

数论基础

ZZQ323

2025 年 3 月 2 日

目录

1	小学的整数知识	2
2	带余除法	3
2.1	欧几里得算法	3
2.2	贝祖定理 (Bézout's Identity) or 裴蜀定理	4
3	GCD 相关的知识	6
3.1	奇奇怪怪的等式	6
3.2	拉梅定理	7
3.3	欧几里得算法、更相减损数、Stein 算法	7
3.4	LCM	7
3.5	代数基本定理	7
3.6	计算方法证明	7
3.7	LCM 与 GCD 的关系	8
4	丢番图	8
5	二阶丢番图方程的通解问题	9
5.1	图解证明	9

5.2	扩展欧几里得算法求特解	10
5.3	丢番图例题 * 可跳过	11
5.4	多元丢番图	11
6	同余	12
6.1	基本性质	13
6.2	一元线性同余方程	13
6.3	费马小定理	13
6.3.1	二项式展开证明	13
6.3.2	多项式展开证明	14
6.3.3	模算法证明	14
6.4	求逆	15
6.5	中国剩余定理 *	16
7	欧拉函数	18
7.1	同余类、剩余系	18
7.1.1	概念与性质	18
7.1.2	剩余系的复合	19
7.2	欧拉函数与欧拉定理	19
7.3	素数幂模同余方程、素数模同余方程与整数多项式	19

摘要

还在想的摘要 hh

你好, LaTeX!

1 小学的整数知识

- 整数可以表示成多项式 $n = c_k m^k + c_{k-1} m^{k-1} + c_{k-2} m^{k-2} + \cdots + c_1 m^1 + c_0 m^0$, 其中最高项不为 0 $c_k \neq 0$

- 如果 b 能整除 a , 那么 b 表示为 $b|a$, 这个时候 b 是 a 的因数; r 就是其中的余数。
- 素数 (只能被……的自然数)、合数、整除、公因子、最大公因子 \gcd (或者用括号表示)、最小公倍数 lcm
- 若 $(a_1, a_2, a_3, \dots, a_{n-1}, a_n) = 1$, 那么称 $a_1, a_2, a_3, \dots, a_{n-1}, a_n$ 互素; 只有 $i, j \in [1, n] \&\& i \neq j \&\& (a_{i,a_j}) = 1$, 这样才叫两两互素
- 整数之间的除法才有余数可言
- 因数分解最佳算法复杂度是 $\ln \left(\frac{\ln n}{\ln(\ln n)} \right)^{\frac{1}{2}} (n)$
- 为什么说最大的梅森素数是 $M_{44497} = 2^{44497} - 1$? 更大的就计算不出来了吗?
- 约定: $a \% n$ 得到结果的正负由被除数 a 决定, 与 n 无关
- 四则运算的结合律、交换律、分配律不影响取模

2 带余除法

2.1 欧几里得算法

首先根据带余除法这个式子, 我们可以列出:

$$a = bq_1 + r_1 \quad (1)$$

$$b = r_1q_2 + r_2 \quad (2)$$

$$r_1 = r_2q_3 + r_3 \quad (3)$$

$$\vdots$$

$$r_{n-3} = r_{n-2}q_{n-1} + r_{n-1} \quad (4)$$

$$r_{n-2} = r_{n-1}q_n + r_n \quad (5)$$

首先 $r_1 < b$ ，否则的话多的部分会使得 q 变大来吸纳；其次可以知道 $b > r_1 > r_2 > \cdots > r_{n-1} > r_n > 0$ ；这满足数列收敛的条件——因此 r_n 有极限且极限为 0；由此一来，只要 n 足够大，那么最后一项就是：

$$r_{n-2} = r_{n-1}q_n \quad (6)$$

ps：有些参考书会写到 $n+1$ ，其实都是一样的。

那么一步步反带入：

$$r_{n-2} = r_{n-1}q_n \quad (7)$$

$$r_{n-3} = r_{n-2}q_{n-1} + r_{n-1} = r_{n-1}q_nq_{n-1} + r_{n-1} \quad (8)$$

$$r_{n-4} = r_{n-3}q_{n-2} + r_{n-2} = (r_{n-1}q_nq_{n-1} + r_{n-1})q_{n-2} + r_{n-1}q_n \quad (9)$$

