

求解同余方程

小节概要₄

Linear Congruences(线性同余方程)

The Chinese Remainder Theorem(中国剩余定理)

Fermat's Little Theorem(费马小定理)

Pseudoprimes(伪素数)

Primitive Roots and Discrete Logarithms

(原根和离散对数)

Linear Congruences (线性同余式)

定义: 形如 $ax \equiv b \pmod{m}$ 的同余式，其中 m 为正整数， a 和 b 是整数， x 是变量，称为线性同余式。

线性同余式 $ax \equiv b \pmod{m}$ 的解是所有满足该同余式的整数 x 。

定义: 如果整数 \bar{a} 满足 $\bar{a}a \equiv 1 \pmod{m}$ 称 \bar{a} 为 a 的模 m 的逆元。

例子: 5 是 3 模 7 的逆元，因为 $5 \cdot 3 = 15 \equiv 1 \pmod{7}$

求解线性同余式的一种方法是使用 a 的逆元 \bar{a} （如果它存在）。虽然不能直接将同余式的两边除以 a ，但可以通过乘以 \bar{a} 来解 x 。

a modulo m 的逆元

以下定理保证了，当 a 和 m 互素时， a 在模 m 下的逆元存在。两个整数 a 和 b 当 $\gcd(a,b)=1$ 时互素。

定理 1:如果 a 和 m 是互素的整数且 $m>1$ ，那么 a 在模 m 下有逆元。此外，这个逆元在模 m 下是唯一的。（这意味着存在一个唯一的正整数 \bar{a} 小于 m ，它是 a 在模 m 下的逆元，且 a 的任何其他模 m 下的逆元都与 \bar{a} 在模 m 下同余）

证明:由于 $\gcd(a,m)=1$ ，根据贝祖定理，存在整数 s 和 t 使得：

- 因此, $sa + tm \equiv 1 \pmod{m}$.
- 由于 $tm \equiv 0 \pmod{m}$, 所以 $sa \equiv 1 \pmod{m}$
- 因此, s 是 a 在模 m 下的逆元.
- 逆元的唯一性也可以被证明.

求解逆元₁

欧几里得算法和贝祖系数为我们提供了一种系统的方法来寻找逆元.

例子:求 3 在模 7 下的逆元.

解:由于 $\gcd(3,7)=1$, 根据上页定理1, 3 在模 7 下的逆元存在.

- 使用欧几里得算法: $7 = 2 \cdot 3 + 1$.
- 从这个等式, 我们得到: $-2 \cdot 3 + 1 \cdot 7 = 1$, 可以看出, -2 和 1 是 3 和 7 的贝祖系数.
- 因此, -2 是 3 在模 7 下的逆元.
- 同时, 任何与 -2 在模 7 下同余的整数也是 3 的模 7 逆元, 例如 5、-9、12 等.

求解逆元₂

例子:求 101 在模 4620 下的逆元.

解:先使用欧几里得算法证明 $\text{gcd}(101, 4620) = 1$.

$$4620 = 45 \cdot 101 + 75$$

$$101 = 1 \cdot 75 + 26$$

$$75 = 2 \cdot 26 + 23$$

$$26 = 1 \cdot 23 + 3$$

$$23 = 7 \cdot 3 + 2$$

$$3 = 1 \cdot 2 + 1$$

$$2 = 2 \cdot 1$$

因为最后一个非零余数是 1,
 $\text{gcd}(101, 4620) = 1$

贝祖系数 : -35 and 1601

Working Backwards:

$$1 = 3 - 1 \cdot 2$$

$$1 = 3 - 1 \cdot (23 - 7 \cdot 3) = -1 \cdot 23 + 8 \cdot 3$$

$$1 = -1 \cdot 23 + 8 \cdot (26 - 1 \cdot 23) = 8 \cdot 26 - 9 \cdot 23$$

$$1 = 8 \cdot 26 - 9 \cdot (75 - 2 \cdot 26) = 26 \cdot 26 - 9 \cdot 75$$

$$1 = 26 \cdot (101 - 1 \cdot 75) - 9 \cdot 75$$

$$= 26 \cdot 101 - 35 \cdot 75$$

$$1 = 26 \cdot 101 - 35 \cdot (4620 - 45 \cdot 101)$$

$$= -35 \cdot 4620 + 1601 \cdot 101$$

101 在模 4620 下的逆元为 1601

用逆元求解线性同余式

我们可以通过两边同时乘以 \bar{a} 来解方程 $ax \equiv b \pmod{m}$.

例子:解方程 $3x \equiv 4 \pmod{7}$

解:之前已经找到 -2 是 3 在模 7 下的逆元。我们将方程两边同时乘以 -2

$$-2 \cdot 3x \equiv -2 \cdot 4 \pmod{7}.$$

因为 $-6 \equiv 1 \pmod{7}$ 并且 $-8 \equiv 6 \pmod{7}$, 因此如果 x 是解, 则有 $x \equiv -8 \equiv 6 \pmod{7}$

我们需要确定每一个满足 $x \equiv 6 \pmod{7}$ 的 x 是否都是方程的解。假设 $x \equiv 6 \pmod{7}$, 可验证 $3x \equiv 3 \cdot 6 = 18 \equiv 4 \pmod{7}$, 这表明所有这样的 x 都满足这个同余式。

