Pólya 计数法的应用

南京外国语学校 陈瑜希

目录

Pólya 计数法的应用......1
目录......1
摘要......2
关键字......2
问题的提出......2
[例一]He's Circles SGU294......2
预备知识......3
Burnside 引理......4
Pólya 计数法......6
应用......9
[例二]Cubes UVA 10601......9
[例三]Transportation is fun SPOJ 419 SPOJ422......10
[例四] Isomorphism SGU282......11
总结......14
参考文献......14

摘要

在信息学竞赛中，我们会遇到许许多多的计数问题，很多问题看似困难，但熟练掌握 Pólya 计数法后，可以轻松解决。本文从一道信息学竞赛中出现的例题谈起，首先介绍了发现这题用普通计数法解决所遇到的困难，然后介绍了群、置换、置换群的基本概念、性质，并在此基础上引入 Burnside 定理，最后得出 Pólya 计数法，并给出证明。最后通过几道例题说明了 Pólya 计数法在信息学竞赛中的应用，并进行总结。

关键字

Burnside 定理 Pólya 计数法

问题的提出

[例一] He's Circles SGU294

有一个长度为 N 的环，上面写着'X'和'E'，问本质不同的环有多少种。(N 不超过 200000)。

[分析]

这个问题由于是一个环，许多未经过旋转时不同的方案，经过旋转之后就成了相同的方案，如果单纯的利用乘法原理来计算，无法排除这些相同的方案。如果想要用枚举法来做，需要枚举所有方案。枚举量不会低于本质不同的环的个数。事实证明，本质不同的环的个数是 2^n 级别的。对于 N=200000 的数据规模，答案会有 6 万多位，显然枚举是行不通的。

组合数学中，有一种计数法，可以很好的解决这类问题。

预备知识

下面，我们介绍一种重要的计数工具——Pólya 计数法。为了理解 Pólya 计数法，我们首先来看一下它所需要用到的概念。

群

给定一个集合 G={a,b,c,…}和集合 G 上的二元运算，并满足:

(a) 封闭性：$\forall a,b \in G, \exists c \in G, a*b=c$。

(b) 结合律：$\forall a,b,c \in G, (a*b)*c=a*(b*c)$。

(c) 单位元：$\exists e \in G, \forall a \in G, a*e=e*a=a$。

(d) 逆元：$\forall a \in G, \exists b \in G, a*b=b*a=e$，记 $b=a^{-1}$。

则称集合 G 在运算*之下是一个群，简称 G 是群。一般 a*b 简写为 ab。

置换

n 个元素 1,2,…,n 之间的一个置换$\begin{pmatrix} 1 & 2 & \cdots & n \\ a_1 & a_2 & \cdots & a_n \end{pmatrix}$表示 1 被 1 到 n 中的某个数 a_1取代，2 被 1 到 n 中的某个数 a_2取代，直到 n 被 1 到 n 中的某个数 a_n取代，且 $a_1,a_2,\dots,a_n$互不相同。

置换群

置换群的元素是置换，运算是置换的连接。例如：

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 1 & 2 & 4 \end{pmatrix}\begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 \end{pmatrix}=\begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 1 & 2 & 4 \end{pmatrix}\begin{pmatrix} 3 & 1 & 2 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix}=\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix}$$

可以验证置换群满足群的四个条件。

例1中置换群G={转0格、转1格、转2格、转3格……转(n-1)格}

$$\begin{pmatrix}1 & 2 & \cdots & n\\ 1 & 2 & \cdots & n\end{pmatrix}\begin{pmatrix}1 & 2 & \cdots & n\\ 2 & 3 & \cdots & 1\end{pmatrix}\begin{pmatrix}1 & 2 & \cdots & n\\ 3 & 4 & \cdots & 2\end{pmatrix}\begin{pmatrix}1 & 2 & \cdots & n\\ 4 & 5 & \cdots & 3\end{pmatrix}\cdots\cdots$$

$$\begin{pmatrix}1 & 2 & \cdots & n\\ n-1 & n & \cdots & n-2\end{pmatrix}$$

Burnside 引理

介绍

下面我们介绍Pólya计数法所要用到的一个引理——Burnside定理。

用$D(a_j)$表示在置换aj下不变的元素的个数。L表示本质不同的方案数。

$$L=\frac{1}{|G|}\sum_{j=1}^{s}D(a_j)$$

在例一中，对于N=4的情况。一共有4个置换：

$$\begin{pmatrix}1 & 2 & 3 & 4\\ 1 & 2 & 3 & 4\end{pmatrix}\begin{pmatrix}1 & 2 & 3 & 4\\ 2 & 3 & 4 & 1\end{pmatrix}\begin{pmatrix}1 & 2 & 3 & 4\\ 3 & 4 & 1 & 2\end{pmatrix}\begin{pmatrix}1 & 2 & 3 & 4\\ 4 & 1 & 2 & 3\end{pmatrix}$$

