母函数与DP的综合运用

金恺

计数问题

- Bijective proof
- Double counting
- 容斥 及各种反演
- DP
- 母函数

例1【PolandBall and Many Other Balls】CF-755G
例2【Timber】CF-1821F
例3【Count】uoj-424 （集训队作业2018）
例4【Unlucky string】lightoj unlucky-strings
--------休息--------
例5【Descents of Permutations】gym-280379B
例6【Same Descent Set】 AGC060d
--------练习--------
例7【数的分解问题】无题号

例1【PolandBall and Many Other Balls】

CF-755G

题意

- n个球排成一行。需要选出m组球，每组1个或2个球；每个球只能属于一组。方案个数记作$f_{n,m}$。
- 求$f_{n,1},\dots,f_{n,k}$。（对 998244353取模）。N ≤ 1e9,k<2^{15}。

DP解法

$$f_{n,m}=f_{n-1,m}+f_{n-1,m-1}+f_{n-2,m-1}\ (n\geq 2)$$

优化

记$F_n(x)=f_{n,0}+f_{n,1}x+f_{n,2}x^2+\cdots+f_{n,k}x^k$。

$F_n(x)=F_{n-1}(x)+F_{n-1}(x)x+F_{n-2}(x)x$。

考虑倍增。 $F_n(x) \to F_{2n}(x)$?

$F_n(x) = f_{n,0} + f_{n,1}x + \cdots + f_{n,k}x^k$。

$f_{n+n,m} = \sum_{i=0}^{m} f_{n,i} f_{n,m-i} + \sum_{i=0}^{m-1} f_{n-1,i} f_{n-1,m-i-1}$

这意味着

$$F_{2n}(x) = F_n(x)F_n(x) + F_{n-1}(x)F_{n-1}(x)x$$

类似的,

$$F_{2n-1}(x) = F_{n-1}(x)F_n(x) + F_{n-2}(x)F_{n-1}(x)x$$
$$F_{2n-2}(x) = F_{n-1}(x)F_{n-1}(x) + F_{n-2}(x)F_{n-2}(x)x$$

因此: $F_n(x), F_{n-1}(x), F_{n-2}(x) \to F_{2n}(x), F_{2n-1}(x), F_{2n-2}(x)$.

注意, 很难用$F_n(x) \to F_{2n}(x)$. 因为$F_{2n}(x)$还需要$F_{n-1}(x)$。
也很难用$F_n(x), F_{n-1}(x) \to F_{2n}(x), F_{2n-1}(x)$。因为$F_{2n-1}(x)$倚赖$F_{n-2}(x)$

具体细节

- 令$G_i = \left(F_i(x), F_{i-1}(x), F_{i-2}(x)\right)$。
- $G_i \rightarrow G_{i+1}$， 也可$G_i \rightarrow G_{2i}$。
- 因此G_n 可以通过至多$\log(n)$次→求出。
- 每一次倍增，需要计算多项式乘法即可。复杂度为$O(k\log k)$。

$$F_{2n}(x) = F_n(x)F_n(x) + F_{n-1}(x)F_{n-1}(x)x$$
$$F_{2n-1}(x) = F_{n-1}(x)F_n(x) + F_{n-2}(x)F_{n-1}(x)x$$
$$F_{2n-2}(x) = F_{n-1}(x)F_{n-1}(x) + F_{n-2}(x)F_{n-2}(x)x$$

最终复杂度为 $O(k\log k\log n)$。

例2【Timber】CF-1821F

题意:

- 从坐标1~n中选m个位置种树，有多少选法“合法”(定义如下)?

- 假定所有树有相同高度k。 (n,m,k<=3e5）。
- 若往左推倒某棵位于x的树，则会**占据**位置[x-k,x] （要求$x-k \ge 1$)
- 若往右推倒某棵位于x的树，则会**占据**位置[x,x+k] （要求$x+k \le n$)
- 合法⇔ 存在一种推倒策略，使所有树占据的位置无重叠。

举例 n=4,m=2,K=1。
那么 {1,2} 不合法、{3,4}不合法。其他合法。

如何判定某方案合法?

- 贪心即可:
  - 从左往右枚举树；能往左推到就往左；否则往右推。

DP解法（思考 1min+1min)

- $F[i][j]$ 表示 前面j个树，有多少种选法满足如下条件:
  - 按贪心算法倒下来后 占据的最大坐标为i。
- $F[i][j] = \sum_{i'=0}^{i-2k-1} F[i'][j-1] + 2\sum_{i'=i-2k}^{i-k-1} F[i'][j-1]$

O(nm)

$$F[i][j]=\sum_{i'=0}^{i-2k-1}F[i'][j-1]+2\sum_{i'=i-2k}^{i-k-1}F[i'][j-1]$$

- 令$F_j(x)$ 为 $F[0][j],F[1][j],\ldots,F[n][j]$ 的母函数。
- $F_j(x)=F_{j-1}(x)\sum_{t=2k+1}^{n}x^t+2F_{j-1}(x)\sum_{t=k+1}^{2k}x^t$

因此$F_j(x)=\left(\sum_{t=2k+1}^{n}x^t+2\sum_{t=k+1}^{2k}x^t\right)F_{j-1}(x)$。

令 $\mathrm{A}(x)=\sum_{t=2k+1}^{n}x^t+2\sum_{t=k+1}^{2k}x^t$。

那么 $F_m(x)=A(x)^mF_0(x)=A(x)^m$.