$$= r_{n-1}q_nq_{n-1}q_{n-2} + r_{n-1}q_{n-2} + r_{n-1}q_n$$

\vdots

$$b = g(q_n, q_{n-1}, q_{n-2}, \dots, q_4, q_3) \cdot r_{n-1} \quad (10)$$

$$r_1 = f(q_n, q_{n-1}, q_{n-2}, \dots, q_3, q_2) \cdot r_{n-1} \quad (11)$$

$$a = h(q_n, q_{n-1}, q_{n-2}, \dots, q_2, q_1) \cdot r_{n-1} \quad (12)$$

由于不同项数的组合顺序是不一样的，而且 q_i 之间也大概率不相同（偷懒）；所以不难看出： $f \neq g \neq h$ （三个互不相等）。进而证明了， a 、 b 之间的 gcd 就是 r_{n-1} 。

2.2 贝祖定理（Bézout's Identity）or 裴蜀定理

如果 $d = (a, b)$ ，则 $\exists q, p \in \mathbb{Z}$ ，st. $d = pa + qb$ 。在算法中，我们可以理解为“有一系列的 d ，但是只有最小公倍数是这个式子里面最小的——从而化简式子”证明如下：因为存在 r_{n-1} 对 a 、 b 的唯一表示；

$$b = g(q_n, q_{n-1}, q_{n-2}, \dots, q_4, q_3) \cdot r_{n-1} \quad (13)$$

$$a = h(q_n, q_{n-1}, q_{n-2}, \dots, q_2, q_1) \cdot r_{n-1} \quad (14)$$

所以，必然存在：

$$r_{n-1} = \frac{b}{g(q_n, q_{n-1}, q_{n-2}, \dots, q_4, q_3)} \quad (15)$$

$$r_{n-1} = \frac{a}{h(q_n, q_{n-1}, q_{n-2}, \dots, q_2, q_1)} \quad (16)$$

也就是

$$gcd(a, b) = d = \frac{b}{g(q_n, q_{n-1}, q_{n-2}, \dots, q_4, q_3)} \quad (17)$$

$$gcd(a, b) = d = \frac{a}{h(q_n, q_{n-1}, q_{n-2}, \dots, q_2, q_1)} \quad (18)$$

但是还是不够明确，回到之前的：

$$r_{n-2} = r_{n-1}q_n \quad (19)$$

$$r_{n-3} = r_{n-2}q_{n-1} + r_{n-1} = r_{n-1}q_nq_{n-1} + r_{n-1} \quad (20)$$

$$r_{n-4} = r_{n-3}q_{n-2} + r_{n-2} = (r_{n-1}q_nq_{n-1} + r_{n-1})q_{n-2} + r_{n-1}q_n \quad (21)$$

$$= r_{n-1}q_nq_{n-1}q_{n-2} + r_{n-1}q_{n-2} + r_{n-1}q_n$$

\vdots

$$b = g(q_n, q_{n-1}, q_{n-2}, \dots, q_4, q_3) \cdot r_{n-1} \quad (22)$$

$$r_1 = f(q_n, q_{n-1}, q_{n-2}, \dots, q_3, q_2) \cdot r_{n-1} \quad (23)$$

$$a = h(q_n, q_{n-1}, q_{n-2}, \dots, q_2, q_1) \cdot r_{n-1} \quad (24)$$

$$a = bq_1 + r_1 \quad (25)$$

然后帶入到最初的式子：

$$a = bq_1 + r_1 \quad (26)$$

$$a = bq_1 + f(q_n, q_{n-1}, q_{n-2}, \dots, q_3, q_2) \cdot r_{n-1} \quad (27)$$

$$r_{n-1} = \frac{1}{f(q_n, q_{n-1}, q_{n-2}, \dots, q_3, q_2)}a + \frac{q_1}{f(q_n, q_{n-1}, q_{n-2}, \dots, q_3, q_2)}b \quad (28)$$

