=0}^{k} \left\{ {k \atop i} \right\} \binom{m}{i} i!$$

另一道题

- 先将 k 次方幂拆开

$$m^k = \sum_{i=0}^{k} \begin{Bmatrix} k \\ i \end{Bmatrix} \binom{m}{i} i!$$

- 也就是说，如果一张图有 m 个连通块，我们需要计算出从这 m 个连通块中选出 i 个连通块的方案数之和

另一道题

- 先将 k 次方幂拆开

$$m^k = \sum_{i=0}^{k} \left\{ {k \atop i} \right\} \binom{m}{i} i!$$

- 也就是说，如果一张图有 m 个连通块，我们需要计算出从这 m 个连通块中选出 i 个连通块的方案数之和
- 记 $\hat{H}_k(x)$ 为无向图从所有连通块中选出 k 个连通块的方案数

另一道题

$$
\begin{aligned}
\hat{H}_k(x) &= \sum_{i\geq 0} \frac{\hat{G}(x)^i}{i!}\binom{i}{k} \\
&= \sum_{i\geq 0} \frac{\hat{G}(x)^i}{i!} \times \frac{i!}{k!(i-k)!} \\
&= \frac{1}{k!}\sum_{i\geq 0} \frac{\hat{G}(x)^i}{(i-k)!} \\
&= \frac{\hat{G}(x)^k}{k!}\sum_{i\geq 0} \frac{\hat{G}(x)^{i-k}}{(i-k)!} \\
&= \frac{\hat{G}(x)^k}{k!}\exp \hat{G}(x) = \frac{\hat{G}(x)^k}{k!}F(x)
\end{aligned}
$$

它 lei 了

清华集训 2017 生成树计数

- 在一个 s 个点的图中，存在 $s-n$ 条边，使图中形成了 n 个连通块，第 i 个连通块中有 a_i 个点
- 现在我们需要再连接 $n-1$ 条边，使该图变成一棵树。对于一种连边方案，设原图中第 i 个连通块连出了 d_i 条边，那么这棵树 T 的价值为

$$\mathrm{val}(x) = \left(\prod_{i=1}^{n} d_i^m\right)\left(\sum_{i=n}^{n} d_i^m\right)$$

- 求所有可能的生成树的价值之和

$$n \le 3 \times 10^4, m \le 30$$

- 在座的各位有谁能现场想出来我当场把 zzh 吃掉

清华集训 2017 生成树计数

- 将一开始就连通的点缩起来，我们考虑枚举缩点之后的一棵生成树 T

清华集训 2017 生成树计数

- 将一开始就连通的点缩起来，我们考虑枚举缩点之后的一棵生成树 T

$$ans = \sum_T \sum_{i=1}^{n} d_i^m \prod_{j=1}^{n} d_j^m a_j^{d_j}$$

清华集训 2017 生成树计数

- 将一开始就连通的点缩起来，我们考虑枚举缩点之后的一棵生成树 T

$$ans = \sum_{T} \sum_{i=1}^{n} d_i^m \prod_{j=1}^{n} d_j^m a_j^{d_j}$$

- 需要注意的是缩点之后每个连通块连出的边都可以选择这个连通块内的任意一个点作为起点，因此每条边都有 a_j 种选法。一个连通块连出了 d_j 条边，总方案数为 $a_j^{d_j}$

清华集训 2017 生成树计数

- 将一开始就连通的点缩起来，我们考虑枚举缩点之后的一棵生成树 T

$$ans = \sum_{T} \sum_{i=1}^{n} d_i^m \prod_{j=1}^{n} d_j^m a_j^{d_j}$$

- 需要注意的是缩点之后每个连通块连出的边都可以选择这个连通块内的任意一个点作为起点，因此每条边都有 a_j 种选法。一个连通块连出了 d_j 条边，总方案数为 $a_j^{d_j}$
- 将 $\prod$ 中 $i = j$ 的项提到前面来

$$ans = \sum_{T} \sum_{i=1}^{n} d_i^{2m} a_i^{d_i} \prod_{j \neq i} d_j^m a_j^{d_j}$$