A(x)易求。利用多项式快速幂可求出$F_m(x)$ 。 复杂度$O(n\log n)$。

改进算法

小Trick

$$A^*(x)=\sum_{t=2k+1}^{\infty}x^t+2\sum_{t=k+1}^{2k}x^t=x^{k+1}\left(\sum_{t=k}^{\infty}x^t+2\sum_{t=0}^{k-1}x^t\right)$$

$$=x^{k+1}\left(\sum_{t=0}^{\infty}x^t+\sum_{t=0}^{k-1}x^t\right)=x^{k+1}\left(\frac{1}{1-x}+\frac{1-x^k}{1-x}\right)=x^{k+1}\frac{2-x^k}{1-x}$$

- $Ans=\ [x^n]A^*(x)^m=\left[x^{n-m(k+1)}\right]\left(\frac{2-x^k}{1-x}\right)^m$
- 令$N=n-m(k+1)$。 要求 $[x^N](2-x^k)^m(1-x)^{-m}$

A(x)B(x)的某一项的系数，不需要去FFT。

- $(2-x^k)^m$ 的某项的系数，可用二项式定理 求得。
- $(1-x)^{-m}$的第n项的系数，可用牛顿二项式定理求得$\binom{m+n-1}{m-1}$。

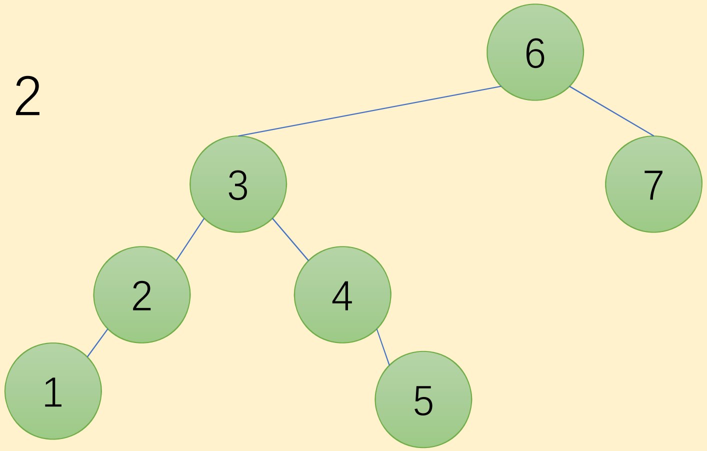
例3【Count】uoj-424　（集训队作业2018）

题意:

- 一个长为n的序列称作挺好 如果
  - 由1~m组成，且1~m都出现过至少一次。
- 对于序列A，定义$f_A(l,r)$为A_l~A_r的最大值的下标
  - 若有多个最大值，选下标最小的。
- 两个序列A和B同构 ⇔ 所有的区间l,r都满足$f_A(l,r) = f_B(l,r)$。
- 给定n,m <=1e5。问：共有多少种 不同构的 挺好序列?

理解“同构”

- 每个序列A都可以定义一棵二叉树T_A（笛卡尔树）
- 举例
  - 1 2 3 3 2 4 2

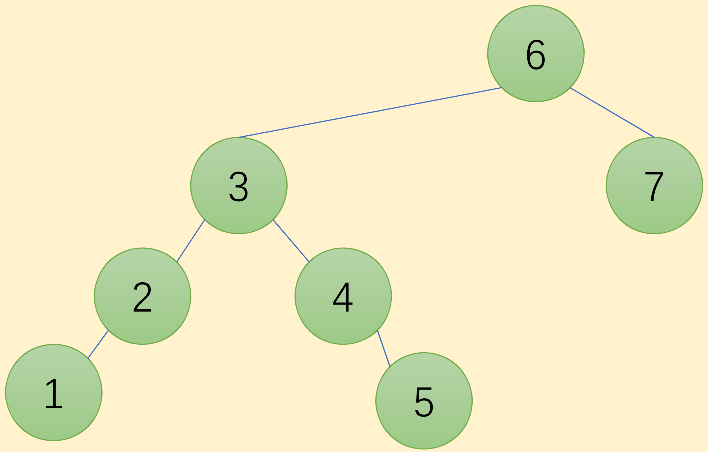


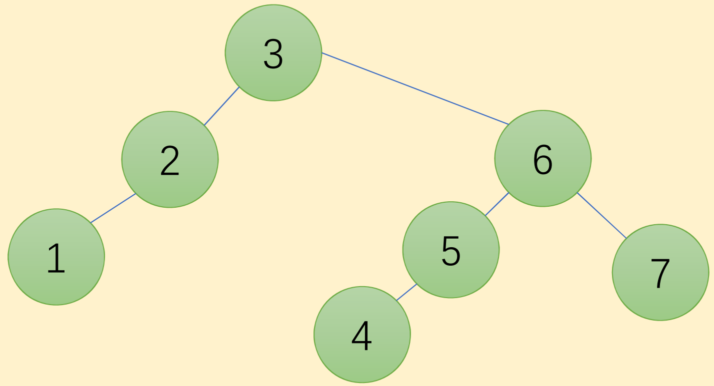
- 两个序列A,B同构 $\Leftrightarrow$ $T_A=T_B$。
- 问：<u>由多少种树T，存在某个挺好的序列A使得T_A=T。</u>

什么样的树，存在对应它的挺好序列?

