生成函数初步与杂谈

张昕渊

November 10, 2023

普通生成函数与指数生成函数

- 形式幂级数：$f = \sum_{n=0}^{+\infty} a_n x^n$。（我们只是形式化的用它，不考虑收敛域等问题）
- 我们通常令$[x^n]f$代表f在第n项的系数。
- 给定序列$a_0, a_1, a_2, \ldots$，我们通常考虑以下两种形式的生成函数
  - 普通生成函数（Ordinary Generating Function, OGF）：

$$f = \sum_{n=0}^{+\infty} a_n x^n$$

  - 指数生成函数（Exponential Generating Function，EGF）：

$$f = \sum_{n=0}^{+\infty} \frac{a_n x^n}{n!}$$

- 为什么要研究生成函数：序列对应的生成函数形式简单/易于得到，可以通过形式幂级数的快速运算加速计算。

常见的形式幂级数运算

- 令形式幂级数$f=\sum_{n=0}^{+\infty}a_nx^n$。
- 求导：$f'=\sum_{n=1}^{+\infty}na_nx^{n-1}=\sum_{n=0}^{+\infty}(n+1)a_{n+1}x^n$。
- 积分：$\int f=\sum_{n=0}^{+\infty}\frac{a_nx^{n+1}}{n+1}+C$，其中$C$为任意常数。
- 多项式求逆：假设$[x^0]f\neq 0$，则存在唯一多项式f^{-1}满足$f\cdot f^{-1}=1$。如果只需要算前n项，我们可以通过倍增计算：
  - 假设我们已经求得了前m项，即找到了f_m^{-1}满足

    $$f\cdot f_m^{-1}\equiv 1 \mod x^m,$$

    现在我们需要求f_{2m}^{-1}。由定义，

    $$f_m^{-1}-f_{2m}^{-1}\equiv 0 \mod x^m,$$

    两边平方后乘以f可知，

    $$f(f_m^{-1})^2-2f_m^{-1}+f_{2m}^{-1}\equiv 0 \mod x^{2m},$$

    解得$f_{2m}^{-1}\equiv 2f_m^{-1}-f(f_m^{-1})^2 \mod x^{2m}$，可由FFT得到。

牛顿迭代法

- 令形式幂级数$f = \sum_{n=0}^{+\infty} a_n x^n$。
- 给定一个函数f（从形式幂级数到形式幂级数的函数），求多项式g满足$f(g(x)) \equiv 0 \mod x^n$。
- 类似的，我们考虑倍增求法：
  - 假设已知模x^m的多项式g_m，求g_{2m}。
  - 将f在$g_m(x)$处展开：

$$f(g_{2m}(x)) = \sum_{i=0}^{+\infty} \frac{f^{(i)}(g_m(x))}{i!}(g_{2m}(x) - g_m(x))^i,$$

    其中$f^{(i)}$为f的第i阶导数。
  - 令$g(x) - g_m(x) \equiv 0 \mod x^m$，则上述式子两边模$x^{2m}$后，只剩$f$本身和其一阶导数，整理得$g_{2m} \equiv g_m - \frac{f(g_m(x))}{f'(g_m(x))} \mod x^{2m}$。

多项式基本操作

- 牛顿迭代法：$g_{2m} \equiv g_m - \frac{f(g_m(x))}{f'(g_m(x))} \mod x^{2m}$。
- 多项式求逆（回顾）：
  - 令$f(P) = 1/P - h(x)$，则$f(g(x)) = 0$当且仅当g是h的逆。
  - $f'(P) = -1/P^2$，带入得

$$g_{2m} \equiv g_m + \frac{1/g_m - h}{1/g_m^2} = 2g_m - g_m^2 \cdot h \mod x^{2m}$$

- 多项式求exp（我们需要常系数a_0等于0，对ln而言我们需要常数项为1）：
  - 给定多项式h，$\exp h = 1 + \sum_{k=1}^{+\infty} \frac{h^k}{k!}$, $\ln h = \sum_{k=1}^{+\infty} \frac{(-1)^k h^k}{k}$。
  - 令$f(P) = \ln P - h(x)$，则$f(g(x)) = 0$当且仅当$g = \exp(h(x))$。
  - $f'(P) = 1/P$，带入得

$$g_{2m} \equiv g_m - \frac{\ln g_m - h}{1/g_m} = g_m \cdot (1 + h - \ln g_m) \mod x^{2m}$$