所有方案在置换a_1下都不变，$D(a_1)=16$

XXXX和EEEE在置换a_2下不变，$D(a_2)=2$

XXXX和EEEE以及XEXE与EXEX在置换a_3下不变，$D(a_3)=4$

XXXX和EEEE在置换a_4下不变，$D(a_4)=2$

计算出$L=\frac{1}{4}(16+2+4+2)=6$

证明

证明Burnside定理需要这样一个推论：

设c为$\mathcal{C}$中的一种着色，那么与c等价的着色数等于G中的置换个数除以

c 的稳定核中的置换个数。

定理 1：对于每一种着色 c ，c 的稳定核 G (c)是一个置换群，而且对 G 中任意置换 f 与 g，g*c=f*c 当且仅当 $f^{-1}\circ g$ 属于 G(c)。

证明：如果 f 和 g 都使 c 保持不变，则先 f 后 g 将使。保持不变，即(g○f)（c）＝c。于是，在合成运算下，G(c)具有封闭性。显然，单位元ι使得所有着色不变。如果 f 使 c 不变，那么 f^{-1} 也使 c 不变，于是 G(c)具有对逆的封闭性。由于满足置换群定义的所有性质，所以，G(c)是一个置换群。

假设 f*c=g*c 则

$$(f^{-1}\circ g)*c=f^{-1}*(g*c)=f^{-1}*(f*c)=(f^{-1}\circ f)*c=\iota*c=c$$

所以 $f^{-1}\circ g$ 使 c 不变，因此，$f^{-1}\circ g$ 属于 G(c)， 反之，假设 $f^{-1}\circ g$ 属于 G(c)，通过类似的计算可证得

f*c=g*c

作为定理 1 的一个推论，我们可以从已知的一种着色 c 出发，确定出在 G 的作用下不同的着色数。

推论 2：设 c 为 $\mathcal{C}$ 中的一种着色，那么与 c 等价的着色数

$$|\{f*c:f\in G\}|$$

等于 G 中的置换个数除以 c 的稳定核中的置换个数，即

$$\frac{|G|}{|G(c)|}$$

证明：设 f 是 G 中的一个置换，根据定理 1，满足

g*c=f*c

的置换 g 实际上就是

$$\{f\circ h:\ h\in G\ (c)\}$$

中的那些置换。由消去律，则从 f○h=f○h’得到 h=h’。于是，集合中

$\{f\circ h:\ h\in G\ (c)\}$ 的置换个数等于 G(c)中置换的个数。从而，对每个置换 f ，恰好存在|G(C)|个置换，这些置换作用在 c 上跟 f 有同样的效果。因为总共有|G|个置换，

所以，与 c 等价的着色数 $|\{f*c: f\in G\}|$ 等于 $\frac{|G|}{|G(c)|}$

有了这个推论，证明 Burnside 定理就是我们前面已多次用到的一些技巧的简单应用，即先采取两种不同的方式进行计数，然后使计数相等。究竟计什么数呢？我们要数使 f 保持 c 不变即 f*c＝c 的对偶(f,c)的个数。一种计数的方式是考察 G 中的每个 f ，并计算 f 保持着色不变的着色数，然后相加所有的量。因 D(f)是通过 f 保持着色不变的着色集，所以用这种方式计数得到

$$L=\frac{1}{|G|}\sum_{j=1}^{s}D(a_j)$$

另一种计数的方式是考察 $\mathcal{C}$ 中的每个 c ，计算满足 f *c=c 的置换 f 的个数，然后相加所有的量。对每种着色 $\mathcal{C}$ ，满足 f *c＝c 的所有 f 的集合就是我们所称的 c 的稳定核 G (c)。因此，每个 c 对和的贡献是