- **观察**：根必须填m。根的左边必须<m。往左走时须用更小的数字去填。

m=3
无解





m=3
有解

- 定义一个树的**左拐度**=“往下走时，最多可左拐几次”
- **结论**：左拐度≥m时，无对应的挺好序列。（显然）
- **结论**：左拐度<m时，存在对应的挺好序列。（不难证明，课间思考）

左拐度<m的二叉树计数

（且前序遍历为1~n）

- $F_{n,m}$ 表示 n个节点的二叉树有多少满足 左拐度<m 。
- $\begin{cases} F_{n,0} = [n = 0] \\ F_{n,1} = 1 \\ F_{n,m} = \sum_{i=0}^{n-1} F_{i,m-1} \cdot F_{n-1-i,m} \end{cases}$
- 暴力转移超时。
- 设$F_m(x)$为$F_{?,m}$ 的母函数。易得 $F_m(x) = xF_{m-1}(x)F_m(x)$。
- 因此，$F_m(x) = \frac{1}{1-xF_{m-1}(x)}$

如何计算$F_m(x) = \dfrac{1}{1-xF_{m-1}(x)}$？

（且前序遍历为1~n）

- 注意$F_0(x) = 1$. $F_1(x) = \frac{1}{1-x}$. $F_m(x)$为有理式。
- 假定$F_m(x) = \frac{A_m(x)}{B_m(x)}$。
- 可得$F_{m+1}(x) = \frac{1}{1-x\frac{A_m(x)}{B_m(x)}} = \frac{B_m(x)}{B_m(x)-xA_m(x)}$。 分子为之前的分母。

- 假定$F_m(x) = \frac{B_{m-1}(x)}{B_m(x)}$
- 可得： $F_{m+1}(x) = \frac{1}{1-x\frac{B_{m-1}(x)}{B_m(x)}} = \frac{B_m(x)}{B_m(x)-xB_{m-1}(x)}$。
- 因此$B_{m+1}(x) = B_m(x) - xB_{m-1}(x)$.

如何计算$B_m(x) = B_{m-1}(x) - xB_{m-2}(x)$?

- 倍增思路。
- 写成$\begin{bmatrix} B_m \\ B_{m-1} \end{bmatrix} = \begin{bmatrix} 1 & -x \\ 1 & 0 \end{bmatrix} \begin{bmatrix} B_{m-1} \\ B_{m-2} \end{bmatrix}$。那么转换为求$\begin{bmatrix} 1 & -x \\ 1 & 0 \end{bmatrix}^m$。
- 可以用矩阵快速幂的方法。复杂度: $O(\log m * m \log m)$。

- 更好的方法
  - 观察：$B_m(x)$为m次多项式。可以结合拉格朗日插值。
  - 先要求出$B_m(x)$在 m个点的值——O(m log m)。
  - 然后再插值得到$B_m(x)$的系数。(取值需要特殊，细节略)

例4【Unlucky string】

lightoj.com/problem/unlucky-strings

题意

- 字符集大小为m。给一个字符串$s=s_1 \ldots s_k$。
- 求： 长为n的串中不含子串s的有几个？n,m,k<=1e5（非原题）

DP解法

- f_i表示有多少方法填入i个字符且s第一次出现在[i-k+1,i]。
- 令$D = \{d \geq 1 \mid s[d+1,k] = s[1,k-d]\}$
- i>=k时,

$$f_i = m^{i-k} - \sum_{0 \leq j \leq i-k} f_j m^{i-k-j} - \sum_{d \in D} f_{i-d}$$

计算$f_i = m^{i-k} - \sum_{0\le j\le i-k} f_j m^{i-k-j} - \sum_{d\in D} f_{i-d}$？

- 令$f(x) = \sum f_i x^i$。令$d(x) = \sum_{d\in D} x^d$。
- $f(x) = \frac{x^k}{1-mx} - \frac{x^k}{1-mx} f(x) - f(x)d(x)$。
- 因此$f(x)\left(1 + d(x) + \frac{x^k}{1-mx}\right) = \frac{x^k}{1-mx}$
- 因此$f(x) = \frac{x^k}{(1+d(x))(1-mx)+x^k}$。
- 答案即为：$g_n = m^n - \sum_{i\ge 0} f_i m^{i-j}$。

例5【Descents of Permutations】

gym-280379B

题意:

- 对[n]的排列$p=(p_1,\ldots,p_n)$，定义它的“descent集”为 $\{i \mid p_i>p_{i+1}\}$。
  - 举例：(1,4,2,5,3)的 descent集 为{2,4}。
- 给定n,k ≤ 5e5 。求长为n的排列中有多少个满足descent集为

  $\{k,2k,3k,\ldots\} \cap [n-1]$

  (即k的倍数处下降，其他地方上升)

- 固定[n-1]的任何一个子集S。
- 令f(S)表示有多少个排列的 descent集**恰好等于**为S。
- 目标：计算f(K)，其中 K={1k,2k,3k,…} ∩ [n-1] 。

