组合数学

张昕渊

October 13, 2023

第一节内容

- 容斥原理、莫比乌斯反演与min-max容斥

容斥原理

Theorem (容斥原理)

对于集合$A_1, A_2, \ldots, A_n$,

$$|\bigcup_{i=1}^{n} A_i| = \sum_{S \subseteq [n]} (-1)^{|S|-1} |\bigcap_{i \in S} A_i|.$$

容斥原理

Theorem (容斥原理等价形式)

对于集合$B_1, B_2, \ldots, B_n \subseteq U$,

$$|\bigcap_{i=1}^{n} \overline{B_i}| = \sum_{S\subseteq[n]} (-1)^{|S|} |\bigcap_{i\in S} B_i|,$$

当S为空集时，$\bigcap_{i\in S} B_i = U$（全集）。

- 如何理解且应用容斥原理：B_i为一系列的“坏事件”。计算所有坏事件都不发生困难，但计算某些坏事件发生要来得简单的时候就可以考虑容斥原理！
- 本质是对一些难以处理的条件的“放松”。

容斥原理与莫比乌斯反演

- 莫比乌斯反演实际上可以看作是容斥原理的一个示例：
- 令$f(n) = \sum_{d|n} g(d)$。已知$f(1), f(2), \ldots, f(n)$，求$g(n)$。
- 令$G_1, G_2, \ldots, G_n$为不交集合满足$|G_i| = g(i)$，$F_i = \cup_{d|i} G_i$。
- 则$G_n = F_n \setminus \bigcup_{p|n} F_{n/p}$，此时全集为$F_n$，坏事件为元素落在某个$F_{n/p}$中。
- 假设n有$p_1, p_2, \ldots, p_m$这m个素因子，由容斥原理：$|G_n| = \sum_{S \subseteq [m]} (-1)^{|S|} |F_{n/p_S}| = \sum_{d|n} \mu(d) f(n/d)$。

错排

- 有多少个排列$p = [p_1, p_2, \ldots, p_n]$满足对于所有的$i$都有$p_i \neq i$。
- 令B_i为$p_i = i$的坏事件，则k个坏事件的交为k个位置确定之后的排列方案数，为$(n-k)!$种。因此，

$$
\begin{aligned}
|\bigcap_{i=1}^{n} \overline{B_i}| &= \sum_{S\subseteq[n]} (-1)^{|S|} |\bigcap_{i\in S} B_i| \\
&= \sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)! \\
&= n! \sum_{k=0}^{n} \frac{(-1)^k}{k!}.
\end{aligned}
$$

不定方程非负整数解计数

- 求$x_1, x_2, \ldots, x_n$满足$0 \le x_i < C$且$\sum_{i=1}^{n} x_i = m$的整数解个数。
- 考虑容斥原理：$x_i < C$这个条件难以处理，我们令B_i为这个条件被违反的坏事件。
- $\bigcap_{S\in[n]} B_i$这个集合指的是所有$i \in S$有$x_i \ge C$，且$\sum_{i=1}^{n} x_i = m$的非负整数解个数。
- 这等价于求解$\sum_{i=1}^{n} x_i = m - kC$的非负整数解个数。
  - 插板法解决：考虑$m - kC$个球，$n - 1$个板子将这一些球分成n块，每一块内球的个数对应着x_i的取值，因此解的个数为$\binom{m-kC+n-1}{n-1}$。
  - $\sum_{i=1}^{n} x_i \le m$呢？只需引入一个冗余变量即可。

简单复形体积问题

- 求下列n维几何体的体积：

$$S = \{0 < x_i < 1 | \sum_{i=1}^{n} x_i \leq m\}.$$

- $x_i < 1$不好处理，我们考虑坏事件B_i为$x_i > 1$，容斥后只需求$S = \{0 < x_i | \sum_{i=1}^{n} x_i \leq m - C\}$的体积，为$(m - C)/n!$。

带标号的DAG计数

- 求n个带标号顶点的DAG个数。
- 令f_n为答案，A_i为i号顶点入度为0的事件，则$\bigcap_{i\in S} A_i$为$|S|$个顶点入度均为0，满足该种条件的DAG方案数为$2^{|S|(n-|S|)}f_{n-|S|}$。因此

$$f_n = |\bigcup_{i\in S} A_i| = \sum_{S\subseteq[n]}(-1)^{|S|-1}|\bigcup_{i\in S} A_i|$$
$$= \sum_{k=1}^{n}\binom{n}{k}(-1)^{k-1}2^{k(n-k)}f_{n-k}$$

例题：逆序对个数

- 求长度为n且逆序对个数为K的排列个数，模10^9+7。
- $n, K \le 10^5$。

例题：逆序对个数

- 题目等价于求$x_1, x_2, \dots, x_{n-1}, x_n$满足下述条件的方案数。
  - $0 \le x_i < i$;
  - $\sum_{i=1}^{n} x_i = K$
- 考虑容斥原理，令B_i为$x_i < i$被违反的坏事件，则对于任意的$S \subseteq [n]$，

$$|\bigcap_{i \in S} B_i| = \binom{K - \sum_{i \in S} i + |S| - 1}{|S| - 1},$$

  因此，我们只需统计$f_{i,j}$使得$|S| = i$且$\sum_{k \in S} k = j$的方案数即可。
- $f_{i,j} = f_{i,j-i} + f_{i-1,j-i} - f_{i-1,j-(n+1)}$，且第一维为$O(\sqrt{K})$量级。
- 整体复杂度为$O(K\sqrt{K})$。