$$|G(c)|=\frac{|G|}{(\text{与 c 等价的着色数})}$$

于是，用这种方式计数，得到

$$\sum_{c\in C}\frac{|G|}{(\text{与 c 等价的着色数})}$$

但如果我们按等价类将着色归类，那么和式 $\sum_{c\in C}\frac{|G|}{(\text{与 c 等价的着色数})}$ 可以

简化。在同一等价类中，两种着色对和贡献了同样的量，每个等价类的总贡献是 |G| 。由于等价类的个数就是不等价的着色数 L ，所以，

$$\sum_{c\in C}\frac{|G|}{(\text{与 c 等价的着色数})}$$ 等于 L|G|

解出 L 即得

$$L=\frac{1}{|G|}\sum_{j=1}^{s}D(a_j)$$

Pólya 计数法

介绍

我们发现要计算 $D(a_j)$的值不是很容易，如果采用搜索的方法，总的时间规模为 O(nsp)。(n 表示元素个数，s 表示置换个数，p 表示格子数，这里 n 的规模是很大的) 下一步就是要找到一种简便的 $D(a_j)$的计算方法。先介绍一个循环的概念：

循环： [redacted]

记

称为 n 阶循环。每个置换都可以写若干互不相交的循环的乘积，两个循环

$$\begin{pmatrix}1 & 2 & 3 & 4 & 5\\ 3 & 5 & 1 & 4 & 2\end{pmatrix}=(13)(25)(4)$$

$(a_1a_2\ldots a_n)$和$(b_1b_2\ldots b_n)$互不相交是指 $a_i\neq b_j$, i,j=1,2,…,n。例如：

这样的表示是唯一的。置换的循环节数是上述表示中循环的个数。例如(13)(25)

(4)的循环节数为 3。

设 G 是 p 个对象的一个置换群，用 m 种颜色涂染 p 个对象，则不同染色方案为：

$$L=\frac{1}{|G|}(m^{c(g_1)}+m^{c(g_2)}+\cdots+m^{c(g_s)})$$

其中 $G=\{g_1,\ldots g_s\}$ $c(g_i)$为置换 g_i的循环节数$(i=1\ldots s)$

在例一中，我们给 N=4 的环标号：

1　2

4　3

构造置换群 $G'=\{g_1,g_2,g_3,g_4\}$, ｜G'｜ ＝4，令 g_i的循环节数为 $c(g_i)$ $(i=1,2,3,4)$

在 G'的作用下，其中

g_1表示转 0°，　即 $g_1=(1)(2)(3)(4)$　　$c(g_1)=4$

g_2表示转 90°,　即 $g_2=(4\ 3\ 2\ 1)$　　$c(g_2)=1$

g_3表示转 180°,　即 $g_3=(1\ 3)(2\ 4)$　　$c(g_3)=2$

g_4表示转 270°,　即 $g_4=(1\ 2\ 3\ 4)$　　$c(g_4)=1$

$2^{c(g1)}=2^4=16=D(a_1)$　　$2^{c(g2)}=2^1=2=D(a_2)$

$2^{c(g3)}=2^2=4=D(a_3)$　　$2^{c(g4)}=2^1=2=D(a_4)$

这就是所谓的 Pólya 定理。我们发现利用 Pólya 定理的时间复杂度为 $O(s\times p)$(这里 s 表示置换个数，p 表示格子数)，与前面得到的 Burnside 引理相比之下，又有了很大的改进，其优越性就十分明显了。Pólya 定理充分挖掘了研究对象的内在联系，总结了规律，省去了许多不必要的盲目搜索，把解决这类问题的时间规模降到了一个非常低的水平。

证明：

要得到在置换下稳定不动的方案，即把置换的每个循环节都染上相同的颜色，所以 $D(g_i)=m^{c(gi)}$

应用

Pólya 定理在信息学竞赛中有着许多应用实例。这些问题往往不能直接套用公式计算，而需要更细致的分析。下面我们通过几道例题来看一下信息学竞赛中出现的利用 Pólya 定理的计数问题。

[例二]Cubes UVA 10601

要求把正方体的 12 条棱染色，并且每种颜色的个数给定，求总方案数（旋转后相同的方案算一种）。

[分析]