然后……好像也没证明

正确的证明是：

$$a = bq_1 + r_1 \quad (29)$$

$$r_1 = a - bq_1 \quad (30)$$

$$b = r_1q_2 + r_2 \quad (31)$$

$$= (a - bq_1)q_2 + r_2$$

$$r_2 = b - (a - bq_1)q_2 \quad (32)$$

$$r_1 = r_2q_3 + r_3 \quad (33)$$

$$= (b - (a - bq_1)q_2)q_3 + r_3$$

$$\vdots$$

$$r_{n-1} = F(q)a + G(q)b \quad (34)$$

3 GCD 相关的知识

3.1 奇奇怪怪的等式

$$\begin{aligned} \gcd(a, b) &= \gcd(a, a + b) \\ &= \gcd(a, k \cdot a + b) \\ &= \gcd(a + k \cdot b, b) \end{aligned}$$

由贝祖定理知：如果 $d = (a, b)$ ，则 $\exists q, p \in \mathbb{Z}$ ，st. $d = pa + qb$ 。而上面两个式子无非就是令 $b = ka + b$ 或者 $a = kb + a$ ，展开来都是一致的，不需要证明什么东西。甚至，你令 $a = \frac{a+b}{2}$ 、 $b = \frac{a-b}{2}$ ，这样搞换底也是可以的。

3.2 拉梅定理

用欧几里得算法计算两个正整数的最大公约数，需要的除法次数不会超过两个整数中较小的那个十进制数的位数的 5 倍。

其实也就是： $O(n) \leq 5 \log_{10}(\min(a, b))$

3.3 欧几里得算法、更相减损数、Stein 算法

不知道怎么插入代码

3.4 LCM

3.5 代数基本定理

任意一个大于 1 的正整数都可以被分解为素数的乘积；

$$n = P_1^{\alpha_1} P_2^{\alpha_2} P_3^{\alpha_3} P_4^{\alpha_4} P_5^{\alpha_5}$$

3.6 计算方法证明

下面假设：

$$a = p_1^{c_1} p_2^{c_2} \cdots p_n^{c_n} \quad (35)$$

$$b = p_1^{f_1} p_2^{f_2} \cdots p_n^{f_n} \quad (36)$$

所以：

$$gcd(a, b) = p_1^{c_1} p_2^{c_2} \cdots p_n^{c_n} \cdot p_1^{f_1} p_2^{f_2} \cdots p_n^{f_n} \quad (37)$$

$$= p_1^{\min(c_1, f_1)} p_2^{\min(c_2, f_2)} \cdots p_n^{\min(c_n, f_n)} \quad (38)$$

$$lcm(a, b) = p_1^{\max(c_1, f_1)} p_2^{\max(c_2, f_2)} \cdots p_n^{\max(c_n, f_n)} \quad (39)$$

$$gcd(a, b) \cdot lcm(a, b) = a \cdot b \quad (40)$$

3.7 LCM 与 GCD 的关系

观察例题 1:

问题描述: 给定两个正整数 G 和 L , 问满足 $\gcd(x, y, z) = G$ 和 $\text{lcm}(x, y, z) = L$ 的 (x, y, z) 有多少个? 注意, $(1, 2, 3)$ 和 $(1, 3, 2)$ 是不同的。

图 1: 一道 hdu4497 的例题

这里有一个显然的性质:

$$L \% G \equiv 0$$

4 丢番图

在学习丢番图方程时, 常从线性或简单二次形式入门, 再逐步了解更复杂的高次或几何形式。

主要有以下类型:

- 线性丢番图: $ax + by = c$
- 多元线性丢番图: $a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_{n-1}x_{n-1} + a_nx_n = c$
- 高次丢番图: $x^n + y^n = z^n$
 - 勾股定理: $x^2 + y^2 = c^2$
 - 大费马定理: $x^n + y^n = z^n$ when $n > 2$ the equation is invalid.
 - Pell 方程 (一个双曲线): $x^2 - Dy^2 = 1$
- 指数丢番图: $a^x + b^y = c^z$

相关问题: 椭圆方程上的有理点构造问题。扩展欧几里得算法 (线性情况)、连分数法 (二次 Pell 方程)、Lattice-based 方法 (格上求解)。

5 二阶丢番图方程的通解问题

5.1 图解证明

对于 $ax + by = c$, 如果 $\gcd(a, b) \mid c$ (也就是 $c \% \gcd(a, b) = 0$)