例题：LEQ and NEQ (easy version)

- 求满足下列条件的序列$A_1, A_2, \ldots, A_n$个数，模$10^9 + 7$。
  - $A_i \leq X_i$;
  - $A_i \neq A_{i+1}$.
- $n \leq 5000, X_i \leq 10^9$。

例题：LEQ and NEQ(easy version)

- 非常自然的考虑对条件$A_i \neq A_{i+1}$进行容斥：令B_i为事件$A_i = A_{i+1}$，则$|\bigcap_{i\in S} B_i|$为一系列区间最小值的乘积。
- 令$dp_{i,k}$为A_i和A_{i+1}不在同一个区间，坏事件交的个数奇偶性为k的总代价。
- 考虑$i+1$到$i+\ell$被合并至一起，则$dp_{i,k}$可以转移到$dp_{i+\ell,k+(\ell-1) \mod 2}$。
- 答案为$dp_{n,0} - dp_{n,1}$。
- 本题可以加速至线性时间。

例题：Perfect matching

- 给一个$2n$个点的树T，求T的补图中完美匹配的个数，答案对998244353取模。
- $n \leq 2000$。

例题：Perfect matching

- 由容斥原理（令B_i为第i条边在树上的坏事件），问题转化为求树上大小为K的匹配个数，这可以在$O(n^2)$的时间内通过经典的树dp求出。

例题：ABC string

- 求有多少个由A, B, C构成的序列s，使得A, B, C的出现次数分别为a, b, c，且序列中不出现连续子串ABC，BCA或者CAB。
- $a, b, c \le 10^6$。

例题：ABC string

- 考虑容斥原理，问题转化如下：
  - 选择k个位置$x_1, x_2, \ldots, x_k$使得这$s[x_i : x_i + 2]$恰好为ABC, BCA或者CAB。此时方案数乘以$(-1)^k$贡献给总答案。求总答案为多少
- 对于每个位置x_i，我们将x_i, x_{i+1}, x_{i+2}相连。则最终$[n]$可以划分为$I_1, I_2, \ldots, I_\ell$这ℓ个区间的并。
- 固定某种特定的区间划分，我们首先计算所有可以导出这种区间划分的位置选择加权和，权重为$(-1)^k$。
- 每个区间单独处理，令L_i为第i个区间长度，则第L_i个区间权重和为1如果$L_i \mod 3 = 1$，权重和为0如果$L_i \mod 3 = 2$，否则为-1。
- 问题转化成如下：
  - $[n]$划分成若干段，每一段都是$ABC/BCA/CAB$的若干次重复加上$A/B/C$或者不加上任何元素。每一种划分对答案的贡献是此时填ABC满足条件的方案数乘以$(-1)^w$，其中w是不加$A/B/C$的区间数。
- 将ABC打包看成D，固定D的个数，对于每一种$ABCD$的序列，划分总贡献为$(-2)^D$ 或者$-3 * (-2)^{D-1}$，取决于最后一个位置是否为D。

min-max容斥

- 给定集合$U = \{a_1, a_2, \ldots, a_n\}$，

$$\min(U) = \sum_{\emptyset \neq S \subseteq U} (-1)^{|S|-1} \max(S),$$

  min, max交换也成立。
- 不妨假设$a_1 \leq a_2 \leq \ldots \leq a_n$
- 当$S = \{a_1\}$时，$\max(S) = a_1$。
- 当$S \neq \{1\}, \emptyset$，$\max(S) = \max(S \oplus \{1\})$，其他会相互抵消。
- 推论：当$a_1, a_2, \ldots, a_n$为随机变量时，

$$E[\min(U)] = \sum_{\emptyset \neq S \subseteq U} (-1)^{|S|-1} E[\max(S)].$$

min-max容斥

- 给定集合$U = \{a_1, a_2, \ldots, a_n\}$,

$$\mathrm{kth} - \min(U) = \sum_{\emptyset \neq S \subseteq U} (-1)^{|S|-k} \binom{|S|-1}{k-1} \max(S),$$

- 同样的有期望版本。

例题：按位或

- 给定正整数n以及变量$x = 0$，每一秒从$[0, 2^n - 1]$中以概率分布p独立选取正整数a，将x更新至$x|a$。当$x = 2^n - 1$时过程停止，求过程终止的期望时间。
- $n \le 20$。

例题：按位或

- 每一位单独考虑，令X_i为第i位被置为1所需时间，则答案为$E[\max(X_0, X_1, \ldots, X_{n-1})]$。
- 由min-max容斥：

$$E[\max(X_0, X_1, \ldots, X_{n-1})] = \sum_{\emptyset \neq S \subseteq [n]} (-1)^{|S|-1} E[\min(X_S)],$$

- $E[\min(X_S)]$: S中某一位被置为1的期望时间。令q_S为抽取数字a包含S中某一位的概率，则$E[\min(X_S)] = 1/q_S$。
- 对于每一个S，计算q_S的问题等价于计算高维前缀和，可以在$O(n2^n)$的时间内求解。