- 令g(S)表示有多少个排列的descent集属于S。g(S)更容易计算。

马上将看到

- 另外，很显然f和g可以子集反演:

$$g(S) = \sum_{T\subseteq S} f(T) \Rightarrow f(S) = \sum_{T\subseteq S} (-1)^{|T|-|S|} g(T)$$

g(S)的计算

g(S): 有多少个排列，它的descent集属于S

- 假设$S=\{s_1,\ldots,s_m\}$，其中$s_1<\ldots<s_m$。
- g(S) = 下述排列的个数：

  在$s_1,\ldots,s_m$这些地方 可降可升；其他位置必须升。

- 将$p_1\sim p_n$分为m+1段： $p[s_{i-1}+1]\sim p[s_i]$为一段。每段上升.
- 易知，$g(S)=\frac{n!}{l_1!\ldots l_{m+1}!}$，其中$l_1,\ldots l_{m+1}$表示各段的长度。

  具体来说，$l_1=s_1, l_2=s_2-s_1,\ldots,l_{m+1}=n-s_m$。

回到f(K)的计算。

- 不妨设 $n=pk+r$。其中$1\leq r\leq k$。
- $K=\{1k,2k,3k,\ldots\}\cap[n-1]=\{1k,2k,\ldots pk\}$。
- $f(K)=\sum_{T\subseteq K}(-1)^{|T|-p}g(T)$。
- 考虑K的每个子集T，都可以一一对应到一个正整数序列$(b_1,\cdots,b_{m+1})$，其中$b_1,\cdots,b_m$为k的倍数，且$\Sigma bi=n$。（称good序列）
- $f(K)=\sum_{(b_1,\ldots,b_{m+1})good}(-1)^{m-p}\frac{n!}{b_1!\ldots b_{m+1}!}$

如何计算此式子？

$$f(K) = \sum_{(b_1,\ldots,b_{m+1})good} (-1)^{p-m} \frac{n!}{b_1! \ldots b_{m+1}!}$$

继续简化

- $(b_1 \sim b_{m+1})$ Good ⇔ 和为n且均为正数且$b_1 \sim b_m$为k的倍数。
- 做一个映射:
  - $c_1=(b_1-k)/k,\cdots,c_m=(b_m-k)/k$, $c_{m+1}=(b_{m+1}-r)/k$。
  - $c_1 \sim c_{m+1}$非负 且和为 p-m.

$$f(K) = n! \sum_{\substack{(c_1,\ldots,c_{m+1}):\\ \text{和为}p-m}} \frac{(-1)^{c_1+\cdots+c_{m+1}}}{(kc_1+k)!\ldots(kc_m+k)!\,(kc_{m+1}+r)!}$$

$$\sum_{(c_1,\dots,c_{m+1}):\text{和为}\,p-m} \frac{(-1)^{c_1+\cdots+c_{m+1}}}{(kc_1+k)!\dots(kc_m+k)!\,(kc_{m+1}+r)!} = ?$$

- 令$A(x)=\sum_{i\geq 0}\frac{(-1)^i}{(ki+k)!}x^i$，令$B(x)=\sum_{i\geq 0}\frac{(-1)^i}{(ki+r)!}x^i$
- 当固定m $(0\leq m\leq p)$时，要计算： $[x^{p-m}]\,(A(x)^m B(x))$
  - 即，要计算$[x^p]\,(x^m A(x)^m B(x))$ 。（小技巧）
- 由于m可以随意，最终要计算 $[x^p]\left(\sum_{m\geq 0} x^m A(x)^m B(x)\right)$
- $\sum_{m\geq 0}(xA(x))^m B(x)=\frac{B(x)}{1-xA(x)}$。 可以求出B(x),A(x)，然后多项式逆。

总结：子集反演（可视作DP)+母函数。

例6【Same Descent Set】 AGC060d

题意

- 给定N<=2e5，问：有多少对(P,Q)= $(P_1,\cdots P_N),(Q_1,\cdots,Q_N)$满足
  - P和Q都是1~N的排列 （可以相等）
  - 且P与Q的descent集 相等。
- 答案对998244353取模。

- f(S): Descent集恰好为S的 排列的数量。
- 目标：求 $\Sigma_S f(S)^2$。

$g(S)$：Descent集属于S的排列的数量。

$f(S) = \sum_{T\subseteq S}(-1)^{|T|-|S|}g(T)$。

求 $\sum_S f(S)^2$

$$= \sum_S\left(\sum_{T\subseteq S}(-1)^{|T|-|S|}g(T)\right)^2$$

$$= \sum_S\left(\sum_{T_1\subseteq S}(-1)^{|T_1|-|S|}g(T_1)\right)\left(\sum_{T_2\subseteq S}(-1)^{|T_2|-|S|}g(T_2)\right)$$

$$= \sum_S\sum_{T_1\subseteq S}\sum_{T_2\subseteq S}(-1)^{|T_1|+|T_2|}g(T_1)g(T_2)$$

$$= \sum_{T_1}\sum_{T_2}(-1)^{|T_1|+|T_2|}g(T_1)g(T_2)2^{n-1-|T_1\cup T_2|}$$

$$= \sum_{T_1}\sum_{T_2}(-1)^{|T_1|+|T_2|}g(T_1)g(T_2)2^{n-1-|T_1|-|T_2|+|\mathrm{T}_1\cap T_2|}$$

$$= 2^{n+1}\sum_{T_1}\sum_{T_2}(-1)^{|T_1|+|T_2|}g(T_1)g(T_2)2^{-|T_1|-1-|T_2|-1+|T_1\cap T_2|}$$