清华集训 2017 生成树计数

- 枚举生成树的一种方法就是使用 prufer 序，考虑枚举每个点在 prufer 序中的出现次数 c_i，那么有 $d_i = c_i + 1$

清华集训 2017 生成树计数

- 枚举生成树的一种方法就是使用 prufer 序，考虑枚举每个点在 prufer 序中的出现次数 c_i，那么有 $d_i = c_i + 1$
- 对于一个长度为 $n-2$ 的 prufer 序，如果我们知道了每个点的出现次数，那么对应的序列就有 $\frac{(n-2)!}{\prod c_i!}$ 种

清华集训 2017 生成树计数

- 枚举生成树的一种方法就是使用 prufer 序，考虑枚举每个点在 prufer 序中的出现次数 c_i，那么有 $d_i = c_i + 1$
- 对于一个长度为 $n-2$ 的 prufer 序，如果我们知道了每个点的出现次数，那么对应的序列就有 $\frac{(n-2)!}{\prod c_i!}$ 种

$$ans = \sum_{\sum c_i = n-2} \frac{(n-2)!}{\prod c_i!} \sum_{i=1}^{n} (c_i+1)^{2m} a_i^{c_i+1} \prod_{j \neq i} (c_j+1)^m a_j^{c_j+1}$$

清华集训 2017 生成树计数

- 考虑根据这个式子构造 EGF

清华集训 2017 生成树计数

- 考虑根据这个式子构造 EGF
- 由于我们要满足 $\sum c_i = n-2$，不妨将 c_i 作为指数

清华集训 2017 生成树计数

- 考虑根据这个式子构造 EGF
- 由于我们要满足 $\sum c_i = n-2$，不妨将 c_i 作为指数

$$\hat{A}_i(x) = \sum_k a_i^{k+1}(k+1)^{2m}\frac{x^k}{k!}$$

$$\hat{B}_i(x) = \sum_k a_i^{k+1}(k+1)^{m}\frac{x^k}{k!}$$

清华集训 2017 生成树计数

- 考虑根据这个式子构造 EGF
- 由于我们要满足 $\sum c_i = n-2$，不妨将 c_i 作为指数

$$\hat{A}_i(x) = \sum_k a_i^{k+1}(k+1)^{2m}\frac{x^k}{k!}$$

$$\hat{B}_i(x) = \sum_k a_i^{k+1}(k+1)^{m}\frac{x^k}{k!}$$

- 单独考虑每个点对答案的贡献，可以得到

$$ans = (n-2)![x^{n-2}]\sum_{i-1}^{n} A_i(x)\prod_{j\neq i} B_j(x)$$

清华集训 2017 生成树计数

- 化简 $A_i(x), B_i(x)$：先积分，再求导

清华集训 2017 生成树计数

- 化简 $A_i(x), B_i(x)$：先积分，再求导

$$T_i(x) = \int \hat{A}_i(x)\,\mathrm{d}x = \sum_k a_i^{k+1} \frac{(k+1)^{2m}}{(k+1)!} x^{k+1}$$

清华集训 2017 生成树计数

- 化简 $A_i(x), B_i(x)$：先积分，再求导

$$T_i(x) = \int \hat{A}_i(x)\,\mathrm{d}x = \sum_k a_i^{k+1} \frac{(k+1)^{2m}}{(k+1)!} x^{k+1}$$

$$= \sum_k a_i^k \frac{k^{2m}}{k!} x^k = \sum_k a_i^k \sum_{j=0}^{2m} \left\{ {2m \atop j} \right\} \frac{k!}{(k-j)!} \times \frac{1}{k!} x^k$$

清华集训 2017 生成树计数

- 化简 $A_i(x), B_i(x)$：先积分，再求导

$$
\begin{aligned}
T_i(x) = \int \hat{A}_i(x)\,\mathrm{d}x &= \sum_k a_i^{k+1} \frac{(k+1)^{2m}}{(k+1)!} x^{k+1} \\
&= \sum_k a_i^k \frac{k^{2m}}{k!} x^k = \sum_k a_i^k \sum_{j=0}^{2m} \begin{Bmatrix} 2m \\ j \end{Bmatrix} \frac{k!}{(k-j)!} \times \frac{1}{k!} x^k \\
&= \sum_{j=0}^{2m} \begin{Bmatrix} 2m \\ j \end{Bmatrix} a_i^j x^j \sum_k \frac{(a_i x)^{k-j}}{(k-j)!}
\end{aligned}
$$