一维带碰撞壁随机游走模型

Problem

求满足以下条件的长度为n的序列$a = [a_1, a_2, \ldots, a_n]$个数：

- $a_1 = 0, \ a_n = A;$
- $|a_{i+1} - a_i| = 1;$
- $a_i > -C$。

- 对于这种问题，我们一般考虑“André反射原理”。
- 我们将不合法的折线序列与从$-2C$出发的折线序列一一对应起来：即我们考虑所有满足下列条件的序列b
  - $b_1 = 2C, \ b_n = A;$
  - $|b_{i+1} - b_i| = 1$。
- 非法的a与b的一一对应关系：从开始到第一个达到$-C$的点的这段区间上下翻转。

Figure: Andre's reflection principle

一维带碰撞壁随机游走模型

Problem

求满足以下条件的长度为n的序列$a = [a_1, a_2, \ldots, a_n]$个数：

- $a_1 = 0,\ a_n = A;$
- $|a_{i+1} - a_i| = 1;$
- $a_i > -C$。

- 总方案数：0到A的长度为n的折线方案数：$\binom{n}{\frac{n+A}{2}}$;
- 坏的方案数：$-2C$到A的长度为n的折线方案数：$\binom{n}{\frac{n+A+2C}{2}}$。

一维带碰撞壁随机游走模型

Problem

求满足以下条件的长度为n的序列$a = [a_1, a_2, \ldots, a_n]$个数:

- $a_1 = 0,\ a_n = A;$
- $|a_{i+1} - a_i| = 1;$
- $-C < a_i < C$。

- 如果有两侧的限制将会如何：不满足条件的折线可能会来回穿过$y = -C$与$y = C$。
- 令A_i为折线存在碰壁$y = C,\ y = -C,\ y = C,\ y = -C\ldots$的长度为$i$的子序列；$B_i$为折线存在碰壁$y = -C$，$y = C$，$y = -C$，…的长度为$i$的子序列。
- 合法方案数为总方案数减去$\sum_{i=1}^{+\infty}(-1)^{i-1}(|A_i| + |B_i|)$（容斥原理中的算两次技巧）。
- $|A_i|$方案应该如何计算：类似于反射原理，我们可以将原点按$y = C, y = -C,\ y = C, \ldots$依次翻转$n$次，则$A_i$中的折线与起始点$2kC$，终点为$A$的折线方案数一一对应。

第二节内容

- 群与群作用；
- Burnside引理与Pólya计数

群的定义

- 令G为一个集合，$\cdot : G \times G \to G$为一个二元运算。我们称$(G, \cdot)$为一个群，若下列条件同时从成立：
  1. **(结合律)** $\forall g_1, g_2, g_3 \in G$, $(g_1 \cdot g_2) \cdot g_3 = g_1 \cdot (g_2 \cdot g_3)$;
  2. **(幺元/单位元)** $\exists e \in G$, $\forall g \in G$, $e \cdot g = g \cdot e = g$;
  3. **(逆元)** $\forall g \in G$, $\exists g^{-1} \in G$，$g \cdot g^{-1} = g^{-1} \cdot g = e$.
- 我们之前提到的两个例子都可以被抽象为群：
  1. $(Z_n, +)$（**模n加法群**）：这里的加法指的是模n意义下的加法，单位元为0，$x > 0$的逆元为$n - x$。
  2. $(Z_p^*, \times)$（**模p乘法群**）：集合为$\{1, 2, \ldots, p-1\}$，单位元为1，逆元为模p下的逆元。
  3. $(R_{360}, \cdot)$（**所有可能旋转及其复合构成的群**）：这里的复合操作是自然意义下的复合，单位元为不进行旋转操作，$x \in \{1, 2, \ldots, 359\}$度旋转的逆元是$360 - x$度旋转。

群的定义

- 令G为一个集合，$\cdot : G \times G \to G$为一个二元运算。我们称$(G, \cdot)$为一个群，若下列条件同时从成立：
  1. **(结合律)** $\forall g_1, g_2, g_3 \in G$, $(g_1 \cdot g_2) \cdot g_3 = g_1 \cdot (g_2 \cdot g_3)$;
  2. **(幺元/单位元)** $\exists e \in G$, $\forall g \in G$, $e \cdot g = g \cdot e = g$;
  3. **(逆元)** $\forall g \in G$, $\exists g^{-1} \in G$，$g \cdot g^{-1} = g^{-1} \cdot g = e$.
- 算法中常常提及的置换(permutation)全体与置换的复合同样构成群$(S_n, \cdot)$（**置换群**）。
  1. $p = (p_1, p_2, \ldots, p_n)$称为一个置换，若$1, 2, \ldots, n$均恰好在$p$中出现1次；
  2. 两个置换p, q的复合被定义为$(p \cdot q)_i = p_{q_i}$；
  3. 单位元$e = (1, 2, \ldots, n)$；
  4. p的逆元p^{-1}满足$p^{-1}_{p_i} = i$。