计算$\sum_{T_1}\sum_{T_2}(-1)^{|T_1|+|T_2|}g(T_1)g(T_2)2^{-|T_1|-1-|T_2|-1+|T_1\cap T_2|}$

- $=\sum_{T_1}\sum_{T_2}\left(-\frac{1}{2}\right)^{|T_1|+1}g(T_1)\left(-\frac{1}{2}\right)^{|T_2|+1}g(T_2)2^{|T_1\cap T_2|}$
- $=\sum_{T_1}\sum_{T_2}\left(-\frac{1}{2}\right)^{|T_1|+1}g(T_1)\left(-\frac{1}{2}\right)^{|T_2|+1}g(T_2)\sum_{S\subseteq T_1,S\subseteq T_2}1$
- $=\sum_{S}\sum_{T_1\supseteq S}\left(-\frac{1}{2}\right)^{|T_1|+1}g(T_1)\sum_{T_2\supseteq S}\left(-\frac{1}{2}\right)^{|T_2|+1}g(T_2)$
- $=\sum_{S}\left(\sum_{T\supseteq S}\left(-\frac{1}{2}\right)^{|T|+1}g(T)\right)^2$
- $=\sum_{b:\Sigma b_i=n}\left(\sum_{c_1\sim c_M:是b的细化}\left(-\frac{1}{2}\right)^{M}\frac{n!}{c_1!\dots c_M!}\right)^2$
- $=(n!)^2\sum_{b:\Sigma b_i=n}\left(\sum_{c_1\sim c_M:是b的细化}\frac{-1}{2c_1!}\cdots\frac{-1}{2c_M!}\right)^2$

如何计算 $\sum_{b:\Sigma b_i=n}\left(\sum_{c_1\sim c_M:是b的细化}\frac{-1}{2c_1!}\cdots\frac{-1}{2c_M!}\right)^2$

- **注意**：要将b=(b_1~b_m)细化，是将每一段分别细化。每段独立！
- 记 $f_{len}=\sum_{c_1\sim c_k是len的拆分}\frac{-1}{2c_1!}\cdots\frac{-1}{2c_M!}$.
- 上式=$\sum_{b:\Sigma b_i=n}\left(f_{b_1}\cdots f_{b_m}\right)^2=\sum_{b:\Sigma b_i=n}f_{b_1}^2\cdots f_{b_m}^2$。可借助gf解决。

- 转化为求 f_{len}。
- 同样容易借助gf解决。令 $F(x)=\sum_{i\geq 1}\frac{-1}{2i!}x^i$
- 那么 $f_{len}=[x^{len}](F(x)+F(x)^2+F(x)^3+\cdots)=[x^{len}]\frac{1}{1-F(x)}$

本题的关键步骤回顾

- $\sum_S f(S)^2$
- $= \sum_S \left(\sum_{T\subseteq S}(-1)^{|T|-|S|} g(T)\right)^2$ 子集反演
- $= 2^{n+1} \sum_{T_1}\sum_{T_2}(-1)^{|T_1|+|T_2|} g(T_1)g(T_2)2^{-|T_1|-1-|T_2|-1+|T_1\cap T_2|}$ 并换交
- $=2^{n+1}\sum_S \left(\sum_{T\supseteq S}\left(-\frac{1}{2}\right)^{|T|+1} g(T)\right)^2$ 交换求和
- $=2^{n+1}(n!)^2\sum_{b:\Sigma b_i=n}\left(\sum_{c_1\sim c_M:\text{是}b\text{的细化}}\frac{-1}{2c_1!}\cdots\frac{-1}{2c_M!}\right)^2$ S和T看成拆分
- $=2^{n+1}(n!)^2\sum_{b:\Sigma b_i=n}\left(f_{b_1}\cdots f_{b_m}\right)^2$ 每段的独立性

例7 数的分解问题

题意

- 给定n≤10000， k≤30。
- 将 n! 分解成 k个正整数的乘积$b_1*...*b_k$。
- 有多少种分解方式满足：$b_1,\cdots,b_k$**各不相同**。

定义

- 对于一个给定的分解$(b_1,\cdots,b_k)$， 可定义出{1,..k}的一个划分：
  - 当且仅当$b_i=b_j$时，将下标i与j划分到同一组。 （记作Π_b）
  - 举例， b={1,2,2,1,2} 对应的划分为 Π_b={1,4} {2,3,5}。
- 问题等价于：有多少个分解$(b_1,\cdots,b_k)$满足 $\Pi_b=\{1\}\{2\}\ldots\{k\}$。

有多少分解($b_1,\cdots,b_k$)满足 $\Pi_b= \{1\}\ldots\{k\}$?

i,j同组⇔ $b_i=b_j$

概念（细分 refine）

- 记$[k]=\{1,\ldots,k\}$。考虑$[k]$的两个分解 Π 与 Π'。
- 我们说 Π是Π'的一个细分(refine)，记作$\Pi\leq\Pi'$，如果
  - (i,j)在Π中同组→ (i,j)在Π'的同组。
- 举例：$\Pi'=\{13\}\{2456\}$的一个细分 为$\Pi= \{1\}\{3\}\{25\}\{4\}\{6\}$。
  - （细分即，把每一个组拆成若干个组）

- 令$f(\Pi)$表示：由多少分解b满足$\Pi_b=\Pi$。
- 令$g(\Pi)$表示：由多少分解b满足$\Pi_b\geq\Pi$ 。

思路：目标 $f(\{1\}\ldots\{k\})$。
g比较好求；f不太好求。
不妨容斥从g推出f（how?)