清华集训 2017 生成树计数

- 化简 $A_i(x), B_i(x)$：先积分，再求导

$$
\begin{aligned}
T_i(x) &= \int \hat{A}_i(x)\,\mathrm{d}x = \sum_k a_i^{k+1}\frac{(k+1)^{2m}}{(k+1)!}x^{k+1} \\
&= \sum_k a_i^k \frac{k^{2m}}{k!}x^k = \sum_k a_i^k \sum_{j=0}^{2m} \begin{Bmatrix} 2m \\ j \end{Bmatrix} \frac{k!}{(k-j)!} \times \frac{1}{k!}x^k \\
&= \sum_{j=0}^{2m} \begin{Bmatrix} 2m \\ j \end{Bmatrix} a_i^j x^j \sum_k \frac{(a_i x)^{k-j}}{(k-j)!} \\
&= \sum_{j=0}^{2m} \begin{Bmatrix} 2m \\ j \end{Bmatrix} a_i^j x^j \exp a_i x
\end{aligned}
$$

清华集训 2017 生成树计数

- 然后我们再求导回去

清华集训 2017 生成树计数

- 然后我们再求导回去

$$A_i(x) = T_i'(x) = \left[e^{a_i x} \left(\sum_{j=0}^{2m} \left\{ {2m \atop j} \right\} a_i^j x^j \right) \right]'$$

清华集训 2017 生成树计数

- 然后我们再求导回去

$$A_i(x)=T_i'(x)=\left[e^{a_ix}\left(\sum_{j=0}^{2m}\left\{{2m \atop j}\right\}a_i^jx^j\right)\right]'$$

$$=e^{a_ix}\sum_{j=0}^{2m}\left\{{2m \atop j+1}\right\}a_i^{j+1}(j+1)x^j+a_i\sum_{j=0}^{2m}\left\{{2m \atop j}\right\}a_i^jx^je^{a_ix}$$

清华集训 2017 生成树计数

- 然后我们再求导回去

$$
\begin{aligned}
A_i(x) = T_i'(x) &= \left[e^{a_i x} \left(\sum_{j=0}^{2m} \left\{ {2m \atop j} \right\} a_i^j x^j \right) \right]' \\
&= e^{a_i x} \sum_{j=0}^{2m} \left\{ {2m \atop j+1} \right\} a_i^{j+1} (j+1) x^j + a_i \sum_{j=0}^{2m} \left\{ {2m \atop j} \right\} a_i^j x^j e^{a_i x} \\
&= e^{a_i x} \sum_{j=0}^{2m} a_i^{j+1} x^j \left(\left\{ {2m \atop j+1} \right\} (j+1) + \left\{ {2m \atop j} \right\} \right)
\end{aligned}
$$

清华集训 2017 生成树计数

- 然后我们再求导回去

$$A_i(x)=T_i'(x)=\left[e^{a_ix}\left(\sum_{j=0}^{2m}\begin{Bmatrix}2m\\j\end{Bmatrix}a_i^jx^j\right)\right]'$$

$$=e^{a_ix}\sum_{j=0}^{2m}\begin{Bmatrix}2m\\j+1\end{Bmatrix}a_i^{j+1}(j+1)x^j+a_i\sum_{j=0}^{2m}\begin{Bmatrix}2m\\j\end{Bmatrix}a_i^jx^je^{a_ix}$$

$$=e^{a_ix}\sum_{j=0}^{2m}a_i^{j+1}x^j\left(\begin{Bmatrix}2m\\j+1\end{Bmatrix}(j+1)+\begin{Bmatrix}2m\\j\end{Bmatrix}\right)$$

$$=e^{a_ix}\sum_{j=0}^{2m}a_i^{j+1}x^j\begin{Bmatrix}2m+1\\j+1\end{Bmatrix}$$

清华集训 2017 生成树计数

- 同理，我们可以得出

$$B_i(x) = e^{a_i x} \sum_{j=0}^{m} a_i^{j+1} x^j \left\{ \begin{matrix} m+1 \\ j+1 \end{matrix} \right\}$$