  - $\ln g$：$(\ln g)' = \frac{g'}{g}$，因此可以求$\frac{g'}{g}$后积分即可。

多项式基本操作

- 多项式幂次/开方：
  - 多项式k次幂：当常数项为1时，$\exp(k\ln f)$即可；否则，将多项式化为常系数为1的情形。
  - 多项式开方：我们仅考虑常数项为1的情形。类似上述，我们只需求$\exp(\ln f/k)$即可。
- 多项式取模：给定两个多项式f,g，求多项式Q,R满足

$$f\equiv R \mod g, f=g\cdot Q+R.$$

- 由于$f(x)=g(x)Q(x)+R(x)$，两边令$x\to 1/x$变为

$$x^{\mathrm{deg(f)}}f(1/x)=x^{\mathrm{deg(f)}}g(1/x)Q(1/x)+x^{\mathrm{deg}(f)}R(1/x)$$

- 注意到$deg(R)<deg(g)$，因此两边模$x^{deg(f)-deg(g)+1}$，有

$$x^{\mathrm{deg(f)}}f(1/x)\equiv x^{\mathrm{deg(g)}}g(1/x)x^{\mathrm{deg(Q)}}Q(1/x) \mod x^{deg(f)-deg(g)+1}$$

- 多项式求逆即可。

多项式基本操作

- 多项式多点求值：给定一个n次多项式$f(x)$以及n个点$x_1, x_2, \ldots, x_n$，求$f(x_1), f(x_2), \ldots, f(x_n)$。
- 考虑分治：
  - 考虑$x_1, x_2, \ldots, x_{n/2}$，求$f$在这一系列点上的取值等于$f \bmod \prod_{i=1}^{n/2}(x - x_i)$在这一系列点上的取值。
  - 通过多项式取模可以将问题转化为两个规模为$n/2$的子问题，时间复杂度$O(n \log^2 n)$。

普通生成函数(OGF)

- OGF: $a = (a_0, a_1, a_2, \ldots) \to f = \sum_{n \geq 0} a_n x^n$。
- 一些（简单）例子：
  - $a = (1, 1, 1, \ldots)$： $f = \sum_{n=0}^{+\infty} a_n x^n = \sum_{n=0}^{+\infty} x^n = \frac{1}{1-x}$。
  - 斐波那契数列$a_{n+2} = a_{n+1} + a_n$：两边乘以x^{n+2}后对$n = 0$到正无穷求和，可知

$$\sum_{n=0}^{+\infty} a_{n+2} x^{n+2} = x \sum_{n=0}^{+\infty} a_{n+1} x^{n+1} + x^2 \sum_{n=0}^{+\infty} a_n x^n$$

左边是$f - 1 - x$，右边是$x(f - 1) + x^2 f$，整理可得$f = \frac{1}{1-x-x^2}$。

普通生成函数(OGF)

- 复杂一点的例子：Catalan数（长度为$2n$的合法括号序列数）。
- 递推式$a_{n+1}=\sum_{i=0}^{n} a_i a_{n-i}$，$a_0=1$。
- 两边乘以x^{n+1}后求和：

$$\sum_{n=0}^{+\infty} a_{n+1}x^{n+1}=\sum_{n=0}^{+\infty} x^{n+1}\sum_{i=0}^{n} a_i a_{n-i}=x(\sum_{n=0}^{+\infty} a_n x^n)^2,$$

- 因此，OGF满足$f-1=xf^2$，这是一个二次方程。可以解得$f=\frac{1-\sqrt{1-4x}}{2x}$。我们舍弃了正项，原因是常数项需要是1。
- 由二项公式：$(1+x)^{\alpha}=\sum_{n\geq 0}\binom{\alpha}{n}x^n$，其中$\binom{\alpha}{n}=\frac{\alpha(\alpha-1)\ldots(\alpha-n+1)}{n!}$（$\alpha$可以是实数）。
- $\sqrt{1-4x}=\sum_{n\geq 0}\binom{1/2}{n}(-4x)^n=1-\sum_{n\geq 1}\frac{2}{n}\binom{2n-2}{n-1}x^n$，
- $\frac{1-\sqrt{1-4x}}{2x}=\sum_{n\geq 0}\frac{1}{n+1}\binom{2n}{n}x^n$。