注: 这里的 c 也可能是负的……因为……

对于一个特解 x_0, y_0 , 我可以很顺利地得到对应的整数通解:

$$\begin{aligned}x &= x_0 + \frac{b}{\gcd(a, b)}n \\y &= y_0 - \frac{a}{\gcd(a, b)}n\end{aligned}$$

原因就是: x 每增加一个 $\frac{b}{\gcd(a, b)}$ $x = x_0 + n$, 那么对应到等式中 y 就需要减少一个 $\frac{a}{\gcd(a, b)}$

那么 x 每增加一个 b $x = x_0 + bn$, 那么对应到等式中 y 就需要减少一个 $\frac{ab}{b} = a$

基于此, 给 x 和 y 的系数同时除以 \gcd , 那么就可以得到**最小步长的通解公式**, 保障不会漏掉什么通解。

但要是如果 $\gcd(a, b) \nmid c$ (也就是 $c \% \gcd(a, b) \neq 0$), 那么就不会有任何一个点在格子点上, 自然也不会有什么整数解……一个解都没有!

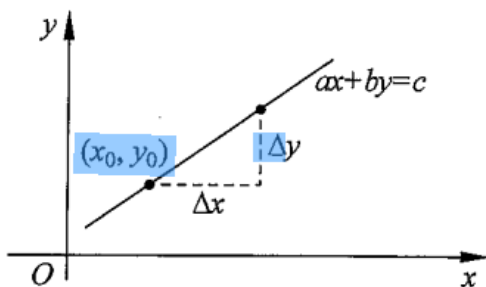


图 2: 二阶丢番图方程图解

5.2 扩展欧几里得算法求特解

首先已知: $pa + qb = \gcd(a, b)$; 这里只是把 $pa + qb = \gcd(a, b)$ 写成了 $xa + yb = \gcd(a, b)$; 由前面的步骤可知, x 、 y 都是 $f(q)$, 只要从上往下化简, 表示出 r_{n-1} , 其中的一大坨 q_n 就是 x 以及 y 了;

但是具体的:

$$a = bq_1 + r_1 \quad (41)$$

$$r_1 = a - bq_1 \quad (42)$$

$$b = r_1q_2 + r_2 \quad (43)$$

$$= (a - bq_1)q_2 + r_2$$

$$r_2 = b - (a - bq_1)q_2 \quad (44)$$

$$r_1 = r_2q_3 + r_3 \quad (45)$$

$$= (b - (a - bq_1)q_2)q_3 + r_3$$

$$\vdots$$

$$r_{n-1} = F(q)a + G(q)b \quad (46)$$

在这里我们改写成: $r_{n-1} = xa + yb$, 也即:

$$x = F(q)$$

$$y = G(q)$$

怎么计算这么长的 F 、 G 呢?

注意到, 在最小一个子问题的讨论中, 我们会得到 $a'x + b'y = \gcd(a, b)$ 。

那么, 在上一层的求解中, 我们就知道了下一层已经满足了这个条件; 但是, 我们保存了每一层的 a 、 b , 子递归中的 a 、 b 并非我们所有的 a 、 b , 所以我们需要调整 x 、 y 以适应这一层 a 、 b 的结论

$$\begin{cases} a' = b, \\ b' = a \pmod b, \end{cases}$$

$$\begin{aligned}
&\Rightarrow bx + (a \pmod b)y = \gcd(a, a \pmod b) = \gcd(a, b) \\
&\Rightarrow bx + (a \pmod b)y = bx + (a - \lfloor \frac{a}{b} \rfloor * b)y = \gcd(a, b) \\
&\Rightarrow ay + bx - \lfloor \frac{a}{b} \rfloor * by = \gcd(a, b) \\
&\Rightarrow ay + b(x - \lfloor \frac{a}{b} \rfloor y) = \gcd(a, b)
\end{aligned}$$