清华集训 2017 生成树计数

- 同理，我们可以得出

$$B_i(x)=e^{a_ix}\sum_{j=0}^{m}a_i^{j+1}x^j\left\{{m+1 \atop j+1}\right\}$$

- 记 $A_i(x)=e^{a_ix}A_i^1(x), B_i(x)=e^{a_ix}B_i^1(x)$

清华集训 2017 生成树计数

- 同理，我们可以得出

$$B_i(x) = e^{a_i x} \sum_{j=0}^{m} a_i^{j+1} x^j \left\{ {m+1 \atop j+1} \right\}$$

- 记 $A_i(x) = e^{a_i x} A_i^1(x), B_i(x) = e^{a_i x} B_i^1(x)$

$$ans = (n-2)![x^{n-2}] \sum_{i-1}^{n} A_i(x) \prod_{j \neq i} B_j(x)$$

清华集训 2017 生成树计数

- 同理，我们可以得出

$$B_i(x) = e^{a_i x}\sum_{j=0}^{m} a_i^{j+1} x^j \left\{ {m+1 \atop j+1} \right\}$$

- 记 $A_i(x) = e^{a_i x}A_i^1(x), B_i(x) = e^{a_i x}B_i^1(x)$

$$ans = (n-2)![x^{n-2}]\sum_{i-1}^{n} A_i(x)\prod_{j\neq i} B_j(x)$$
$$= (n-2)![x^{n-2}]e^{\sum a_i x}\sum_{i=1}^{n} A_i^1(x)\prod_{j\neq i} B_j^1(x)$$

清华集训 2017 生成树计数

- 每个 $A_i^1(x)$ 都只有 $2m$ 项，每个 $B_i^1(x)$ 都只有 m 项

清华集训 2017 生成树计数

- 每个 $A_i^1(x)$ 都只有 $2m$ 项，每个 $B_i^1(x)$ 都只有 m 项
- 等式右边的那个东西可以用分治 NTT 求出来，维护区间的答案以及区间 $B_i^1(x)$ 的乘积即可

清华集训 2017 生成树计数

- 每个 $A_i^1(x)$ 都只有 $2m$ 项，每个 $B_i^1(x)$ 都只有 m 项
- 等式右边的那个东西可以用分治 NTT 求出来，维护区间的答案以及区间 $B_i^1(x)$ 的乘积即可
- 将 $e^{\sum a_i x}$ 展开，注意我们只需要计算 x^{n-2} 的系数

清华集训 2017 生成树计数

- 每个 $A_i^1(x)$ 都只有 $2m$ 项，每个 $B_i^1(x)$ 都只有 m 项
- 等式右边的那个东西可以用分治 NTT 求出来，维护区间的答案以及区间 $B_i^1(x)$ 的乘积即可
- 将 $e^{\sum a_i x}$ 展开，注意我们只需要计算 x^{n-2} 的系数
- 时间复杂度 $O(nm\log n)$

But

- 你会在洛谷 T 飞

But

- 你会在洛谷 T 飞
- 需要用到一些神仙优化

清华集训 2017 生成树计数

- 令

$$A(x)=\sum_{k}\frac{(k+1)^{2m}}{k!}x^k$$

$$B(x)=\sum_{k}\frac{(k+1)^{m}}{k!}x^k$$

清华集训 2017 生成树计数

- 令

$$A(x)=\sum_k \frac{(k+1)^{2m}}{k!}x^k$$
$$B(x)=\sum_k \frac{(k+1)^{m}}{k!}x^k$$

- 对比之前的式子，可以得到

$$ans=(n-2)!\prod a_i[x^{n-2}]\sum_{i=1}^{n} A(a_ix)\prod_{j\neq i} B(a_jx)$$

清华集训 2017 生成树计数

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n} A(a_i x) \prod_{j \neq i} B(a_j x) \\
&= \prod_{j=1}^{n} B(a_j x) \sum_{i=1}^{n} \frac{A(a_i x)}{B(a_i x)} \\
&= \exp\left(\sum_{j=1}^{n} \ln B(a_j x)\right) \sum_{i=1}^{n} \frac{A(a_i x)}{B(a_i x)}
\end{aligned}
$$