群作用

- 令G为群，X为集合，群作用为映射$\circ: G \times X \to X$满足以下条件：
  1. 结合律：对于$f, g \in G$，$x \in X$，我们有$f \circ (g \circ x) = (f \cdot g) \circ x$；
  2. 单位元：对于$x \in X$，我们有$e \circ x = x$。
- 一些具体例子：
  - G为群，G作用在本身也是一种群作用；
  - $(R_{360}, \cdot)$为旋转群，集合为多边形全体：群作用为将某个多边形依照原点旋转若干角度。
  - S_n为置换群，集合为长度为n的序列全体：群作用为将序列$a = [a_1, a_2, \ldots, a_n]$变为$a_p = [a_{p_1}, a_{p_2}, \ldots, a_{p_n}]$。
- 假设X为一个有限集合，我们称元素x, y等价（一般写作$x \sim y$），如果存在群元素$g \in G$满足$g \circ x = y$。

Problem

给定群G、集合X以及群作用$\circ$，问有多少个等价类（X/G表示等价类全体）。

Burnside引理

Problem

给定群G、集合X以及群作用$\circ$，问有多少个等价类（X/G表示等价类全体）。

- 项链染色计数为上述问题的一个特殊版本：
  - 给一个圈上的n个点用两种颜色染色，问有多少种本质不同的染色方案。两种方案视为一致如果可以通过旋转从一种变为另一种。
  - 循环群$C_n = \{(2, 3, \ldots, n, 1)^t | t \geq 0\}$，集合$X$为长度$n$的01序列全体，群作用是置换群到集合$X$的群作用。
- $n = 4$，6个等价类：
  1. $\{0000\}$;
  2. $\{0001, 0010, 0100, 1000\}$；
  3. $\{0101, 1010\}$，$\{1001, 0011\}$；
  4. $\{1110, 1101, 1011, 0111\}$;
  5. $\{1111\}$。

Burnside引理

Problem

给定群G、集合X以及群作用$\circ$，问有多少个等价类（X/G表示等价类全体）。

- 给每个元素$x\in X$附上权值$w_x=\frac{1}{|Gx|}$，其中$Gx=\{gx|g\in G\}$为x所在等价类中元素的个数。则$|X/G|=\sum_{x\in X}w_x$。
- 等价类中元素个数与稳定子的个数密切相关。
    - 元素$x\in X$的稳定子$G^x=\{g\in G|gx=x\}$；
    - $|Gx||G^x|=|G|$：令G'为集合满足$|G'|=|Gx|$且$\{gx|g\in G'\}=Gx$的群元素，我们只需证明$G'\times G^x\to G:(g_1,g_2)\to g_1\cdot g_2$是一个双射即可。
    - 一方面，若$g_1\cdot g_2=g_3\cdot g_4$，则$g_3^{-1}\cdot g_1=g_4\cdot g_2^{-1}$。由于$g_2,g_4$为稳定子，则$g_4\cdot g_2^{-1}$也是稳定子，因此$g_3^{-1}\cdot g_1=e$（若不然，则$g_3$与$g_1$作用在$x$上得到的元素一致，与$|G'|=|Gx|$矛盾）。
    - 另一方面，对于任意的$g\in G$，令$g'\in G'$满足$g'x=gx$。则$g'^{-1}\cdot g$为稳定子，拆分为g'与$g'^{-1}\cdot g$。

Burnside引理

Problem

给定群G、集合X以及群作用。，问有多少个等价类（X/G表示等价类全体）。

$$
\begin{aligned}
|X/G| &= \sum_{x\in X}\frac{1}{|Gx|} = \sum_{x\in X}\frac{|G^x|}{|G|} = \frac{1}{|G|}\sum_{x\in X}\sum_{g\in G}[gx=x] \\
&= \frac{1}{|G|}\sum_{g\in G}\{x : gx = x\}.
\end{aligned}
$$

- 上述就是Burnside定理。数等价类的个数只需数群内每个元素不动点的个数即可。

Burnside引理

$$|X/G| = \sum_{x\in X} \frac{1}{|Gx|} = \sum_{x\in X} \frac{|G^x|}{|G|} = \frac{1}{|G|} \sum_{x\in X} \sum_{g\in G} [gx = x]$$
$$= \frac{1}{|G|} \sum_{g\in G} \{x : gx = x\}.$$

- 项链染色问题
  1. $g = (1, 2, 3, 4)$: 不动点个数为16（所有染色）；
  2. $g = (2, 3, 4, 1), (4, 1, 2, 3)$：不动点个数为2（一致染色）；
  3. $g = (3, 4, 1, 2)$：不动点个数为4（1-3染色一致，2-4染色一致）。
  4. 等价类个数：$\frac{16+2*2+4}{4} = 6$。

Burnside引理

$$|X/G| = \sum_{x\in X}\frac{1}{|Gx|} = \sum_{x\in X}\frac{|G^x|}{|G|} = \frac{1}{|G|}\sum_{x\in X}\sum_{g\in G}[gx=x]$$
$$= \frac{1}{|G|}\sum_{g\in G}\{x : gx = x\}.$$

- 项链染色问题
  1. $g = (1,2,3,4)$：不动点个数为16（所有染色）；
  2. $g = (2,3,4,1),(4,1,2,3)$：不动点个数为2（一致染色）；
  3. $g = (3,4,1,2)$：不动点个数为4（1-3染色一致，2-4染色一致）。
  4. 等价类个数：$\frac{16+2*2+4}{4} = 6$。