这个问题是求对正方体的染色，且要求旋转后不变，很容易联想到 Pólya 计数法。

一个正方体共有 24 种旋转。根据这些不同的旋转方法，构造对应的关于边的置换群。

如果直接使用 Pólya 计数法的计算公式来做，不能保证它每种颜色使用的个数与题目要求的匹配。因此，需要一些改进。

回顾一下 Pólya 计数法的公式的推导过程：

根据 Burnside 定理，本质不同的方案数为在每个置换下稳定不动的方案的总合除以总置换数。

而要得到在置换下稳定不动的方案，即把置换的每个循环节都染上相同的颜色，所以 $D(g_i)=m^{c(g_i)}$

本题也是如此，也是要把置换的每个循环节都染上相同的颜色。

因此只需要对于每个置换，求出每个循环节的点的个数，用状态压缩动态规划求出每个循环节都染上相同的颜色，并且每种颜色的总和符合题目要求的方案总数，即可。

本题的难点在于给定了颜色总数的限制，使得不能直接套用公式计算，需要掌握Pólya计数法的来源，经过进一步分析，才能得到满足这个限制的解法。

而直接使用Pólya计数法还会有这样的困难：由于Pólya计数法的时间复杂度为$O(s_{\times}p)$ (这里s表示置换个数，p表示格子数)，有时置换个数非常多，因此，直接套用会超时。要优化时间复杂度，就要考虑降低需要枚举的置换数目。

[例三]Transportation is fun SPOJ 419 SPOJ422

给你一个2^a2^b的矩阵，在内存中的存放方式是先存第一行的，再存第二行的……每行也是从左到右存放。现在你想求它的转置矩阵（也是一样的储存方式），但是只能用交换操作，问需要交换多少步。

SPOJ419有100组输入数据

SPOJ422有400000组输入数据

[分析]

为了描述方便起见，先看一个例子：假设a=5，b=3。那么考虑元素(12,1)，用二进制表示是(01010,001)，那么它原来的地址就是01010 001，而新的地址就是001 01010。可见这个转置操作其实就是把每个元素的地址循环向右移动了b位。

考虑地址的循环节如果有k个，那么答案就是2^{a+b-k}。而地址循环节的个数可以看成是应用对地址的“循环移动b位”后本质不同的地址个数，这个可以用Pólya原理算出。

由于SPOJ422数据组数增多，我们要寻求更好的算法。

对于a和b，首先求出a和b的最大公约数g，答案可以写成：

$$\frac{a+b}{g}\bullet\sum_{i=1}^{\frac{a+b}{g}}(2^g)^{\gcd(i,\frac{a+b}{g})}$$

首先预处理出2的k次幂（因为Pólya只要用到2的若干次幂）。然后用筛法求素数“顺便”找出每个合数的最小质因数。

可见，问题的瓶颈是计算各个$\gcd(i,\frac{a+b}{g})$。

考虑到$\gcd(i,\frac{a+b}{g}) \mid \frac{a+b}{g}$，那么用 f(x)表示满足 $x=\gcd(i,\frac{a+b}{g})$的 i 的个数即可。

首先考虑怎么找出所有合法的 x。因为前面预处理了“每个合数的最小质因数”，现在可以在 O(logn)时间内找到$\frac{a+b}{g}$的所有质因数。有了质因数，再去找它的所有因子，就很容易了。

再看 f(x)怎么求。其实办法很简单，在刚才推算$\frac{a+b}{g}$的因子的时候，首先把$\frac{a+b}{g}$个数全部分配给 1，然后每次由数字 x 得到 xp 的时候，就把 f(x)的 1/p“分”给 f(xp)。这样，在找到所有因子的同时，也就找到了 f(x)的值。这样，代入前面公式计算，就很简单了。

这个题把它转化成 Pólya 计数模型并不困难，而难点在于转化后对算式的计算。置换的个数偏多而导致不能对每个置换都算其循环数，是 Pólya 计数的主要障碍，下面再看一个例题，在建立了 Pólya 计数模型之后，它的优化时间方法十分巧妙。

[例四]SGU282 Isomorphism[1]

染色图是无向完全图，且每条边可被染成 M 种颜色中的一种。两个染色图是同构的，当且仅当可以改变一个图的顶点的编号，使得两个染色图完全相同。问 N 个顶点，M 种颜色，本质不同（两两互不同构）的染色图个数（模质数 P）。

（1<=N<=53，1<=M<=1000，N<P<=10^9，时间限制 10s)

[1] 上海江苏 06 年省队选拔

[分析]

这样的问题，符合 Pólya 的适用范围。

放在这个问题中，置换群中的对象就是 C_n^2 条边，k 种颜色就是 M，G 就是由点的置换引起的边的置换的群。

同时我们知道，用 Pólya 定理直接计算可以在 O(ps)的时间解决问题，其中 s=|G|

但是对这个问题，|G|的数量达到 N!，当 N=53 时，这个数量相当惊人的，因此我们需要对问题进行更深入的分析。

考虑一个对点的置换 f=i→P[i]