计算g(Π)——有多少分解b满足$\Pi_b \geq \Pi$

- 举例: Π= {1}{2,5}{3,4,6}.
- g(Π)= size of { b | Π_b 的一个细分为 Π } = size of { b | $b_2=b_5$，$b_3=b_4=b_6$}。

- 满足$b_2=b_5$ 且 $b_3=b_4=b_6$ 的 分解方案b 的个数很容易计算。

- 1.设N!= $p_1^{a_1} \ldots p_t^{a_t}$。 将N!分解 等价于 依次将每个$p_i^{a_i}$ 做分解。
- 2.将$p_i^{a_i}$分解为 6份，要求第2，5份相同，第3，4，6份相同
  转化为 将a_i 拆分成 $x_1,\ldots,x_6$的和，要求$x_2=x_5$，且$x_3=x_4=x_6$。（x非负）
  - 这是一个典型的背包问题；可以用DP进行求解！ (细节略)

如何通过容斥求$f(\{1\}\{2\}\ldots\{k\})$?

- 先转化一下问题（合并类型相同的划分）
- **类型的定义**
  - {1}{2,3,4}{5} 与 {2}{3}{1,4,5} 都是 113类型的。
  - {1}{2}{3}{4}{5} 的类型是 11111. {12345}的类型是5.
- **同类合并**:
  - 两个划分的“**类型**相同”，则 f 值相同，g值相同。
  - $G_T := \sum_{\Pi:\text{类型为}T} g_\Pi$. $F_T := \sum_{\Pi:\text{类型为}T} f_\Pi$. （G容易求；$F_{11..1}$待求）
  - 类型的个数，记作m，不超过4565。($k \le 30$的拆分数)

小结: Π= {1}{2,5}{3,4,6}（划分）。123（类型），细分{1}{2}{5}{3}{46}。

如何从G计算 $F_{11..1}$? (举例k=4)

- **任务**：从$G_{1111}, G_{112}, G_{13}, G_{22}, G_4$ 求 F_{1111} 。
- **观察**：

$$\begin{bmatrix} G_{1111} \\ G_{112} \\ G_{13} \\ G_{22} \\ G_4 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 3 & 2 & 6 \\ 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} F_{1111} \\ F_{112} \\ F_{13} \\ F_{22} \\ F_4 \end{bmatrix}$$

- 可以依次求出F_4,F_{22},F_{13},F_{112},F_{1111}。复杂度为$O(m^2)$ 。
- 细节（求中间这个系数矩阵）可暂时跳过。将采用更好的方法。

更好的方法

- **定理**：$F_{11\ldots1} \;=\; \sum_T c_T G_T$。　（大致是O(m)）
  - c_T的定义如下：T的每个大小为i的组的贡献为$(-1)^{i-1}(i-1)!$
- 举例
  - 大小为1,2,3,4的组，贡献为 1,-1,2,-6,..
  - c_{1111}=1*1*1*1=1。c_{112}=1*1*(-1)=-1。c_{13}=1*2=2。c_{22}=(-1)(-1)=1。c_4=-6。
  - 因此，根据定理 **$F_{1111}= G_{1111} - G_{112} + 2G_{13} + G_{22} - 6G_4$.**
- **暴力验证k=4的情况**
  - $F_4=G_4$, $F_{22}=G_{22}-3G_4$，$F_{13}=G_{13}-4G_4$,
  - $F_{112}=G_{112}-3G_{13}-2G_{22}+12G_4$.
  - **$F_{1111}=G_{1111}-G_4-(G_{22}-3G_4)-(G_{13}-4G_4)-(G_{112}-3G_{13}-2G_{22}+12G_4)$**

- **定理**：$F_{11\ldots1} \;=\; \sum_T c_T G_T$。　（大致是O(m)）
  - c_T的定义如下：T的每个大小为i的组的贡献为$(-1)^{i-1}(i-1)!$
- 举例
  - 大小为1,2,3,4的组，贡献为 1,-1,2,-6,..
  - c_{1111}=1*1*1*1=1。c_{112}=1*1*(-1)=-1。c_{13}=1*2=2。c_{22}=(-1)(-1)=1。c_4=-6。
  - 因此，根据定理 $F_{1111}= G_{1111} - G_{112} + 2G_{13} + G_{22} - 6G_4$.
- **暴力验证k=4的情况**
  - $F_4=G_4$, $F_{22}=G_{22}-3G_4$，$F_{13}=G_{13}-4G_4$,
  - $F_{112}=G_{112}-3G_{13}-2G_{22}+12G_4$.
  - $F_{1111}=G_{1111}-G_4-(G_{22}-3G_4)-(G_{13}-4G_4)-(G_{112}-3G_{13}-2G_{22}+12G_4)$

证明$\sum_T c_T G_T = F_{11\ldots1}$ (nontrivial)

- 已知$G_T = \sum_{T'} a_{T,T'} F_{T'}$。
  - $a_{T,T'}$代表矩阵的T行T'列。

$$\begin{bmatrix} G_{1111} \\ G_{112} \\ G_{13} \\ G_{22} \\ G_4 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 3 & 2 & 6 \\ 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} F_{1111} \\ F_{112} \\ F_{13} \\ F_{22} \\ F_4 \end{bmatrix}$$

