

大阶乘模质数的幂学习笔记

以下参考了[这篇博客](#)

求 $n! \bmod \{p^e\}$ (包括 p 的幂次和互质部分的模)，其中 p 为质数，可以在 $\mathcal{O}(pe + e \log_p n)$ 下完成 (不考虑大整数部分的复杂度)。

定义 $f(m) = \prod_{i=1, \text{gcd}(i,m)=1}^m i$ 那么有 $n! = f(n) \cdot p^{\lfloor \frac{n}{p} \rfloor} \cdot f(\lfloor \frac{n}{p} \rfloor) \cdot p^{\lfloor \frac{\lfloor \frac{n}{p} \rfloor}{p} \rfloor} \dots$ 也就是说我们可以通过 $\log_p n$ 次 f 的计算来得到答案。

设 $n = up + v$ ($0 \leq v < p$) 下面来计算 $f(n)$

$$f(n) = \left(\prod_{i=0}^{u-1} \prod_{j=1}^{p-1} (ip+j) \right) \cdot \prod_{j=1}^v (up+j)$$

其中

$$\prod_{i=1}^x (t+i) = \sum_{i=0}^{x-1} \frac{\begin{bmatrix} x+1 \\ i+1 \end{bmatrix} t^i}{\begin{bmatrix} x+1 \\ i+1 \end{bmatrix}}$$

因此

$$f(n) = \left(\prod_{i=0}^{u-1} \sum_{j=0}^{p-1} \begin{bmatrix} p \\ j+1 \end{bmatrix} (ip)^j \right) \cdot \sum_{j=0}^v \begin{bmatrix} v+1 \\ j+1 \end{bmatrix} (up)^j$$

$$= \left(\prod_{i=0}^{u-1} \sum_{j=0}^{\min\{p-1, e-1\}} \begin{bmatrix} p \\ j+1 \end{bmatrix} (ip)^j \right) \cdot \sum_{j=0}^{\min\{v, e-1\}} \begin{bmatrix} v+1 \\ j+1 \end{bmatrix} (up)^j$$

首先我们可以预处理 $\mathcal{O}(pe)$ 项的第一类斯特林数，右边这一项即可以 $\mathcal{O}(e \log_p n)$ 的复杂度计算。我们对左边继续化简

$$= \left(\begin{bmatrix} p \\ 1 \end{bmatrix} \right)^u \cdot \prod_{i=0}^{u-1} \sum_{j=0}^{\min\{p-1, e-1\}} \frac{\begin{bmatrix} p \\ j+1 \end{bmatrix}}{\begin{bmatrix} p \\ 1 \end{bmatrix}} (ip)^j$$

$$= \left(\begin{bmatrix} p \\ 1 \end{bmatrix} \right)^u \cdot \prod_{i=0}^{u-1} \left(1 + \sum_{j=1}^{\min\{p-1, e-1\}} \frac{\begin{bmatrix} p \\ j+1 \end{bmatrix}}{\begin{bmatrix} p \\ 1 \end{bmatrix}} (ip)^j \right)$$

左边可以把 $\log_p n$ 次计算的 u 加起来，通过快速幂计算一次即可。这部分的复杂度是 $\mathcal{O}(\log^2 n)$

下面我们证明，右边是关于 u 的一个 $2e-2$ 次多项式：

设 $g(i, x) = 1 + \sum_{j=1}^t a_j (ix)^j$ 关于 x 取 $\log$ 为

$$\sum_{j=1}^{+\infty} b_j (ix)^j$$

那么有

$$\log \prod_{i=0}^{u-1} g(i, x) = \sum_{i=0}^{u-1} \log g(i, x)$$

$$= \sum_{j=1}^{+\infty} b_j \left(\sum_{i=0}^{u-1} i^j \right) x^j$$

$$= \sum_{j=1}^{+\infty} b_j s_j(u) x^j$$

其中 $s_j(u)$ 是一个关于 u 的 $j+1$ 次多项式

$$\prod_{i=0}^{u-1} g(i, x) = \exp \left(\sum_{j=1}^{+\infty} b_j s_j(u) x^j \right)$$

$$= \prod_{j=1}^{+\infty} \exp \left(b_j s_j(u) x^j \right)$$

观察容易发现，结果多项式的第 i 项的系数是 u 的一个 $2i$ 次多项式 $t_i(u)$ (最高次在 $j=1$ 取到)。即

$$\log \prod_{i=0}^{u-1} g(i, x) = \sum_{i=0}^{+\infty} t_i(u) x^i$$

将 $x=p$ 代入，有

$$\log \prod_{i=0}^{u-1} g(i, p) = \sum_{i=0}^{e-1} t_i(u) p^i$$

因此上式是一个关于 u 的 $2e-2$ 次多项式。

考虑使用插值计算。首先可以用 $\mathcal{O}(pe)$ 的时间预处理出 $g(0, p) \sim g(2e-2, p)$ 的值，然后使用拉格朗日插值来计算。设 $h(u) = \prod_{i=0}^{u-1} g(i, p)$ 那么

$$h(u) = \sum_{i=0}^{2e-2} h(i) \prod_{j=0, j \neq i}^{2e-2} \frac{u-j}{i-j}$$

其中分母部分的阶乘逆元可以预处理，而分子部分也可以用前缀后缀积的技巧 $\mathcal{O}(1)$ 查询。这样的总复杂度为 $\mathcal{O}(e \log_p n)$ 但是这里存在一个小问题，分母中可能有 p 因此我们必须将和 p 互质的部分和 p 的幂次分开处理。分母部分可以预处理问题不大，但是如果暴力试除分子，最坏可以达

到 $O((\log_p)^2 n)$ 的复杂度，在 n 恰为 p 的幂次时取到。

首先我们证明，分母部分含 p 的个数至多为 $(2e-2)!$ 中 p 的个数。注意到分母部分的绝对值等于 $\frac{(2e-2)!}{(2e-2 \choose i)}$ 因此上面的结论是显然的。因此，分母部分 p 的个数为 $O(\frac{e}{p})$

容易发现 $h(u)$ 与 p 互质，因此至少存在一个 i 使得 $\prod_{j=0, j \neq i}^{2e-2} \frac{u-j}{i-j}$ 与 p 互质。也就是说，除了这个 $u-i$ 以外，剩下的所有 $u-j$ 中 p 的个数的和为 $O(\frac{e}{p})$ 而对于这个 $u-i$ 来说，显然我们只要分解出 $e + O(\frac{e}{p})$ 个 p 就没有再分解下去的必要了。这样这部分的复杂度就降为了 $O(e \log_p n)$

总复杂度 $O(pe + e \log_p n)$

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:intrepidword:zhongzihao:big_fac_mod_prime_pow

Last update: 2021/04/01 21:34

