多项式同余方程

主讲人：邓哲也

多项式同余方程

这节课我们来探讨如何求解 $f(x) \equiv 0 \pmod{m}$ 的同余方程。

其中 $f(x)$ 是次数大于 1 的整系数多项式。

例如：

$$2x^3 + 7x - 4 \equiv 0 \pmod{200}$$

多项式同余方程

注意到，若 m 有质因子分解 $m = p_1^{a1}p_2^{a2}\cdots p_k^{ak}$

那么求解 $f(x) \equiv 0 \pmod{m}$ 就等价于求解同余方程组：

$$f(x) \equiv 0 \pmod{p_i^{ai}}, \quad i = 1, 2, \cdots, k.$$

只要解出这 k 个同余方程，就可以利用中国剩余定理求出模 m 的解。

多项式同余方程

【例】求解：$2x^3 + 7x - 4 \equiv 0 \pmod{200}$

因为 $200 = 2^3 5^2$，所以化为求解：

$$2x^3 + 7x - 4 \equiv 0 \pmod{8}$$

$$2x^3 + 7x - 4 \equiv 0 \pmod{25}$$

模 8 的解是 $x \equiv 4 \pmod{8}$

模 25 的解是 $x \equiv 16 \pmod{25}$

使用中国剩余定理可以求出联立解 $x \equiv 116 \pmod{200}$。

多项式同余方程

通过对 $x = 0, 1, 2, 3, 4$ 直接验证，可见

$$2x^3 + 7x - 4 \equiv 0 \pmod{5}$$

的解是 $x \equiv 1 \pmod{5}$.

设 $x = 5t+1$,代入化简得到 $65t + 5 \equiv 15t + 5 \equiv 0 \pmod{25}$

因此 $3t + 1 \equiv 0 \pmod{5}$

$t \equiv 3 \pmod{5}$

$x \equiv 1 + 5t \equiv 16 \pmod{25}$

NOIP 2014 Day2 T3 解方程

给出一个多项式方程：

$$f(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n = 0$$

求这个方程在 [1, m] 内的整数解。

$n <= 100$, $|a_i| <= 10^{10000}$, $a_n \neq 0$, $m < 1000000$

NOIP 2014 Day2 T3 解方程

首先考虑，对于一个 x，我们如何判断它是不是方程的解。

由于 a_i 实在太大了，我们不可能求出准确的值。

因此我们可以在模意义下考虑。

任取一个 P （最好是质数），把所有的 a_i 对 P 取模。

NOIP 2014 Day2 T3 解方程

十进制数对 P 取模的代码：

```
int trans(char *str, int P){
	int ans = 0;
	for (int i = 0; str[i];i ++)
		ans = (10LL * ans + str[i] - '0') % P;
	return ans;
}
```

NOIP 2014 Day2 T3 解方程

这样一来，我们只要计算方程在模 P 意义下的值就行了。

如果算出来是 0，那说明有可能真实值是 0。

多取几个 P 验证。

如何快速计算？

霍纳法则

计算 $f(x) = 2x^4 - x^3 + 3x^2 + x - 5$ 最少需要几次乘法？

维护一个 x 的幂次，需要 $2n$ 次。

运用霍纳法则，可以做到 n 次。

$$
\begin{aligned}
f(x) &= x(2x^3 - x^2 + 3x + 1) - 5 \\
&= x(x(2x^2 - x + 3) + 1) - 5 \\
&= x(x(x(2x - 1) + 3) + 1) - 5
\end{aligned}
$$

这样可以使运行时间缩为原来的一半。

NOIP 2014 Day2 T3 解方程

但是我们要在 [1, m] 中找解，需要枚举 m 个数。

每次需要 O(n) 的时间验证。

这样是 O(nm) ，只能得到 70分。

NOIP 2014 Day2 T3 解方程

既然我们能想到在模 P 意义下验证是否是解。

我们也可以试试在模 P 意义下找解。先用较小的 P 试根。

对[0, P-1]的整数在模 P 意义下进行验证，时间复杂度

O(Pn)

如果 x 不是模 P 意义下的解，那么 P + x，2P + x 也就

不可能是模 P 意义下的解，也不可能是真正的解。

因此可以减少后面验证的次数。

NOIP 2014 Day2 T3 解方程

由拉格朗日定理可以知道方程在模 P 意义下有不超过 n 个解。

那么在 [1, m]中最多有 n * (m / P)个解。

对这些解模另一个质数进行验证，时间复杂度 $O(n^2m/P)$

中和一下 O(Pn) 和 $O(n^2m/P)$ ，取 $P=\sqrt{nm}$

时间复杂度就是 $O(n\sqrt{nm})$.

下节课再见