指数生成函数（EGF）

- EGF: $a = (a_0, a_1, a_2, \ldots) \to f = \sum_{n\geq 0} \frac{a_n}{n!} x^n$。更加适用于一些带有组合数的卷积。
- 简单例子：$a = (1, 1, 1, \ldots)$，$f = \sum_{n\geq 0} \frac{x^n}{n!} = e^x$。
- 复杂一点的例子：Bell数（第二类stirling数的行和，n个元素划分为若干个集合的划法）。
- $a_n = \sum_{k=0}^{n-1} \binom{n-1}{k} a_k$。两边乘以$\frac{x^n}{n!}$可得

$$\frac{a_n}{n!}x^n = \frac{x^n}{n!}\sum_{k=0}^{n-1} \frac{(n-1)!}{(n-k-1)!k!} a_k = \frac{x}{n}\sum_{k=0}^{n-1} \frac{x^{n-k-1}}{(n-k-1)!} \frac{a_k}{k!},$$

因此，$\sum_{n\geq 1} n \cdot \frac{a_n}{n!} x^{n-1} = \sum_{n\geq 1} \sum_{k=0}^{n-1} \frac{x^{n-k-1}}{(n-k-1)!} \frac{a_k}{k!}$。

- 左边是EGF的导数，右边是EGF乘以e^x，因此有常系数微分方程$f' = fe^x$，解得$f = \exp(e^x + C)$，其中C为常数。考虑常数项可知$C = -1$。

序列操作与生成函数操作

- 我们下面考虑两个序列a, b以及对应的OGF/EGF为f, g。
- 序列相加：OGF/EGF均为$f+g$。
- 序列卷积：
  - $c_n = \sum_{k=0}^{n} a_k b_{n-k}$：这对应OGF相乘。
  - $c_n = \sum_{k=0}^{n} \binom{n}{k} a_k b_{n-k}$：这对应EGF相乘。

$$\sum_{i=0}^{+\infty}\sum_{j=0}^{+\infty}\frac{a_i x^i}{i!}\frac{b_j x^j}{j!} = \sum_{n\geq 0}\frac{x^n}{n!}\sum_{i=0}^{+\infty}\frac{n! a_i b_{n-i}}{i!(n-i)!} = \sum_{n\geq 0}\frac{x^n}{n!}\sum_{i=0}^{+\infty}\binom{n}{i} a_i b_{n-i}.$$

- 序列平移：$c_n = a_{n+k}$。
  - OGF: $f \to \frac{f - a_0 - a_1 x - \ldots - a_{k-1}x^{k-1}}{x^k}$。
  - EGF: f求k次导数。
- 序列乘以n：OGF/EGF均为$f \to xf'$。
- 前缀和序列：$c_n = \sum_{k=0}^{n} a_k$。对于OGF而言，$f \to \frac{1}{1-x} f$。

第一类斯特林数

- n个元素划分为k个圆排列的方案数/长度为n，圈的个数为k的排列个数。
- 考虑长度为n的圆排列个数$a_n = (n-1)!$，考虑其EGF

$$f = \sum_{n \geq 1} \frac{x^n}{n} = -\ln(1-x).$$

- 假设每个圆排列是有标号的，则方案数为

$$\sum_{\substack{i_1, i_2, \ldots, i_k \\ i_1 + i_2 + \ldots + i_k = n}} \frac{n!}{i_1! i_2! \ldots i_k!} a_{i_1} a_{i_2} \ldots a_{i_k}.$$

- 这是EGF f的k次方，去除圆排列的标号后为$\frac{1}{k!} f^k$；
- 因此固定k，求前n项可以在$O(n \log n)$的时间内计算。

排列的期望圈数

- 均匀选取一个排列，求排列的圈的个数期望值。
- 做法1：利用线性期望的线性性质，我们只需要统计出现某一个长度为k的特定圈的概率以及所有长度为k的圈的个数即可。
- 做法2：考虑利用生成函数。k个圈的方案数为$[x^n]\frac{1}{k!}f^k$。因此答案为$[x^n]\sum_{k=1}^{+\infty}\frac{1}{(k-1)!}f^k=[x^n]fe^f$。由于$f=-\ln(1-x)$，因此$fe^f=-\frac{\ln(1-x)}{1-x}$。由于$-\ln(1-x)=\sum_{n\geq 1}\frac{x^n}{n}$，因此$[x^n]-\frac{\ln(1-x)}{1-x}=\sum_{k=1}^{n}\frac{1}{k}$（前缀和变换）。

例题：排列

- 给定n，求所有圈长均为质数的排列数。
- $n \leq 10^5$。

例题：排列

- 和上述讨论类似，只不过n非质数时，$a_n = 0$。
- 圈的个数为k个的方案数为$n![x^n]\frac{1}{k!}f^k$。
- 对k求和可知EGF为e^f。