那么对于这一层的 x 、 y ，是不是又能通过下面那一层的 x 、 y 模拟了呢？

$$y = x_{\text{下}} - y_{\text{下}}$$

$$x = y_{\text{下}}$$

5.3 丢番图例题 * 可跳过

主要是算法问题

顺便复习 c++ 饿啊啊啊。

5.4 多元丢番图

实际上就是形如： $a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_{n-1}x_{n-1} + a_nx_n = c$ 式子的这么一个解。

当且仅当 $\gcd(a_1, a_2, a_3, \cdots, a_{n-1}, a_n) \mid c$ ，这个方程组有 tmd 无数个解。然后呢，像下面这样直接求解就行了。



例 6.15 线段上的格点数量

问题描述：在二维平面上，给定两个格点 $p_1 = (x_1, y_1)$ 和 $p_2 = (x_2, y_2)$ ，问线段 $p_1 p_2$ 上除了 p_1, p_2 外还有几个格点？设 $x_1 < x_2$ 。

(a) 题目 3a



例 6.17 青蛙的约会(洛谷 P1516)

问题描述：两只青蛙住在同一条纬度线上，它们各自向西跳，直到碰面为止。除非这两只青蛙在同一时间跳到同一点上，不然是永远都不可能碰面的。为了帮助这两只乐观的青蛙，你被要求写一个程序判断这两只青蛙是否能够碰面，会在什么时候碰面。把这两只青蛙分别叫作青蛙 A 和青蛙 B，并且规定纬度线上 0° 处为原点，由东向西为正方向，单位长度为 1 米，这样就得到了一条首尾相接的数轴。设青蛙 A 的出发点坐标是 x ，青蛙 B 的出发点坐标是 y 。青蛙 A 一次能跳 m 米，青蛙 B 一次能跳 n 米，两只青蛙跳一次所花费的时间相同。纬度线总长 L 米。求它们跳了几次以后才会碰面？

输入：输入 5 个整数 x, y, m, n, L 。

输出：输出碰面所需要的次数，如果永远不可能碰面，则输出一个字符串 “Impossible”。

(b) 题目 3b

图 3

$$\begin{cases} a_1 x_1 + a_2 x = d_2 t_2 \\ d_2 t_2 + a_2 x = d_3 t_3 \\ \vdots \\ d_{n-1} t_{n-1} + a_n x = d_n t_n \end{cases}$$

6 同余

长成： $a \equiv b \pmod{n}$ 就是同余式。

6.1 基本性质

1. 正整数 a, b 对 n 取模，它们的余数相同，记作： $a \equiv b \pmod{n}$
2. 若 $a - k * n = b$ ，则 $a \equiv b \pmod{n}$ ；换言之，我们可以将同余式 $a \equiv b \pmod{n}$ 与等式 $a \equiv b + k * n$ 互化
3. 若 $a \equiv b \pmod{n}$ 且 $c \equiv b \pmod{n}$ ，则 $a \equiv c \pmod{n}$
4. 若 $a \equiv b \pmod{n}$ ，则 $a + c \equiv b + c \pmod{n}$
5. 若 $a \equiv b \pmod{n}$ ，且 $c \equiv d \pmod{n}$ ，则 $a + c \equiv b + d \pmod{n}$ or $a * d \equiv b * c \pmod{n}$ （乘法的结论类似）

6.2 一元线性同余方程

若 $a - k * m = b$ ，则 $a \equiv b \pmod{m}$ ；换言之，我们可以将同余式 $a \equiv b \pmod{m}$ 与等式 $a \equiv b + k * m$ 互化。

那么，基于此 $ax \equiv b \pmod{m} \Rightarrow ax + my = b$;

设 $d = \gcd(a, m)$ ，如果有 $d \mid b$ （也即 $b \pmod{d} == 0$ ），那么有 d 个解答；反之无解。

至于为什么有 d 个，那是因为： $x = x_0 + \frac{m}{d}n$ ，由于解之间的间隔是 $\frac{m}{d}$ ，模 m 下的解是周期性的，而每个解对应于不同的同余类。

如果恰好 $d = 1$ ，那么就有唯一解！

6.3 费马小定理

6.3.1 二项式展开证明

考虑 $(1 + x)^p$ 的二项式展开：

$$(1 + x)^p = \sum_{i=0}^p \binom{p}{i} a_i = \binom{p}{0} a_0 + \binom{p}{1} a_1 + \binom{p}{2} a_2 + \cdots + \binom{p}{p-1} a_{p-1} + \binom{p}{p} a_p$$