- $\sum_T c_T G_T = \sum_T c_T \sum_{T'} a_{T,T'} F_{T'}$

  $= \sum_{T'} F_{T'} (\sum_T c_T a_{T,T'})$
- 显然，$T' = 11\ldots1$时，$\sum_T c_T a_{T,T'} = c_{11\ldots1} = 1$。
- 还需证明：$T' \neq 11\ldots1$时，$\sum_T c_T a_{T,T'} = 0$。

- 对于某个固定的T'，思考$\sum_T c_T a_{T,T'}$代表的含义。

搞懂$\sum_T c_T a_{T,T'}$代表的含义。

- 举例：T’=13。取一个T’类型的划分 Π={a}{bcd}，
  - 它可以细分为{a}{b}{cd}，{a}{c}{bd}，{a}{c}{ad}；因此$a_{112,13}=3$
  - 也可以细分为{a}{b}{c}{d}，因此$a_{1111,13}=1$。 同理 $a_{13,13}=1$。
- $\sum_T c_T a_{T,T'}$ 实际上是先对{a}{bcd}细分，细分后每一段的贡献乘积。
- 对T做细分，等价于对T的每一段做细分！

- *假定T的各段长度为$len_1,\ldots len_t$。*
- *定义len个元素的一个划分的值(value)为划分中个段的贡献的乘积。*
- $\sum_T c_T a_{T,T'} = \prod_{i=1}^{t} \sum_{\pi:[len_i]\text{的一个划分}} value(\pi)$.

$$\prod_{i=1}^{t} \sum_{\pi:[len_i]\text{的一个划分}} value(\pi)$$

- 定义$f(n)=\sum_{\pi:[n]\text{的一个划分}} value(\pi)$
- 上式进一步简化为：$\sum_T c_T a_{T,T'}=\prod_{i=1}^{t} f(len_i)$。 len_1~len_t为T'的各段长度。
- 我们将证明 $f(n)=\begin{cases}1 & n=1\\ 0 & n>1\end{cases}$。
- 这意味着
  - 如果T'的每段长度均为1，则$\sum_T c_T a_{T,T'}$=1；
  - 如果T'的某段长度大于1，则$\sum_T c_T a_{T,T'}$=0.
- 因此根据前述推导，定理得证。

引理： $\sum_{\pi:[n]\text{的划分}} value(\pi) = \begin{cases} 1 & n = 1 \\ 0 & n > 1 \end{cases}$

π每个大小为i的组的贡献为$(-1)^{i-1}(i-1)!$

- 证明：
  - N=1时，显然。n>1时，归纳证明如下。
  - 枚举[n]的划分时，可以根据元素n所在那组的长度进行分类。

$$f(n) = \sum_{\pi:[n]\text{的划分}} value(\pi)$$

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

$$= \binom{n-1}{n-2} (-1)^{n-2}(n-2)! + (-1)^{n-1}(n-1)! = 0$$

本题总结

- 一、$\sum_{\pi:[n]}$的划分 $value(\pi) = \begin{cases} 1 & n = 1 \\ 0 & n > 1 \end{cases}$。
- 二、$\sum_T c_T a_{T,T'} = \prod_{i=1}^{t} f(len_i)$ $= 1$ （T'=11⋯.1) 或 0 （否则）。
- 三、推出定理： $F_{11\ldots1} = \sum_T c_T G_T$。
  - 容斥系数C_T 为T的各段贡献的乘积；长为i的段贡献(-1)^(i-1) * (i-1)!。
  - $G_T := \sum_{\Pi:类型为T} g_\Pi$. $F_T := \sum_{\Pi:类型为T} f_\Pi$.
- 四、g_Π好求（可以背包DP)，容易求得G_T。

- 要求各不相同这种约束条件可考虑该方法。

另一种解法（基于容斥）

- 以k=4为例。
- A_1：满足$b_1=b_2$的解的集合。
- A_2：满足$b_1=b_3$的解的集合。
- A_3：满足$b_1=b_4$的解的集合。
- A_4：满足$b_2=b_3$的解的集合。
- A_5：满足$b_2=b_4$的解的集合。
- A_6：满足$b_3=b_4$的解的集合。
- 目标： $|U| - |A_1 \cup A_2 \cup A_3 \cup A_4 \cup A_5 \cup A_6|$

$|U|-|A_1|\cup|A_2|\cup|A_3|\cup|A_4|\cup|A_5|\cup|A_6|$

- 根据容斥原理:

$$\begin{aligned}&\text{Ans}\\&=|U|-(|A_1|-\cdots-|A_6|)\\&+(|A_1\cap A_2|+\ \ldots+|A_5\cap A_6|)\\&-(|A_1\cap A_2\cap A_3|+\cdots+)\ \ldots\\&+\ |A_1\cap A_2\cap A_3\cap A_4\cap A_5\cap A_6|\end{aligned}$$

- 观察:
  - $|A_1\cap A_2\cap A_4|$ 就是记录的$G_{\{1,2,3\},\{4\}}$。
  - |U|记录的就是$G_{\{1\},\{2\},\{3\},\{4\}}$。
  - 所以Ans是 $\{G_?\}$ 的线性组合。
- 反过来, 如何求系数?
  - G {1}{234}, G{12}{34}, G{1234} 被计算了几次?