那么，它对应的对边的置换就是 f'=(i,j)→(P[i],P[j])

我们考虑这两个不同类型的置换各自的循环节个数 c(f)与 c(f')的关系

我们可以得到如下结论：

假设点 i 与点 j 同属于一个长度为 L 的循环中，那么这样的边(i,j)组成的置换中循环节个数为$\left[\frac{L}{2}\right]$

假设点 i 与点 j 各属于长 L1 和 L2 的两个不同循环中，那么这样的边(i,j)组成的置换中循环节个数为（L1,L2），即 L1 与 L2 的最大公约数。

我们不妨设 $L_1 \ge L_2 \ge \ldots \ge L_m > 0$，且由 $L_1+L_2+\ldots+L_m=N$

所以 $L_1,L_2,\ldots,L_m$ 恰组成 N 的一种划分，由先前分析，每种划分对应的置换的循环节个数相同，为一个关于 $L_1,L_2,\ldots,L_m$ 的函数。

由于 N≤53，所以 N 的划分方案总数并不大，我们可以用回朔法枚举 N 的所有划分，然后对每一种划分，求出该划分对应的置换个数和每个置换的循环节个数。

循环节个数上面已经讨论过了，那么置换个数是多少呢？

假设已确定了 $L_1 \geq L_2 \geq \ldots \geq L_m > 0$，接下来就是将 1…N 这 N 个点分别放入这 m 个循环节中，满足第 i 个循环中恰含有 Li 个点，这一部分可以看作一个简单的组合问题，易知共 $\frac{N!}{L1!L2!\ldots Lm!}$ 有种不同方式。

又对于每一个循环，确定了第一个点，那么剩下各点的排列方式不同，其实对应的置换也是不同的，所以还要乘以 $(L_1-1)!(L_2-1)!\ldots(L_m-1)!$

又如果有 $L_i=L_{i+1}=\ldots=L_j$，那么每(j-i+1)!种方案又是重复的，所以还要除以(j-i+1)!

所以总的置换个数就是

$$\frac{N!}{L_1 L_2 \ldots L_m k_1! k_2! \ldots k_t!}$$

其中 t 表示共有 t 种不同的 L 值，每种值有 ki 个

这些值在枚举划分的时候都是可以计算的

因此，整个问题的答案可以根据不同类别的置换的循环数以及置换数算出。

下面讨论一下如何计算：

一部分是 M^{T1}，其中 T1 并不大，足以用一个 longint 表示，而 M^{T1} mod P 可以用倍增的思想在 log(T1)时间内计算，大家都应当很熟悉，这里不再赘述。

另一部分是 $T2^{-1}$，其中 T2 很大，而且是-1 次的，难道要分解质因数了吗？不是，注意问题的一个很不起眼但又很重要条件：P 是质数，且满足 N<P，这有什么用呢？

显然 $L_1,\ldots,L_m,k_1,\ldots,kt$ 都≤N，当然<P，因此 $L_1,\ldots,L_m,k_1,\ldots,k_t$ 均与 P 互质，所以 T2 也与 P 互质。另一方面，由于 P 是质数，由数论知识可知：

$T2^{p-1} \equiv 1 \pmod p$

$T2^{-1} \equiv T2^{p-2} \pmod p$

所以可以把 $T2^{-1}$ 转化为求 $T2^{p-2}$，就于 M^{T1} 求法相同了。

这个问题遇到了与上题同样的困难：置换的个数偏多而导致不能对每个置

换都算其循环数，而解决的方法，就是找出置换群中相似的置换，而不重复计算，这个去除冗余运算的方法在 Pólya 计数问题中经常用到。利用了这个思想后，对于每类相似置换个数的计算，也需要扎实的数学功底。

总结

本文从一道信息学竞赛中出现的例题谈起，介绍了 Pólya 计数法及其包含的数学思想以及它在信息学竞赛中的应用。Pólya 计数法不仅仅能解决许多计数问题，它的证明过程也是相当有意思的。灵活使用 Pólya 计数法，不仅仅需要熟练掌握此类问题的性质，还要有扎实的数学功底和分析问题能力。再次说明：数学方法是解决问题的工具，而分析问题能力是算法的源泉。

参考文献

[1]：《组合数学》

Richard A.Brualdi 著

冯舜玺 罗平 裴伟东 译

卢开澄 冯舜玺 校

[2]：2001 年 国家集训队论文 符文杰

[3]：2005-2007 年国家集训队作业