例题：无向图

- 给定n，求n个带标号点的2-正则图个数。
- $n \leq 10^5$。

例题：无向图

- 做法1：令a_n为答案，有如下递推式：$a_n = \sum_{k=1}^{n} \binom{n-1}{n-k} a_{n-k} b_k$。其中$b_k$是$k$个点的2-正则连通图的个数（当$k \ge 3$时为$(k-1)!/2$，否则为0）。这是一个半在线卷积的形式，可以分治FFT在$O(n \log^2 n)$的时间内解决。
- 做法2：类似于上述，EGF为e^f，其中f是序列b的EGF，即$f = \sum_{n \ge 3} \frac{1}{2n} x^n$。

例题：二部图

- 求n个带标号点的二部图方案数。
- $n \leq 1000$。

例题：二部图

- 我们首先考虑二染色的二部图方案数（即每个二部图的权重为$2^{\text{连通分支个数}}$，加权求和）：$a_n = \sum_{k\geq 0}\binom{n}{k}2^{k(n-k)}$，假设其EGF为$f$。
- 加入连通性约束后的带权求和：其EGF为g满足$g = \ln f$；
- 对于连通图而言，二染色方案数为2，因此连通二部图方案数的EGF满足$h = g/2$；
- 最终去掉连通性后EGF为$e^h = e^{\ln f/2} = \sqrt{f}$。

例题：Descents of Permutations

- 给定n和k，求满足下列条件的长度为n的排列$p = [p_1, p_2, \ldots, p_n]$个数：
  - $p_i > p_{i+1}$当且仅当$i|k$。
- $n \leq 5 \cdot 10^5$。

例题：Descents of Permutations

- 不妨假设$n = km$是k的倍数（非倍数情形类似，留作课后思考）。
- 问题转化为将n划分到可区分的m组，每一组k个数，前一组的最大值大于后一组最小值的方案数。
- 对最后一个条件容斥，最终答案如下：

$$\sum_{\substack{i_1,i_2,\dots,i_l \\ i_1+i_2+\dots+i_l=m}} (-1)^{m-l} \frac{(km)!}{(ki_1)!(ki_2)!\dots(ki_l)!}.$$

- 对这个式子，我们可以考虑建立递推式后分治fft，也可以考虑EGF。
- 考虑多项式$f = \sum_{k=1}^{+\infty} \frac{(-1)^i x^{ki}}{(ki)!}$，则固定$\ell$后的EGF为$(-1)^l f^l$，对$\ell$求和有最终多项式为$\frac{f}{1+f}$。

例题：Many Easy Problems

- 给定一棵树$T=(V,E)$。对于顶点集合$S\subseteq V$，令$f(S)$为包含S的最小子树大小。对于所有的$1\le k\le n$，求$\sum_{\substack{S\subseteq V\\|S|=k}} f(S)$。
- $n\le 2\cdot 10^5$。

例题：Many Easy Problems

- 对于每一个顶点u以及集合S，u不会贡献到S中当且仅当S在T去掉u的某个连通分支中。
- 对于每个节点，答案为$\binom{n}{k} - \sum_{i=1}^{d} \binom{a_i}{k}$，其中$a_1, a_2, \ldots, a_d$为去掉顶点$u$后的连通分支大小。
- 因此，我们只需要计算如下问题即可：
  - 给定$c_1, c_2, \ldots, c_n$，对于所有的$1 \le k \le n$，求$\sum_{i=1}^{n} c_i \cdot \binom{i}{k}$。
- 注意到$\frac{1}{k!} \cdot \sum_{i=1}^{n} \frac{c_i \cdot i!}{(i-k)!}$，是多项式$f = \sum_{i=1}^{n} c_i \cdot i! x^i$与多项式$g = \sum_{i=0}^{n} \frac{x^{-i}}{i!}$的卷积下$x^k$的系数。

例题：The Child and Binary Tree

- 给定一个数集$S = \{s_1, s_2, \ldots, s_n\}(s_i \geq 1)$，求有多少个二叉树满足点权均属于$S$，点权和为$k$。对于$1 \leq k \leq m$均给出答案。
- $1 \leq n, m \leq 10^5$。