Polya定理

- 当G为置换群S_n的子群（比如我们之前提到的循环群），X为$[q]^n$，群作用为$a \to a_p$时：

  $$|X/G| = \frac{1}{|G|} \sum_{g \in G} q^{c(g)},$$

  其中$c(g)$是有向图$i \to g_i$中圈的个数。
- 这是Burnside引理的自然推论。

Polya定理

- 当G为置换群S_n的子群（比如我们之前提到的循环群），X为$[q]^n$，群作用为$a \to a_p$时：

$$|X/G| = \frac{1}{|G|}\sum_{g\in G} q^{c(g)},$$

  其中$c(g)$是有向图$i \to g_i$中圈的个数。
- 这是Burnside引理的自然推论。

例题：Cube

- 给一个正方体的六个面填正整数，有多少种方案使得六个面的数之和为S。
- 两种填数方案被视为等同，如果我们可以通过旋转正方体从一种变为另一种（数字本身是不认为具有方向性的）
- $S \leq 10^{18}$。

例题：Cube

- 给定一个正方体，一共有24种旋转方案（决定哪一面朝上，之后决定哪一面朝前）。
- 利用Burnside引理，对每种旋转方案统计填数方案即可（填数方案都会是$\sum_{i=1}^{k} a_i x_i = S$的形式，$\sum_{i=1}^{k} a_i = 6$，可以考虑数位$dp$或直接计数之类的）。

例题：Count Unlabeled Graph

- n个无标号顶点涂K种颜色的无向图有多少种;
- 两个图被视为同一种图，如果可以找到顶点的双射一一对应。
- $n=3$，$k=1$：4种（只有四个本质不同的3顶点无向图）；
- $n=3$，$k=2$： 12种。
- $1 \leq n, k \leq 30$。

例题：Count Unlabeled Graph

- 利用Burnside引理：
  - 群：置换群；
  - 元素：带标号、染色的无向图。
  - 群作用：将上述一个染色无向图通过重新标号变为另一个染色无向图。
- Burnside引理：对于每一个群中元素g，求染色无向图个数G使得置换后保持不变。
- 考虑有向图$i \to g_i$，令$C_1, C_2, \ldots, C_k$为有向图中环的拆分。
  - 对于块内部，不妨令$C_1 = \{1, 2, \ldots, m\}$，置换为$(2, 3, \ldots, m, 1)$。则我们需要满足$a_{i,j} = a_{i+1,j+1}$。所以我们只需要考虑$a_{1,i}, i \leq m/2$即可。方案数为$2^{\lfloor m/2 \rfloor}$.
  - 块之间C_i, C_j的边可以被分为$\gcd(|C_i|, |C_j|)$类。
- 方案数是$K^k \cdot 2^{f(|C_1|,|C_2|,\ldots,|C_k|)}$。枚举所有划分即可。
- 时间复杂度$O(p(n)\mathrm{poly}(n))$，$p(n)$是划分数。

FFT与NTT

Problem

考虑多项式$f=\sum_{i=0}^{n} f_i x^i$与$g=\sum_{i=0}^{n} g_i x^i$，如何快速求解多项式的乘积$f\cdot g$?

- FFT的整体框架分为三部分：
  1. 从系数到点值：选取合适的$x_0, x_1, \ldots, x_{N-1}$，快速求$f(x_0), f(x_1), \ldots, f(x_{N-1})$的值（以及$g$）；
  2. 点值与点值相乘：$(f\cdot g)(x_k)=f(x_k)g(x_k)$;
  3. 从点值到系数：给定$fg(x_0), fg(x_1), \ldots, fg(x_{N-1})$，求$fg$的系数。
- 令ω_N为$x^N=1$的N次单位根：
  即$\omega_N=\exp(2\pi i/N)=\cos(2\pi/N)+i\sin(2\pi/N)$（这里的$i$是虚数/复数，只有这里的i是指复数，后续的i是index），我们首先看一下第一步如何快速实现。

第三小节内容

- 卷积：FFT/NTT、FWT/子集卷积

FFT与NTT

- 我们这里不妨假设N是2的幂次。
- 考虑$f_0(x) = a_0 + a_2x + \ldots + a_{N-2}x^{(N-2)/2}$, $f_1(x) = a_1 + a_3x + \ldots + a_{N-1}x^{(N-2)/2}$（奇数项和偶数项分别提出来）。
- 则$f(x) = f_0(x^2) + xf_1(x^2)$;
- 计算$f(x_j) = f(\omega_N^j) = f_0(\omega_{N/2}^j) + \omega_N^j f_1(\omega_{N/2}^j)$。
- 因此，我们只需递归求解对f_0, f_1的子问题即可，复杂度为$O(n \log n)$。

FFT与NTT

- 如何从点值到系数?
- 注意到f的各项系数到点值的过程本质上是一个线性变换。
- 令$v_f = [f_0, f_1, \ldots, f_n]$，矩阵$A = (\omega_N^{ij})_{0\le i,j\le n}$，点值向量$v = Av_f$。
- 因此，$v_f = A^{-1}v$，只需要求范德蒙行矩阵的逆以及矩阵乘法即可。
- 一般的范德蒙矩阵的逆形式复杂，但是对于我们这里的矩阵来说逆的形式相当简单: $A^{-1} = (\frac{\omega^{-ij}}{N})_{0\le i,j\le n}$。
- 利用一致的递归思想即可从点值到系数。
- Remark: FFT有一些具有常数优化的写法（比如蝴蝶变换技巧），大家有兴趣可以了解一下。