根据: $C_n^m = \frac{n!}{(n-m)!m!}$, 且 p 是素数的情况下, 我们知道: 除了 C_p^0 或 C_p^p , 其他都会被 $\text{mod } p$ 给化掉:

$$(1+x)^p = \binom{p}{0}a_0 + \binom{p}{p}a_p = 1 + x^p \pmod{p}$$

$$(1+x)^p - x^p = 1 \pmod{p}$$

这就是经典的幂函数数列 $b_n = n^p$, 上述式子可化为: $b_{n+1} - b_n = 1$ 。由累加可得: $b_n = b_1 + n - 1 = 1^p + n - 1 = n$, 则 $b_a = a^p = a$
德政

6.3.2 多项式展开证明

考虑 a^p 的多项式展开:

$$(1_1 + 1_2 + 1_3 + \cdots + 1_{a-1} + 1_n)^p = \sum_{i=0}^p \binom{p}{k_1, k_2, \dots, k_n} 1^x = \sum_{i=0}^p \binom{p}{k_1, k_2, \dots, k_n}$$

在 $\text{mod } p$ 且 p 是质数的情况下, 除了 $\binom{p}{p, 0, 0, \dots, 0, 0}$ 、 $\binom{p}{0, p, 0, \dots, 0, 0}$ 等等都会变成 0 (因为没有数能消掉 p 除了它自己);

ps: $\binom{n}{n_1, n_2, n_3, \dots, n_{m-1}, n_m} = \frac{n!}{n_1! n_2! \dots n_{m-1}! n_m!}$

那么在全选一个的情况下, 就会有:

$$(1_1 + 1_2 + 1_3 + \cdots + 1_{a-1} + 1_n)^p = a^p = 1^p + 1^p + \cdots + 1^p + 1^p = a \pmod{p}$$

德政

6.3.3 模算法证明

我们首先考虑整数 $a, 2a, 3a, \dots, (p-1)a$ 。这些数都不等于其他数对 p 的模数, 也不等于 0。意思就是, 他们这一组数是独一无二的。

如果这样，那么有： $r \times a \equiv s \times a \pmod{p}$, $1 \leq r < s \leq p-1$ ；那么，两边消去 a 将得到 $r \equiv s \pmod{p}$ ，这是不可能的，因为 r 和 s 都在 1 和 $p-1$ 之间。

因此，前面的假设不成立， $a, 2a, 3a, \dots, (p-1)a$ 在 $\text{mod } p$ 的情况下总能对应 $1, 2, \dots, p-1$ 。

因为是一一对应，但又不好确定谁对谁的关系，所以我们把他们当成整体，全部乘起来： $a \times 2a \times 3a \times \dots \times (p-1) \times a \equiv 1 \times 2 \times 3 \times \dots \times (p-1) \pmod{p}$
这意味着： $a^{p-1} \times (p-1)! \equiv (p-1)! \pmod{p}$ 。

从这个表达式的两边消去 $(p-1)!$ ，我们得到： $a^{p-1} \equiv 1 \pmod{p}$ 。

6.4 求逆

求逆就是一元同余方程有唯一解的时候——如果没有唯一解答，那么就有“不止一个逆元”，这就很怪了。

- 扩展欧几里得，把已知的数字当作 a ，模数当作 b ——这个已经见过多次了
- 费马小定理得知 $a^p \equiv a \pmod{p}$ ，那么其实就是 $a \cdot a^{p-2} \equiv 1 \pmod{p}$
- 如果 a 的模逆是自身 a （在 $\text{mod } p$ 的情况下），那么 $a = 1$



例 6.21 乘法逆元(洛谷 P3811)

问题描述：给定 n, p ，求 $1 \sim n$ 所有整数在模 p 意义下的乘法逆元。 $1 \leq n \leq 3 \times 10^6$ ， $n < p < 20000528$ ， p 为质数。