清华集训 2017 生成树计数

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n} A(a_i x) \prod_{j \neq i} B(a_j x) \\
&= \prod_{j=1}^{n} B(a_j x) \sum_{i=1}^{n} \frac{A(a_i x)}{B(a_i x)} \\
&= \exp\left(\sum_{j=1}^{n} \ln B(a_j x) \right) \sum_{i=1}^{n} \frac{A(a_i x)}{B(a_i x)}
\end{aligned}
$$

- 如果我们求出了 $\ln B(x), \frac{A(x)}{B(x)}$，那么只需要将 $\sum a_i^k$ 乘到这两个多项式 x^k 的系数上，就可以得出 $\sum \ln B(a_j x)$ 和 $\sum \frac{A(a_i x)}{B(a_i x)}$

清华集训 2017 生成树计数

- 对于任意的 $k \in [1, n]$，求出 $\sum a_i^k$

清华集训 2017 生成树计数

- 对于任意的 $k \in [1, n]$，求出 $\sum a_i^k$

$$G(x) = \sum_{k\geq 0}\sum_{i=1}^{n} a_i^k x^k$$
$$= \sum_{i=1}^{n}\sum_{k\geq 0}(a_i x)^k = \sum_{i=1}^{n}\frac{1}{1-a_i x}$$

清华集训 2017 生成树计数

- 对于任意的 $k \in [1, n]$，求出 $\sum a_i^k$

$$
\begin{aligned}
G(x) &= \sum_{k \geq 0} \sum_{i=1}^{n} a_i^k x^k \\
&= \sum_{i=1}^{n} \sum_{k \geq 0} (a_i x)^k = \sum_{i=1}^{n} \frac{1}{1 - a_i x} \\
&= \sum_{i=1}^{n} \left(1 + \frac{a_i x}{1 - a_i x}\right) = n - x \sum_{i=1}^{n} (\ln 1 - a_i x)'
\end{aligned}
$$

清华集训 2017 生成树计数

- 对于任意的 $k \in [1, n]$，求出 $\sum a_i^k$

$$
\begin{aligned}
G(x) &= \sum_{k \ge 0} \sum_{i=1}^{n} a_i^k x^k \\
&= \sum_{i=1}^{n} \sum_{k \ge 0} (a_i x)^k = \sum_{i=1}^{n} \frac{1}{1 - a_i x} \\
&= \sum_{i=1}^{n} \left(1 + \frac{a_i x}{1 - a_i x}\right) = n - x \sum_{i=1}^{n} (\ln 1 - a_i x)' \\
&= n - x \left[\ln \prod_{i=1}^{n} (1 - a_i x)\right]'
\end{aligned}
$$

单位根反演是啥

- 我们称主 n 次单位根 $\omega_n = \cos\frac{2\pi}{n} + \sin\frac{2\pi}{n} i$。在 FFT 的过程中带入的实际上就是 ω_n^0 到 ω_n^{n-1}

单位根反演是啥

- 我们称主 n 次单位根 $\omega_n = \cos\frac{2\pi}{n} + \sin\frac{2\pi}{n}i$。在 FFT 的过程中带入的实际上就是 ω_n^0 到 ω_n^{n-1}
- 如果有模数 p，那么我们可以求出 p 的原根 g，在模 p 意义下的主 n 次单位根就是 $g^{\frac{p-1}{n}}$

单位根反演是啥

- 我们称主 n 次单位根 $\omega_n = \cos\frac{2\pi}{n} + \sin\frac{2\pi}{n} i$。在 FFT 的过程中带入的实际上就是 ω_n^0 到 ω_n^{n-1}
- 如果有模数 p，那么我们可以求出 p 的原根 g，在模 p 意义下的主 n 次单位根就是 $g^{\frac{p-1}{n}}$
- 单位根反演：

$$[n \mid k] = \frac{1}{n}\sum_{i=0}^{n-1}(\omega_n^k)^i$$

单位根反演是啥

- 当 $n \mid k$ 时，$\omega_n^k = 1$，等式显然成立

单位根反演是啥

- 当 $n \mid k$ 时，$\omega_n^k = 1$，等式显然成立
- 否则，等式右边是一个等比数列，且公比不为 1

单位根反演是啥

- 当 $n \mid k$ 时，$\omega_n^k = 1$，等式显然成立
- 否则，等式右边是一个等比数列，且公比不为 1

$$\sum_{i=0}^{n-1} (\omega_n^k)^i = \frac{\omega_n^{kn} - 1}{\omega_n^k - 1} = 0$$