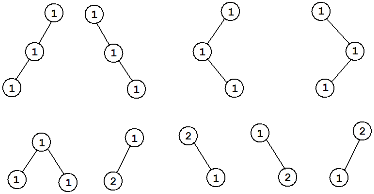

Figure: $S = \{1, 2\}, k = 3$

例题：The Child and Binary Tree

- 递推式：$a_n = \sum_{s\in S}\sum_{k=0}^{n-s} a_k a_{n-s-k}$。
- 考虑两边乘以x^n并求和，可知

$$\sum_{n\geq 1} a_n x^n = \left(\sum_{s\in S} x^s\right)\left(\sum_{n\geq 0} a_n x^n\right)^2$$

- 因此，a_n的OGF满足$f-1=gf^2$，解得$f=\frac{1-\sqrt{1-4g}}{2g}$。

例题：Inversion

- 求长度为n且逆序对个数恰好为k的排列个数。
- $n \le 10^9$，$k \le 10^5$。

例题：Inversion

- 问题等价于求$a_1, a_2, \ldots, a_n$使得$a_i \le i$，$\sum a_i = k$。
- 等价于下列多项式的第k项：

$$
\begin{aligned}
&(1+x)(1+x+x^2)\ldots(1+x+x^2+\ldots+x^n) \\
&= \prod_{i=1}^{n+1} \frac{1-x^i}{1-x} \\
&= (1-x)^{-(n+1)} \prod_{i=1}^{n+1} (1-x^i)
\end{aligned}
$$

- 由广义二项式定理可知$(1-x)^{-(n+1)}$的前k项。因此，我们只需要求$\prod_{i=1}^{n+1}(1-x^i)$的前k项即可。
- 不妨假设$n \ge k$，此时我们只需要求

$$
\prod_{i\ge 1}(1-x^i) = 1 + \sum_{i\ge 1}(-1)^i \left(x^{\frac{i(3i+1)}{2}} + x^{\frac{i(3i-1)}{2}}\right)
$$

的前k项即可。

杂谈：类欧几里得算法

- 求$f(a,b,c,n)=\sum_{i=0}^{n}\lfloor\frac{ai+b}{c}\rfloor\ (a,b\geq 0,c>0)$。
- 首先将问题转化为$a<c,b<c$:

$$f(a,b,c,n)=f(a \mod c, b \mod c, c, n)+\sum_{i=0}^{n}\left\lfloor\frac{a}{c}\right\rfloor\cdot i+\left\lfloor\frac{b}{c}\right\rfloor$$

- 令$m=\left\lfloor\frac{an+b}{c}\right\rfloor$，则

$$\begin{aligned}f(a,b,c,n)&=\sum_{i=0}^{n}\sum_{j=0}^{m-1}[j<\left\lfloor\frac{ai+b}{c}\right\rfloor]\\&=\sum_{i=0}^{n}\sum_{j=0}^{m-1}[i\geq\frac{cj+c-b}{a}]\\&=nm-f(c,c-b-1,a,m).\end{aligned}$$

- 递归到base case $c=1$即可。单独看a,c，每一轮操作是$(a,c)\rightarrow(c,a \mod c)$，这是类似于欧几里得算法的递归。

例题：MetaStock

- 给定两只股票，第一支股票每天以1/2的概率增加x元，1/2的概率不变；第二支股票每天以1/2的概率增加y元，1/2的概率不变。
- 求最大的整数w使得随机事件“n天后第一支股票的价格比第二支股票的价格至少高w元”发生的概率至少为50%。
- 10^5组，$x, y, n \leq 10^9$。

例题：MetaStock

- 非常自然的考虑两只股票的差值：1/4概率x, 1/4概率$x-y$, 1/4概率$-y$, 1/4概率0。
- 注意到这是关于$\frac{x-y}{2}$对称的分布（平移后为$u=\frac{x+y}{2}, v=\frac{x-y}{2}, -v=-\frac{x-y}{2}, -u=-\frac{x+y}{2}$。
- n天后的分布也是关于$\frac{n(x-y)}{2}$对称，因此我们只需要找到最大半整数$w \geq 0$使得$Au+Bv+C(-u)+D(-v)=w$且$A+B+C+D=n$即可。
- 等价于找到$A, B \in \mathbb{Z}$使得$|A|+|B| \leq n$, $A+B+n$是偶数，在这些条件下最小化$w=Au+Bv \geq 0$的值。
- 假设$A>0$，枚举A的奇偶性后问题转化为给定一段区间$[l, r]$，求$\min_{x \in [l,r]} Ax+y \mod B$；
- 这是一个经典问题，我们考虑二分答案，$\min \geq u$当且仅当$\sum_{x=l}^{r} \left\lfloor \frac{Ax+y}{B} \right\rfloor = \sum_{x=l}^{r} \left\lfloor \frac{Ax+y-u}{B} \right\rfloor$