输入：两个正整数 n 和 p 。

输出：输出 n 行，第 i 行表示 i 在模 p 下的乘法逆元。

图 4: 递推求素数降低复杂度的例题

上题，对于任意 $i > 1$ ：

$$\begin{aligned}\text{假设： } \frac{p}{i} &= k \dots r \\ \Rightarrow k \cdot i + r &= 0 \pmod{p} \\ \Rightarrow k + r \cdot i^{-1} &= 0 \pmod{p} \\ \Rightarrow i^{-1} &= -\frac{k}{r} = k \cdot r^{-1} = \left(p - \frac{p}{i}\right) \cdot r^{-1} \pmod{p}\end{aligned}$$

6.5 中国剩余定理 *

首先，我们有同余方程，而有方程就有方程组：

$$\begin{aligned}x &\equiv a_1 \pmod{m_1} \\ x &\equiv a_2 \pmod{m_2} \\ &\vdots \\ x &\equiv a_3 \pmod{m_3}\end{aligned}$$

那么我们一个一个满足：

$$\begin{aligned}x &= a_1 + m_1 n_1 \\ x &= a_2 + m_2 n_2 \\ &\vdots \\ x &= a_r + m_r n_r\end{aligned}$$

也就是会得到：

$$\begin{aligned}
n_1 &= \frac{a_2 + m_2 n_2 - a_1}{m_1} (\text{mod } m_2) = (a_2 + m_2 n_2 - a_1) * m_1^{-1} (\text{mod } m_2) \\
n_1 &= \frac{a_3 + m_3 n_3 - a_1}{m_1} (\text{mod } m_3) = (a_3 + m_3 n_3 - a_1) * m_1^{-1} (\text{mod } m_3) \\
&\vdots \\
n_1 &= \frac{a_r + m_r n_r - a_1}{m_1} (\text{mod } m_r) = (a_r + m_r n_r - a_1) * m_1^{-1} (\text{mod } m_r)
\end{aligned}$$

把 n_1 当成 x , 我们继续:

$$\begin{aligned}
n_2 &= \frac{a_3 + m_3 n_3 - a_2}{m_2} (\text{mod } m_3) = (a_3 + m_3 n_3 - a_2) * m_2^{-1} (\text{mod } m_3) \\
n_2 &= \frac{a_4 + m_4 n_4 - a_2}{m_2} (\text{mod } m_4) = (a_4 + m_4 n_4 - a_2) * m_2^{-1} (\text{mod } m_4) \\
&\vdots \\
n_2 &= \frac{a_r + m_r n_r - a_2}{m_2} (\text{mod } m_r) = (a_r + m_r n_r - a_2) * m_2^{-1} (\text{mod } m_r)
\end{aligned}$$

事情直到 n_{r-1} 结束: $n_{r-1} = \frac{a_r + m_r n_r - a_{r-1}}{m_{r-1}} (\text{mod } m_r) = (a_r + m_r n_r - a_{r-1}) * m_{r-1}^{-1} (\text{mod } m_r)$

然后把 n_{r-2} 算出来: $n_{r-2} = [a_{r-1} + (a_r + m_r n_r - a_{r-1}) * m_{r-1}^{-1} * m_{r-1} - a_{r-2}] * m_{r-2}^{-1}$

还是举个例子:

$$\begin{cases} x = 2 (\text{mod } 3) \\ x = 3 (\text{mod } 5) \\ x = 2 (\text{mod } 7) \end{cases}$$

由第一个式子: $\Rightarrow x = 2 + 3k$

; 然后我们让第一个式子满足第二个式子: $x = 2 + 3k = 3 (\text{mod } 5)$, 解得 $k = 2 (\text{mod } 5)$ 也即 $k = 5n + 2$, 那么最终的式子变成: $x = 2 + 3k = 2 + 3(5n + 2) = 2 + 15n + 6 = 15n + 8$

故技重施： $x = 15n + 8 = 2 \pmod{7}$ ，那么 $15n = 1 \pmod{7}$ ， $n = 1 \pmod{7}$ 。然后 $n = 7m + 1$ ，则 $x = 15n + 8 = 105m + 15 + 8 = 105m + 23$ 。
具体证明：*