FFT与NTT

- 如何计算$f \cdot g$在F_p的系数?
- 同样的，我们需要找到单位根$\omega^N = 1 \mod p$，其中N是2的幂次。
- 由原根性质可知，ω存在当且仅当$N | p-1$。因此，NTT考虑的素数满足$p = 2^K \cdot r + 1$形式。最普遍的$p = 998244353$。
- 令g为原根，$g^{(p-1)/N}$即为单位根，我们只需将FFT中的ω替换为$g^{(p-1)/N}$即可。

FFT与NTT

- 如何计算$f \cdot g$在F_p的系数?
- 同样的，我们需要找到单位根$\omega^N = 1 \mod p$，其中N是2的幂次。
- 由原根性质可知，ω存在当且仅当$N|p-1$。因此，NTT考虑的素数满足$p = 2^K \cdot r + 1$形式。最普遍的$p = 998244353$。
- 令g为原根，$g^{(p-1)/N}$即为单位根，我们只需将FFT中的ω替换为$g^{(p-1)/N}$即可。
- 有任意模数FFT，由于较为繁琐我们这里不展开。

FFT与NTT

- FFT/NTT的最基本应用：
  - 对于所有的n求卷积式$\sum_{i+j=n} a_i b_j$：最直接的FFT。
  - 多个多项式$f_1, f_2, \ldots, f_n$相乘：分治FFT，类似于归并排序的分治思想。时间复杂度为$O(N\log^2 N)$，N为f_i的次数之和。
  - 递推式求解$f_n = \sum_{i=0}^{n-1} f_i g_{n-i}$：半在线卷积。考虑CDQ分治，假设$n$是偶数，我们已经求出了$f_0, f_1, \ldots, f_{n/2-1}$的值，则

$$f_m = \sum_{i=0}^{m-1} f_i g_{m-i} = \sum_{i=0}^{\frac{n}{2}-1} f_i g_{m-i} + \sum_{i=\frac{n}{2}}^{m} f_i g_{m-i}$$

前者已经可以通过FFT处理，后者已经是一个规模为$\frac{n}{2}$的子问题，可以递归求解。

例题：Games

- 给定N个正整数$A_1, A_2, \ldots, A_N$。求满足下列条件中序列$B = [B_1, B_2, \ldots, B_K]$的个数，模998244353：
  - B_i为$A_1, A_2, \ldots, A_N$中的一者；
  - 对于二进制下每一个数位k，B_i在二进制下第k位为1的个数恰好为7的倍数。
- $1 \leq N \leq 100, 0 \leq A_i \leq 100$，$K \leq 10^{18}$。

例题：Games

- 假设$N=2, A_1=0, A_2=1$，我们只需考虑多项式$(x+1)^K \bmod (x^7-1)$即可（ $\bmod x^7-1$相当于是将x^7替换为1），这可以通过FFT+快速幂求解。
- 更简单一点的求解方式：找到$x^7 \equiv 1 \bmod p = 998244353$的7次单位根$\omega$，求$(x+1)^K$在$\omega^0, \omega^1, \ldots, \omega^6$的点值，然后逆变换回去即可。
- 一般问题为高维版本（$\lceil \log A \rceil = 7$个变量），考虑多项式$f(x_1, x_2, \ldots, x_7) = (\sum_{i=1}^{k} x^{A_i})^K$之类的形式，其中$x^{A_i}$是一些类似于$x_1x_3x_5$的单项式。
- 做法和一维版本类似，求$f(\omega^{i_1}, \omega^{i_2}, \ldots, \omega^{i_7})$的值，然后一维一维逆变换回去即可。

拉格朗日插值法

Problem

已知n次多项式f在$n+1$个点$x_0, x_1, \ldots, x_n$的值，如何求f的各项系数$f_0, f_1, \ldots, f_n$。

- 类似于中国剩余定理的构造，

$$f(x) = \sum_{i=0}^{n} f(x_i) \prod_{j \neq i} \frac{x - x_j}{x_i - x_j}.$$

- 暴力的时间复杂度为$O(n^2)$。
- 考虑$x_i = i$这种非常特殊的点值，可以在$O(n)$的时间内求$f(m)$的值。
- 拉格朗日插值法在知道多项式是一个低次多项式，而单点值(特别是值比较小的时候)容易求的时候很有效。