杂谈：亚线性筛

- 这里我们介绍杜教筛。
- 杜教筛的主要思想是利用狄利克雷卷积，将求解函数f前缀和的问题变为求解更简单函数前缀和的问题。
- $f * g = h$: $\sum_{n=1}^{N} h(n) = \sum_{n=1}^{N} \sum_{d|n} f(d)g(n/d)$。
- $S_h(N) = \sum_{n=1}^{N} h(n) = \sum_{n=1}^{N} g(n)S_f(\lfloor \frac{N}{n} \rfloor)$。
- 结合整数分块，假设h的前缀和易于求解，我们可以在$\tilde{O}(n^{2/3})$的时间内解决。
- 一些简单的积性函数可以通过杜教筛处理：
  - $1 * \mu = id = (1, 0, 0, 0, \ldots)$;
  - $1 * \phi = (1, 2, 3, \ldots)$.

杂谈：随机算法（例题1）

- 给定平面上n个两两不同的点，找到一条直线穿过至少$n/4$个点。
- $n \leq 10^5$。

杂谈：随机算法（例题1）

- 随机两个点，过这两个点做一条直线，check即可。
- 如果存在一条好的直线，那么上述方案成功概率约为1/16。多次实验即可。

杂谈：随机算法（例题2）

- 给$n = 10^5$个数$a_1, a_2, \ldots, a_n$，他们的值介于$[1, 10^{18}]$。可以询问$n + 3000$次某个数a_i是否大于等于x。找到最大的数。交互不是适应性的（数组a固定）。
- $n \leq 10^5$。

杂谈：随机算法（例题2）

- 不妨直接假设所有数已经shuffle了。
- 对于第一个数，我们二分找到这个数的准确值；对于后续的数，我们查询当前的数是否大于当前最大值。如果不是，则说明这个数不可能成为最大值；否则，二分找到这个值的精确值。
- 第i个数是最大值的概率为$1/i$。因此期望二分次数为$\sum_{i=1}^{n} \frac{1}{i} = O(\log n)$。
- 查询次数为$n + O(\log n \log d)$。

杂谈：随机算法（例题3）

- 给定2^{k+1}个$[0, 4^k)$的整数，求两个不交非空区间异或和相等。
- $k \leq 16$。

杂谈：随机算法（例题3）

- 不妨假设所有数均不相等且不包含0。
- 我们考虑一个简单版的问题：找到两个非空区间，异或和相等。
- 非空区间个数：$\binom{2^{k+1}}{2} \approx 2^{2k+1}$，因此由鸽笼原理，至少有两个区间异或和相同。
- 不妨假设每个数恰好存在两个区间异或和等于这个值（可以证明，这是最坏情形）。我们随机选择$\theta(2^k)$个区间，由生日悖论，存在两个区间异或值相同的概率至少为$1/2$。
- 这两个区间不同的概率又为$1/2$，因此单次随机成功的概率为$1/4$，重复实验即可。
- 两个任意区间转化为不交区间：
  1. 两个区间相交：$[A,B],[B,C]$的形式，取A和C即可。
  2. 两个区间内包含1：$[A,B,C],[B]$的形式，取A和C即可。
  3. 两个区间内包含2:$[A,B]$，B的形式，此时A的异或和为0。若$len(A) > 1$，取A的第一个元素和剩下的元素即可。否则说明存在0，矛盾。

杂谈：随机算法（例题4）

- 给定n个长度为k的序列$a_1, a_2, \ldots, a_n$，求出任意10^5个(i, j)满足$a_{i,l} \neq a_{j,l}$对所有对$1 \leq l \leq k$都成立。
- $n \leq 10^5$，$k \leq 5$，$1 \leq a_i \leq 100$。

杂谈：随机算法（例题4）

- 将所有元素均匀随机映射到$\{1, 2, \ldots, 2k\}$。
- (i, j)满足映射前的条件：以1/2的概率满足映射后的条件；
- (i, j)不满足映射前的条件：不可能满足映射后的条件。
- 重复$\log n \log \frac{1}{\epsilon}$次，每个pair都会大概率成功一次。
- 复杂度：$O(((2k)^k + n) \log n \log \frac{1}{\epsilon})$。