Exponentiation Commutativity 解题报告

安徽师范大学附属中学 罗哲正

1 试题来源

Codechef EXPCOMM

链接：https://www.codechef.com/problems/EXPCOMM。

2 试题大意

给定素数P，求满足$1 \le n, m \le P(P-1)$且$n^m \equiv m^n \pmod P$的数对(n, m)的个数，对质数M取模。

2.1 限制与约定

T组数据，$T \le 100, 2 \le P \le 10^{12}$.

时间限制：2s

空间限制：2GB

3 算法介绍

3.1 算法1

我们可以暴力枚举所有的n, m，并使用快速幂检验方程的正确性。

由于n, m的取值范围在$[1, P(P-1)]$，所以枚举的复杂度是$O\left(P^4\right)$。

于是总复杂度为$O\left(P^4 \log n\right)$，可以跑出$P \le 50$的数据。

3.2 算法2

首先若n,m中恰有一个是P的倍数，显然不满足，若$P|n$且$P|m$则满足，于是首先统计$P|(n,m)$的(n,m)数目，是$(P-1)^2$。

接下来我们只要考虑n,m不是P的倍数的情况。

指数不容易处理，我们可以使用原根把方幂转化为乘法。

令g为原根，设$n \equiv g^a \pmod P, m \equiv g^c \pmod P, n \equiv d \pmod{P-1}, m \equiv b \pmod{P-1}$。则$n^m \equiv m^n \pmod P \Rightarrow g^{ab} \equiv g^{cd}$。

而对于任意(a,d)，有$n \equiv g^a \pmod P, n \equiv d \pmod{P-1}$，有中国剩余定理得$n$在$P(P-1)$内有唯一值，对$m$同理。于是问题就变成了求多少对$(a,b,c,d)$满足$0 \le a,b,c,d < P-1$且$ab \equiv cd \pmod{P-1}$。

令$C(t) = \sum_{0 \le a,b < P-1} [ab \equiv t \pmod{p^e}]$，那么我们枚举所有的$a,b$就可以算出$C(t)$。那么显然$c^d$的取值和$a^b$相同的，于是答案就是$\sum C(t)^2$。

时间复杂度$O(P^2)$，能通过$P \le 10000$。

3.3 算法3

令$m = P-1$，$N(m)$表示满足条件的(a,b,c,d)对数，设$m = \prod_{i=1}^{k} p_i^{e_i}$为$m$的质因数分解。根据中国剩余定理，有

$$N(m) = \prod_{i=1}^{k} N(p_i^{e_i})$$

。

于是我们只要解决$N(p^e)$。

定义$C(t)$：

$$C(t) = [ab \equiv t \pmod{p^e}]$$

考虑$C(t)$如何计算。

设$t = p^j s$，$a = p^\alpha a', b = p^\beta b'$，所以就要满足$\alpha + \beta = j, a'b' \equiv s \pmod{p^{e-j}}$。如果枚举$\alpha,\beta$，对于任意$a' < p^{e-\alpha}$且$a'$不是$p$的倍数的$a'$，都会在$[0, p^{e-\alpha-\beta})$内有唯一的$b'$满足$a'b' \equiv s \pmod{p^{e-j}}$。

所以在$[0, p^{e-\beta})$内就有p^α个不同的b'满足。而在$[0, p^{e-\alpha})$内满足p不是其倍数的a'的取值共有$p^{e-\alpha-1}(p-1)$种。于是(a,b)的取值就有$p^\alpha * p^{e-\alpha-1}(p-1) = p^{e-1}(p-$

1)种，而$\alpha+\beta=j$的(α,β)共有$j+1$种取值。最终我们推出$C(t)=(j+1)p^{e-1}(p-1)$。

注意$C(0)=(e+1)p^{e-1}(p-1)+p^{e-1}$(考虑$\alpha+\beta\geq e$的情况)。

$$N(p^e)=\sum_{t=0}^{p^e-1}C(t)^2$$

注意到$C(t)$只跟j有关，于是直接枚举j：

$$N(p^e)=\left[(e+1)p^{e-1}(p-1)+p^{e-1}\right]^2+\sum_{j=0}^{e-1}p^{e-j-1}(p-1)\left[(j+1)p^{e-1}(p-1)\right]^2$$

这个式子已经可以暴力计算了，时间复杂度是$O\left(\sqrt{P}+\log p\right)$，但是这个式子或许有一些麻烦，我们考虑继续化简。

3.4 算法4

观察这个等式：

$$N(p^e)=\left[(e+1)p^{e-1}(p-1)+p^{e-1}\right]^2+\sum_{j=0}^{e-1}p^{e-j-1}(p-1)\left[(j+1)p^{e-1}(p-1)\right]^2$$

考虑这个式子与p的关系，可以发现前一部分p的次数不超过$2e$，后一部分每一项p的次数都不超过$3e-j$，于是可以猜想$N(p^e)$是关于p的$3e$次多项式。

那么我们可以举几个p和e的例子，并使用拉格朗日插值法求出系数，由于e是不超过$\log n$的，所以对每个e预处理出多项式即可。但如果我们观察插值的结果，很容易发现规律：

$$N(p^e)=p^{3e}+p^{3e-1}-p^{2e-1}$$

这个规律是可以直接暴力化简式子推出来的，所用到的知识仅仅是多项式求和，但在已知其为多项式的情况下使用拉格朗日插值法求出系数在信息学竞赛中无疑是一种聪明的做法。

接下来的部分就很简单了，$N(m)=\prod_{i=1}^{k}\left(p_i^{3e_i}+p_i^{3e_i-1}-p_i^{2e_i-1}\right)$，于是$ans=(P-1)^2+N(P-1)$。

我们只要对每个P分解质因子然后带入公式计算即可，分解质因子可以筛出质数后暴力分解。

处理所有数据的时间复杂度是$O\left(\sqrt{P}+T\left(\pi(\sqrt{P})+\gamma(P)\right)\right)$，其中$\gamma(P)$是$P$的质因子个数的渐进上界，大约与$(n\log n)^n$的反函数同阶，远小于$O(\log n)$。$\pi\sqrt{P}$可以采用筛法预处理质数做到。

4 总结

这是一道比较考察数论能力和数学推导能力的题目，代码量并不大。本题主要考差了选手对原根，中国剩余定理等数论知识的灵活运用，第一步把方案数转化成求(a,b,c,d)四元组的数目是本题的关键，转化为$ab\equiv t\pmod{P-1}$的方案数计算问题，再通过中国剩余定理分解成$\pmod{p^e}$的形式。接下来的部分则是数学推导，可以暴力化简式子多项式求和，也可以推出答案是多项式之后使用拉格朗日插值法求系数，枚举t中p的幂次是关键。

应该说这是一道小清新的数论题。