A_1: 满足$b_1=b_2$的解的集合。
A_2: 满足$b_1=b_3$的解的集合。
A_3: 满足$b_1=b_4$的解的集合。
A_4: 满足$b_2=b_3$的解的集合。
A_5: 满足$b_2=b_4$的解的集合。
A_6: 满足$b_3=b_4$的解的集合。

G {1}{234}，G{12}{34} 被计算了几次?

- 若要将{234} 连成一个分量:
  - 3个点之间三条边都选：A_4,A_5,A_6
    - 系数为-1
  - 3个点之间选了两条边：3种方案
    - 系数为1.
  - 因此G{1}{234}的系数为3-1=2.
  - 同理，整个G_{13}的系数为2.
- 若要获得{12}、{34}两个分量
  - 必须恰好选了：A_1,A_6。
    - 系数为1.
  - 因此$G_{\{12\}\{34\}}$的系数为1.即，G_{22}的系数为1.

A_1：满足$b_1=b_2$的解的集合。
A_2：满足$b_1=b_3$的解的集合。
A_3：满足$b_1=b_4$的解的集合。
A_4：满足$b_2=b_3$的解的集合。
A_5：满足$b_2=b_4$的解的集合。
A_6：满足$b_3=b_4$的解的集合。

G{1234}在ans中应被计算几次？

- 6条边全选
  - 1种方案。符号为正。
- 选恰好5条边
  - 6种方案。符号为负。
- 选恰好4条边
  - 15种方案。符号为正。
- 选恰好3条边使得1234连通起来
  - 链：4*3*2/2=12。符号为正
  - 爪：4。符号为正。
- 因此，答案为 1-6+15-16=-6。 故 G_4的系数为-6。

A_1：满足$b_1=b_2$的解的集合。
A_2：满足$b_1=b_3$的解的集合。
A_3：满足$b_1=b_4$的解的集合。
A_4：满足$b_2=b_3$的解的集合。
A_5：满足$b_2=b_4$的解的集合。
A_6：满足$b_3=b_4$的解的集合。

如何计算G_{2334}? （k=12)

- Fn:表示 n个点的连通图的 带权和。
  - 这个连通图有偶数条边时，权值为1
  - 这个连通图有奇数条边时，权值为-1.
  - 也就是说，$F_n = \sum_{E \subseteq K_n} [E连通] * (-1)^{|E|}$ 。
- 容易发现，$G_{2334} = F_2 * F_3 * F_3 * F_4$。 （显然成立）

- 举例
  - $F_1=1$，$F_2=-1$，$F_3=2$，$F_4=-6$。
  - $G_{22}= -1*-1=1$，$G_{13}=1*2=2$，$G_4=-6$.

如何计算$F_n = \sum_{E\subseteq K_n}[E连通] * (-1)^{|E|}$

如果不带权。套egf公式解决 。

- 假设$G_n = \sum_{E\subseteq K_n}(-1)^{|E|}$ 。 （不要求连通）(带权图计数）
- 令$\hat{F}(x)$为 <0,F$_1$,F$_2$,⋯,>的EGF。令$\hat{G}(x)$为<G$_0$,G$_1$,G$_2$,⋯>的EGF。
- 猜想： $\hat{G}(x) = \exp(\hat{F}(x))$。 （证明见下页）
- 由此可以算出F_n
  - 首先，$\hat{G}(x) = 1 + x$。 （n<=1 ⇔ |Kn|=0 ⇔ G$_n$=1。）
  - 于是，$\hat{F}(x) = \ln(1+x) = \sum_{n=1}^{\infty}(-1)^{n-1}\frac{1}{n}x^n$ 。
  - 于是，$F_n = (-1)^{n-1} * (n-1)!\ \ (n \geq 1)$

问题解决！

$$F_n=\sum_{E\subseteq K_n}[E连通]*(-1)^{|E|}$$

$$G_n=\sum_{E\subseteq K_n}(-1)^{|E|}$$

证明 $\hat{G}(x)=\exp(\hat{F}(x))$

- 令$G_{k,n}=\sum_{E\subseteq K_n,\ E含k个连通分量}(-1)^{|E|}$

$-1^{|E|}$改为任意$a^{|E|}$，仍然成立！
重复一遍无权版本的证明即可。

- $G_{k,n}=\frac{1}{k!}\sum_{(c_1,\ldots c_k)为[n]的划分}\sum_{E_1:c_1的连通图,\ldots E_k:c_k的连通图}(-1)^{|E_1|+\cdots|E_k|}$
- $G_{k,n}=\frac{1}{k!}\sum_{(c_1,\ldots c_k)为[n]的划分}\mathrm{F}_{|c_1|}*\cdots*F_{|c_k|}$
- $G_{k,n}=\frac{1}{k!}\sum_{\mathrm{i}_1+\cdots i_k=n}\frac{n!}{i_1!\ldots i_k!}\mathrm{F}_{i_1}*\cdots*F_{i_k}.$
- 令$H_{k,n}=k!\,G_{k,\mathrm{n}}$. 则 H_k这个序列的EGF为$\hat{F}(x)^k$。G_k的EGF为$\frac{1}{k!}\hat{F}(x)^k$。 最终$G=G_0+G_1+G_2+\cdots$ 的EGF即为： $\exp(\hat{F}(x))$

谢谢！

- 有问题可联系 QQ:154874681.