关于 Bluestein

- 任意长度 DFT

关于 Bluestein

- 任意长度 DFT
- 在 DFT 的过程中，我们想要求出原多项式在 ω_n^k 处的点值

$$y_k = \sum_{i=0}^{n-1} a_i \omega_n^{ki}$$

关于 Bluestein

- 任意长度 DFT
- 在 DFT 的过程中，我们想要求出原多项式在 ω_n^k 处的点值

$$y_k = \sum_{i=0}^{n-1} a_i \omega_n^{ki}$$

- ki 没法用卷积表示，可以考虑将其拆成 $\frac{k^2+i^2-(k-i)^2}{2}$

关于 Bluestein

- 任意长度 DFT
- 在 DFT 的过程中，我们想要求出原多项式在 ω_n^k 处的点值

$$y_k=\sum_{i=0}^{n-1}a_i\omega_n^{ki}$$

- ki 没法用卷积表示，可以考虑将其拆成 $\frac{k^2+i^2-(k-i)^2}{2}$

$$y_k=\sum_{i=0}^{n-1}a_i\omega_{2n}^{k^2+i^2-(k-i)^2}$$

关于 Bluestein

- 任意长度 DFT
- 在 DFT 的过程中，我们想要求出原多项式在 ω_n^k 处的点值

$$y_k=\sum_{i=0}^{n-1} a_i\omega_n^{ki}$$

- ki 没法用卷积表示，可以考虑将其拆成 $\frac{k^2+i^2-(k-i)^2}{2}$

$$\begin{aligned}y_k&=\sum_{i=0}^{n-1} a_i\omega_{2n}^{k^2+i^2-(k-i)^2}\\&=\omega_{2n}^{k^2}\sum_{i=0}^{n-1} a_i\omega_{2n}^{i^2}\times\omega_{2n}^{-(k-i)^2}\end{aligned}$$

一道例题

白兔之舞

- 有一张 n 行 $L+1$ 列的有向图，(x_1, y_1) 到 (x_2, y_2) 之间有 $w[x_1][x_2]$ 条边 $(x_1 < x_2)$
- 对于 $i \in [0, k)$，求出有多少条从 $(x, 0)$ 出发，终止节点所在的行为 y，且长度模 k 为 i 的路径
- 答案对质数 p 取模

$$n \le 3, L \le 10^8, 10^8 < p < 2^{32}, k \mid p-1, k < 65536$$

白兔之舞

- 考虑 $n=1$ 怎么做

白兔之舞

- 考虑 $n=1$ 怎么做
- 令 $v=w[1][1]$，那么我们实际上要求的是

$$y_t=\sum_{i=0}^{L} v^i\binom{L}{i}[k \mid i-t]$$

白兔之舞

- 考虑 $n=1$ 怎么做
- 令 $v=w[1][1]$，那么我们实际上要求的是

$$y_t=\sum_{i=0}^{L}v^i\binom{L}{i}[k\mid i-t]$$

- 代入反演公式

$$\begin{aligned}y_t&=\sum_{i=0}^{L}v^i\binom{L}{i}\frac{1}{k}\sum_{j=0}^{k-1}(\omega_k^{i-t})^j\\&=\frac{1}{k}\sum_{j=0}^{k-1}\omega_k^{-tj}\sum_{i=0}^{L}v^i\binom{L}{i}\omega_k^{ij}\end{aligned}$$

白兔之舞

- 最右边是一个长的很像二项式定理的东西

白兔之舞

- 最右边是一个长的很像二项式定理的东西

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1}\omega_k^{-tj}(1+v\omega_k^j)^L$$

白兔之舞

- 最右边是一个长的很像二项式定理的东西

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1} \omega_k^{-tj}(1 + v\omega_k^j)^L$$