7 欧拉函数

7.1 同余类、剩余系

7.1.1 概念与性质

同余类：对于所有的非零整数，把这么多数字不相交地分成 m 个集合，其中 $(\text{mod } m)$ 同余（结果是同一个数字 r ）的数字放一起，那么这 m 个集合，每一个都叫做同余类。

同余类全体构成的集合 $Z_m := \{r \pmod{m}, 0 \leq r < m\}$ ；比如说： $\text{mod } 5$ 的 Z_5 集合里面就包含 $[0]_5 = \{\dots, -15, -10, -5, 0, 5, 10, \dots\}$ 、 $[1]_5 = \{\dots, -14, -9, -4, 1, 6, 11, \dots\}$ 、 $[2]_5 = \{\dots, -13, -8, -3, 2, 7, 12, \dots\}$ 、 $[3]_5 = \{\dots, -12, -7, -2, 3, 8, 13, \dots\}$ 、 $[4]_5 = \{\dots, -11, -6, -1, 4, 9, 14, \dots\}$

既约同余类：对于 m 个同余类中的一个同余类 $r \pmod{m}$ ，如果余数的结果 r 和 m 互素，那么这个同余类就是既约同余类、既然约剩余类；

既约同余类全体构成一个集合 $Z_m^* = \{r \pmod{m}, 0 \leq r < m \wedge (r, m) = 1\}$ ；

举个例子：还是比如说 $\text{mod } 16$ 这个例子，那么 Z_{16} 就包含 $[0]_{16} \dots [15]_{16}$ ，但是这里要求 $(r, m) = 1$ ；所以 $[4]_{16}$ 、 $[8]_{16}$ 、 $[2]_{16}$ 这种就会被剔除掉；值得注意的是 $[0]_{16}$ 也得剔除掉；

既约同余类的个数记作 $\varphi(m)$ ，也就是欧拉函数，表示的是小于等于 m 并且和 m 互质的数的个数

（完全）剩余系：在模 m 体系中，对于任意一个整数 x ，我都能在 m 个整数数组 a_1, a_2, \dots, a_n 中找到对应的 r ，那么这 m 个数就叫做（完全）剩余系。

有各种各样的剩余系，但本质上都是对余数 r 加减一个 mod 去实现的。

既约剩余系:在剩余系里面挑出既约同余类的 r , 然后每个满足 $(x, m) = 1$ 的 x , 都能找到对应的 r ;

7.1.2 剩余系的复合

对于模数 $m = m_1 m_2$, 有: $Z_m = aZ_{m_1} + m_1 Z_{m_2}$, 其中 $(a, m_1) = 1$;

证明:

已知: $Z_{m_1} := \{r \bmod m_1, 0 \leq r < m_1\} = \{[0]_{m_1}, [1]_{m_1}, \dots, [m_1 - 1]_{m_1}\}$

$Z_{m_2} := \{r \bmod m_2, 0 \leq r < m_2\} = \{[0]_{m_2}, [1]_{m_2}, \dots, [m_2 - 1]_{m_2}\}$

$Z_m := \{r \bmod m, 0 \leq r < m\} = \{[0]_m, [1]_m, \dots, [m - 1]_m\}$

其中: $Z_m = Z_{m_1 m_2} := \{r \bmod m_1 m_2, 0 \leq r < m_1 m_2\}$

也就是说: 对同一个满足 (a, m_1) 的 a 和 m_1 , 无论 $z \in Z_m, x \in Z_{m_1}, y \in Z_{m_2}$ 怎么变, 总能找到以下等式: $z = ax + m_1 y$

根据代数基本定理, 我们知道一个整数可以被唯一分解为多个素数的乘积; 那其实意味着, 一个数可以被分解为多个不相干了向量 (好像表述不太准确);

7.2 欧拉函数与欧拉定理

既约同余类的个数记作 $\varphi(n) = \sum_{i=1}^n |_{\gcd(n,i)=1}$, 也就是欧拉函数, 表示的是小于等于 n 并且和 n 互质的数的个数。

当 p 是质数, 显然 $\varphi(p) = p - 1$; 特别地 $\varphi(1) = 1, \varphi(2) = 1$;

欧拉函数是积性函数, 因为: 所以。

7.3 素数幂模同余方程、素数模同余方程与整数多项式