FWT与子集卷积

- 有时候，我们需要处理的是类似于下列这种情形的卷积：
  - 对于任意的$S \subseteq [n]$，$a_S = \sum_{\substack{U,V\subseteq[n]\\ U\cup V=S}} b_U c_V$。
  - 并可以替换成交或者异或（对称差）。
- FWT大致思路和FFT/NTT一样，都是考虑将序列进行线性变换后将卷积变成逐位相乘的形式，而后进行逆变换。
- 对于并集（或）而言：
  - 变换为$b_S = \sum_{U\subseteq S} a_U$，逆变换为$b_U = \sum_{S\subseteq U} (-1)^{|U|-|S|} a_S$；
  - 正确性：

$$\sum_{U\subseteq S} \sum_{\substack{L,R\subseteq[n]\\ L\cup R=U}} b_L c_R = \sum_{\substack{L,R\subseteq[n]\\ L\cup R\subseteq S}} b_L c_R = \left(\sum_{L\subseteq S} b_L\right)\left(\sum_{R\subseteq S} c_R\right)$$

  - 时间复杂度：$O(n2^n)$（高维前缀和/SOS dp）
- 对于交而言，我们只需要对所有集合取补集即可。

FWT与子集卷积

- 对于异或而言，我们的线性变换会稍微复杂一点：
  - 变换为$b_S = \sum_{U\subseteq[n]}(-1)^{|S\cap U|}a_U$，逆变换为$b_S = \frac{1}{2^n}\sum_{U\subseteq[n]}(-1)^{|S\cap U|}a_U$；
  - 正确性：

$$
\begin{aligned}
&\sum_{U\subseteq[n]}(-1)^{|S\cap U|}\sum_{L,R\subseteq[n]}[L\oplus R = U]a_L b_R\\
&= \frac{1}{2^n}\sum_{U\subseteq[n]}(-1)^{|S\cap U|}\sum_{L,R\subseteq[n]}\sum_{V\subseteq[n]}(-1)^{|V\cap(L\oplus R\oplus U)|}a_L b_R\\
&= \frac{1}{2^n}\sum_{U,V,L,R\subseteq[n]}(-1)^{|S\cap U|}(-1)^{|V\cap L|+|V\cap R|+|V\cap U|}a_L b_R\\
&= \frac{1}{2^n}\sum_{U,V,L,R\subseteq[n]}(-1)^{|(S\oplus V)\cap U|}(-1)^{|V\cap L|+|V\cap R|}a_L b_R\\
&= \sum_{L,R\subseteq[n]}(-1)^{|S\cap L|+|S\cap R|}a_L b_R.
\end{aligned}
$$

  - 时间复杂度：$O(n2^n)$（类似或的情况，SOS dp）

FWT与子集卷积

- 如果我们需要考虑无交并的卷积?
  - 对于任意的$S \subseteq [n]$，$a_S = \sum_{\substack{U,V\subseteq[n]\\U+V=S}} b_U c_V$。
- 我们将b_S, c_V替换为多项式$b_S x^{|S|}$，考虑或卷积，最终得到的a_S为一个多项式。此时a_S中$x^{|S|}$的系数为所求。

例题： xor

- 给定n个正整数$0 \le a_1, a_2, \ldots, a_n < 2^m$，对所有的$0 \le k < 2^m$求下述的值：
  - $\sum_{1\le i,j\le n}[(a_i \oplus a_j)\&k > 0]$
- $n \le 2^{20}$，$m \le 20$。

例题：xor

- 令c_i为$a_j = i$的个数，考虑xor卷积$\sum_{x \oplus y = z} c_x c_y$。
- $(a_i \oplus a_j) \& k = 0$ iff $k \subseteq \overline{a_i \oplus a_j}$，因此我们可以对卷积后的数组进行高维前缀和得到答案。
- 整体复杂度：$O(m2^m)$。

例题：AND-OR game

- 给定$A_1, A_2, \ldots, A_n$，$B_1, B_2, \ldots, B_m$，v为一变量，初始值为0。求我们可以通过下列操作得到的v的值
    - 选择A_i，将v更新为$v|A_i$;
    - 选择B_i，将v更新为$v\&B_i$。
- $n, m \leq 2^{16}$, $0 \leq A_i, B_i < 2^{16}$

例题：AND-OR game

- 对于存在性问题，我们也可以考虑卷积。
- 不妨假设$A_1 = 0$，$B_1 = 2^{20} - 1$。
- 不断交替进行or和and卷积，直至非零位置不再变化。
- 正确性：每一个元素可以由$O(\log^2 A)$次and/or得到（可能可以证明得更好）。

例题：RNG and XOR

- 给定一个$[0, 2^n-1]$的概率分布p。给定一个变量x，初始为0。考虑如下过程:
  - 依照概率分布p抽取一个数a，将x更新为$x \oplus a$。
- 当x变为i时则停止该过程。对于每个$0 \le i < 2^n$，求期望操作次数。
- $n \le 16$。

例题：RNG and XOR

- 列举线性方程组：对于所有的$i \neq 0$,

$$E_i = \sum_{j=0}^{2^n-1} p_j E_{i \oplus j} + 1$$

- 上述等价于序列p与E做xor卷积得到$E-1$(除了E_0那一项之外)。即

$$(p_0, p_1, \ldots, p_{2^n-1}) \oplus (E_0, E_1, \ldots, E_{2^n-1}) = (E_0' - 1, E_1 - 1, \ldots, E_{2^n-1}$$