- 又出现了熟悉的 $-tj$，但是这次我们不能将其拆成 $\frac{t^2+j^2-(t+j)^2}{2}$，因为 ω_{2k} 在模 p 意义下不一定存在

白兔之舞

- 最右边是一个长的很像二项式定理的东西

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1} \omega_k^{-tj}(1 + v\omega_k^j)^L$$

- 又出现了熟悉的 $-tj$，但是这次我们不能将其拆成 $\frac{t^2+j^2-(t+j)^2}{2}$，因为 ω_{2k} 在模 p 意义下不一定存在
- 拆成 $\binom{t}{2} + \binom{j}{2} - \binom{t+j}{2}$ 可以完美解决问题

白兔之舞

- 最右边是一个长的很像二项式定理的东西

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1}\omega_k^{-tj}(1+v\omega_k^j)^L$$

- 又出现了熟悉的 $-tj$，但是这次我们不能将其拆成 $\frac{t^2+j^2-(t+j)^2}{2}$，因为 ω_{2k} 在模 p 意义下不一定存在
- 拆成 $\binom{t}{2}+\binom{j}{2}-\binom{t+j}{2}$ 可以完美解决问题
- 考虑它的组合意义，tj 可以看作从 $t+j$ 个数的前 t 个数选一个，后 j 个数选一个的方案数，容斥一下就是从 $t+j$ 个数中选出两个数的方案数减去只在一个集合中选数的方案数

白兔之舞

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1}\omega_k^{\binom{t}{2}+\binom{j}{2}-\binom{t+j}{2}}(1+v\omega_k^j)^L$$

白兔之舞

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1} \omega_k^{\binom{t}{2}+\binom{j}{2}-\binom{t+j}{2}} (1 + v\omega_k^j)^L$$

- $(1 + v\omega_k^j)^L$ 与 k 无关，将其记作 a_j

白兔之舞

$$y_t = \frac{1}{k}\sum_{j=0}^{k-1}\omega_k^{\binom{t}{2}+\binom{j}{2}-\binom{t+j}{2}}(1+v\omega_k^j)^L$$

- $(1+v\omega_k^j)^L$ 与 k 无关，将其记作 a_j

$$y_t = \frac{\omega_k^{\binom{t}{2}}}{k}\sum_{j=0}^{k-1}\omega_k^{\binom{j}{2}}a_j \times \omega_k^{-\binom{t+j}{2}}$$

- MTT 即可

白兔之舞

- 将其拓展到 n 更大的情况

白兔之舞

- 将其拓展到 n 更大的情况
- 可以发现，边的转移从一个数变为了一个矩阵。我们记这个矩阵为 v，那么最初的式子就变为了

白兔之舞

- 将其拓展到 n 更大的情况
- 可以发现，边的转移从一个数变为了一个矩阵。我们记这个矩阵为 v，那么最初的式子就变为了

$$
\begin{aligned}
y_t &= \sum_{i=0}^{L} v_{(x,y)}^{i} \binom{L}{i} [k \mid i - t] \\
&= \frac{1}{k} \sum_{j=0}^{k-1} \omega_k^{-tj} \sum_{i=0}^{L} v_{(x,y)}^{i} \binom{L}{i} \omega_k^{ij} \\
&= \frac{1}{k} \sum_{j=0}^{k-1} \omega_k^{-tj} (I + v\omega_k^{j})_{(x,y)}^{L}
\end{aligned}
$$

- 其中 I 为单位矩阵

另一道题

复读机

- 求有多少个长度为 n 的序列，满足序列中的每一个元素均是 $[1,k]$ 中的数，且每个数出现了 d 的倍数次
- 对于 70% 的数据，$n \le 10^9, k \le 5 \times 10^5, d \le 2$
- 对于剩下的数据，$n \le 10^9, k \le 1000, d = 3$
- 答案对 19491001 取模

我会 $d = 1$!

- 输出 k^n 即可

$d = 2$

- 所有偶数的 EGF？

$d = 2$

- 所有偶数的 EGF？

$$f(x) = \frac{e^x + e^{-x}}{2}$$

$d = 2$

- 所有偶数的 EGF？

$$
\begin{aligned}
f(x) &= \frac{e^x + e^{-x}}{2} \\
f(x)^k &= \frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} e^{xi} e^{-x(k-i)} \\
&= \frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} e^{x(2i-k)}
\end{aligned}
$$

$d = 2$

- 所有偶数的 EGF？

$$
\begin{aligned}
f(x) &= \frac{e^x + e^{-x}}{2} \\
f(x)^k &= \frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} e^{xi} e^{-x(k-i)} \\
&= \frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} e^{x(2i-k)} \\
f(x)^k[x^n] &= \frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} \frac{(2i-k)^n}{n!}
\end{aligned}
$$

$d = 3$

- 写出一个数出现次数的 EGF

$d = 3$

- 写出一个数出现次数的 EGF

$$f(x) = \sum_{i \geq 0} \frac{x^i}{i!} [3|i]$$

$d=3$

- 写出一个数出现次数的 EGF

$$
\begin{aligned}
f(x) &= \sum_{i\geq 0}\frac{x^i}{i!}[3|i] \\
&= \sum_{i\geq 0}\frac{x^i}{i!}\times\frac{1+\omega_3^i+\omega_3^{2i}}{3} \\
&= \frac{e^x+e^{x\omega_3^1}+e^{x\omega_3^2}}{3}
\end{aligned}
$$

$d = 3$

- 写出一个数出现次数的 EGF

$$
\begin{aligned}
f(x) &= \sum_{i\geq 0} \frac{x^i}{i!}[3|i] \\
&= \sum_{i\geq 0} \frac{x^i}{i!} \times \frac{1+\omega_3^i+\omega_3^{2i}}{3} \\
&= \frac{e^x + e^{x\omega_3^1} + e^{x\omega_3^2}}{3}
\end{aligned}
$$

- 最后要求的是 $f(x)^k[x^n]$

$d = 3$

- 写出一个数出现次数的 EGF

$$
\begin{aligned}
f(x) &= \sum_{i\geq 0} \frac{x^i}{i!}[3|i] \\
&= \sum_{i\geq 0} \frac{x^i}{i!} \times \frac{1+\omega_3^i+\omega_3^{2i}}{3} \\
&= \frac{e^x + e^{x\omega_3^1} + e^{x\omega_3^2}}{3}
\end{aligned}
$$

- 最后要求的是 $f(x)^k[x^n]$
- 可以理解为每次从 3 项中选出一项，一共选 k 次，枚举前两项被选的次数即可

$d = 3$

- 具体来说，假设第一项被选了 i 次，第二项被选了 j 次

$d = 3$

- 具体来说，假设第一项被选了 i 次，第二项被选了 j 次
- 贡献为 $\binom{k}{i} \times \binom{k-i}{j} \times e^{xi} \times e^{x^j \omega_3^j} \times e^{x^{k-i-j} \omega_3^{2(k-i-j)}}$

$d = 3$

- 具体来说，假设第一项被选了 i 次，第二项被选了 j 次
- 贡献为 $\binom{k}{i} \times \binom{k-i}{j} \times e^{xi} \times e^{x^j \omega_3^j} \times e^{x^{k-i-j} \omega_3^{2(k-i-j)}}$
- 19491001 的原根是 7

$d = 3$

- 具体来说，假设第一项被选了 i 次，第二项被选了 j 次
- 贡献为 $\binom{k}{i} \times \binom{k-i}{j} \times e^{xi} \times e^{xj\omega_3^j} \times e^{x^{k-i-j}\omega_3^{2(k-i-j)}}$
- 19491001 的原根是 7

```
int res = 0, w1 = Pow(7, (mod - 1) / 3), w2 = (LL)w1 * w1 % mod;
for (int i = 0; i <= k; i++)
    for (int j = 0; i + j <= k; j++) {
        int t = k - i - j, w = (LL)C(k, t) * C(k - t, i) % mod;
        int v = ((LL)i * w1 + (LL)j * w2 + t) % mod;
        res = (res + (LL)w * Pow(v, n)) % mod;
    }
print((LL)res * Pow(3, mod - 1 - k) % mod);
```