- 由于$\sum_{i=0}^{2^n-1} p_i = 1$，可以解得$E_0' = 2^n$（不改变求和）
- 因此，我们只需求E满足$(p_0 - 1, p_1, \ldots, p_{2^n-1}) \oplus E = (2^n - 1, -1, -1, \ldots, -1)$。
- 这是FWT的逆问题。考虑做变换将卷积变成逐项相除。
- 唯一出现问题的一点在于p数组做完异或变换（FWT）之后首项是0，不能还原出E数组FWT的首项值。
- 不过，我们可以待定第一项的系数，通过逆变换通过$E_0 = 0$解得第一项系数。

例题：Binary table

- 给定一个$n \times m$的01矩阵A，求通过将行flip或者将列flip操作，表格中1的个数能达到的最小值。
- $n \leq 20, m \leq 10^5$。

例题：Binary table

- 将每一列看成一个二进制数，a_i为二进制数等于i的个数，b_i为min(i的popcount,n-i的popcount)。
- 则反转行的集合为$mask$时，答案为$\sum_{i=0}^{2^n-1} a_i b_{mask \oplus i}$。
- 因此，我们只需要对每一个$mask$求出上述值即可，这就是异或卷积。

第四小节内容

- 期望的线性性质

期望的线性性质

- 期望线性性质本身很简单：令$X_1, X_2, \ldots, X_n$为随机变量，则

$$E[\sum_{k=1}^{n} X_k] = \sum_{k=1}^{n} E[X_k].$$

- 但是最难的一点是怎么将一系列随机变量拆分成多个随机变量之和。

例题：Random Isolation

- 给定树T，我们执行下列的操作直至所有连通块的大小不超过k：
  - 随机从所有处于大小大于k的连通块的顶点中随机选择一个并删除该顶点。
- 求删除的期望次数。
- $n, k \leq 300$。

例题：Random Isolation

- 令X_u为u被删除的indicator，即被删除了是1，未被删除是0。则为了求期望，我们只需要求每一个顶点u被删除的概率。
- 对于每一个顶点u，我们将其作为根节点，其被删除的事件等价于我们选了一堆子节点，删除了这些子节点之后再选了顶点u。假设我们选择了顶点$u_1, u_2, \ldots, u_k$和u，删除u之前的连通块大小为m，这种事件发生的概率为$\frac{k!}{m(m-1)\ldots(m-k+1)}$。这是一个背包问题，可以在$O(n^2)$的时间内处理。

例题：Expected Destruction

- 给定一个包含n个数的集合S，每一个元素都是1到m的整数。每一轮我们考虑如下操作：
  1. 等概率随机选择一个数$x \in S$，并将x从S中去掉;
  2. 如果$x + 1 \leq m$, 则$x + 1$并入至S中。
- 求S被删到空集需要到期望操作步数。
- $n, m \leq 500$。

例题：Expected Destruction

- 假设S为多重集合，则答案是$n(m+1)-sum$;
- 多加的部分为两个间隙消失的部分。假设$S=\{x_1,x_2,\dots,x_n\}$，则我们对于每一个间隙$[x_i,x_{i+1}]$，我们只需要算出来该间隙左端点1/2的概率+1，右端点1/2的概率+1（超过m就不加），最终碰到一起的位置期望是多少即可。（最终答案由期望的线性性质得到）。

第四小节内容

- 图上的一些计数问题：
- 带标号树与Prufer序列；
- 图的生成树计数：矩阵树定理；
- 欧拉回路计数：BEST定理。
- 有向无环图不相交路径数：LGV引理；

Prufer序列

- Prufer序列给出了带标号树到$[n]^{n-2}$的一个一一映射：
- 每一轮选择标号最小的顶点，删除他并记录他连接的顶点编号。
- 从序列到树的构造：
  - 由Prufer序列，每一个顶点的度数为序列中出现次数+1。
  - 找到度数为1的最小编号节点，将该节点与prufer序列第一个值相连；这两个顶点度数−1，重复操作。
- Prufer序列推论：给定度数序列$d_1, d_2, \ldots, d_n$，假设$\sum_{i=1}^{n} d_i = 2n-2$，树的方案数为$\frac{(n-2)!}{\prod_{i=1}^{n}(d_i-1)!}$。

矩阵树定理

- 给定一个无向图$G = (V, E)$，求生成树个数。
- 令$L = D - A$，其中$D = \mathrm{diag}(d_1, d_2, \ldots, d_n)$为一个对角矩阵，第$i$行$i$列第元素$d_i$为顶点$i$的度数，$A$为邻接矩阵。
- 生成树个数为Laplacian矩阵L去掉某一行某一列（相同index）后的行列式的值。
- 带权图/带权生成树几乎一致。

矩阵树定理（有向图版本）与BEST定理

- 给定一个有向图$G=(V,E)$，求底层图为树，所有边指向根r的生成子图个数$t^{\mathrm{root}}(G,r)$。
- 令$L=D-A$，其中$D=\mathrm{diag}(d_1,d_2,\ldots,d_n)$为一个对角矩阵，第$i$行$i$列第元素$d_i$为顶点$i$的出度，$A$为邻接矩阵。
- 以r为根的个数为Laplacian矩阵L去掉第r行r列之后的行列式的值。
- 如果改成入度：所有边指向远离根的方向的生成子图个数。
- 欧拉回路个数：$t^{\mathrm{root}}(G,r)\cdot\prod_{i=1}^{n}(d_i-1)$，其$r$为任意一个顶点