因此, 解集是所有满足 $x \equiv 6 \pmod{7}$ 的整数, 即 $x=6, 13, 20$ 或者 $x=-1, -8, -15$

The Chinese Remainder Theorem (中国剩余定理)¹

在公元一世纪，中国数学家孙子提出了以下问题：“有物不知其数，三分之余二，五分之余三，七分之余二，此物几何？”

这个谜题可以被转化为解以下同余方程组的问题：

$$x \equiv 2 \pmod{3},$$

$$x \equiv 3 \pmod{5},$$

$$x \equiv 2 \pmod{7}$$

我们将看到一个被称为中国剩余定理的定理是如何用于解决孙子的这个问题的。

中国剩余定理²

定理 2: (*The Chinese Remainder Theorem*) 设 m_1, m_2, \dots, m_n 是两两互素的
大于 1 的正整数并且 a_1, a_2, \dots, a_n 是任意整数. 对于以下同余方程组

$$x \equiv a_1 \pmod{m_1}$$

$$x \equiv a_2 \pmod{m_2}$$

.

.

.

$$x \equiv a_n \pmod{m_n}$$

此方程组有唯一的模 $m = m_1 m_2 \cdots m_n$ 的解

(即, 存在一个解 x , 使得 $0 \leq x < m$, 且所有其他解均与该解模 m 同余.)

证明: 我们将通过描述一种构造解的方法来证明解的存在性.

中国剩余定理³

为了构造解，首先令 $M_k = m/m_k$ 对于 $k = 1, 2, \dots, n$ 以及 $m = m_1 m_2 \cdots m_n$. 因为 $\gcd(m_k, M_k) = 1$, 根据前述定理1, 存在整数 y_k , 它是 M_k 在模 m_k 下的逆元, 使得

$$M_k y_k \equiv 1 \pmod{m_k}.$$

然后构造和

$$x = a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots + a_n M_n y_n.$$

注意到因为 $M_j \equiv 0 \pmod{m_k}$ 当 $j \neq k$, 所以在这个和中, 除了第 k 项外, 所有项模 m_k 都同余于 0.

由于 $M_k y_k \equiv 1 \pmod{m_k}$, 我们有 $x \equiv a_k M_k y_k \equiv a_k \pmod{m_k}$, 对于 $k = 1, 2, \dots, n$. 因此, x 是以下 n 个同余方程的同时解.

$$x \equiv a_1 \pmod{m_1}$$

$$x \equiv a_2 \pmod{m_2}$$

.

.

$$x \equiv a_n \pmod{m_n}$$

中国剩余定理的唯一性证明

设有两两互质的正整数 n_1, n_2, \dots, n_k , 以及任意整数 a_1, a_2, \dots, a_k , 那么同余方程组

$$\begin{cases} x \equiv a_1 \pmod{n_1} \\ x \equiv a_2 \pmod{n_2} \\ \vdots \\ x \equiv a_k \pmod{n_k} \end{cases}$$

有唯一解模 $N = n_1 n_2 \dots n_k$ 。

唯一性: 要证明解的唯一性, 假设存在两个解 x 和 y , 它们都满足上述的同余方程组。那么我们有

$$x \equiv a_i \pmod{n_i} \quad \text{和} \quad y \equiv a_i \pmod{n_i} \quad \text{对于所有 } i = 1, 2, \dots, k.$$

因此, 对于所有的 i , 我们有

$$x \equiv y \pmod{n_i}.$$

这意味着 n_i 整除 $x - y$ 对于所有的 i 。

由于 n_1, n_2, \dots, n_k 两两互质, 且 $n_1 | (x-y), n_2 | (x-y), \dots, n_k | (x-y)$, 可证明 $N = (n_1 * n_2 * \dots * n_k) | (x-y)$;

既然 $N | (x-y)$, 这意味着 $x \equiv y \pmod{N}$ 。因此, 任何两个解 x 和 y 在模 N 下都是相同的, 这证明了解的唯一性。

中国剩余定理⁴

例:我们将应用中国剩余定理来解决孙子的问题:

$$x \equiv 2 \pmod{3}, \quad x \equiv 3 \pmod{5}, \quad x \equiv 2 \pmod{7}.$$

- 令 $m = 3 \cdot 5 \cdot 7 = 105$, $M_1 = m/3 = 35$, $M_2 = m/5 = 21$, $M_3 = m/7 = 15$.
- 我们注意到
 - 2 是 $M_1 = 35$ 模 3 的逆元 因为 $35 \cdot 2 \equiv 2 \cdot 2 \equiv 1 \pmod{3}$
 - 1 是 $M_2 = 21$ 模 5 的逆元 因为 $21 \equiv 1 \pmod{5}$
 - 1 是 $M_3 = 15$ 模 7 的逆元 因为 $15 \equiv 1 \pmod{7}$
- 因此,
$$\begin{aligned} x &= a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 \\ &= 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1 = 233 \equiv 23 \pmod{105} \end{aligned}$$
- 所以, 23 是满足所有方程的最小正整数解

反向替换方法

我们还可以通过将线性同余方程组中的每个同余方程转化为等式，代入变量的值到另一个同余方程中，并持续进行这个过程，直到解决所有同余方程，从而解决具有互素模数的线性同余方程组。这种方法称为回代法。

例: 使用回代法找到所有满足以下条件的整数 x 使得 $x \equiv 1 \pmod{5}$, $x \equiv 2 \pmod{6}$, and $x \equiv 3 \pmod{7}$.

解: 第一个同余方程可以重写为 $x = 5t + 1$, 其中 t 是一个整数.

- 代入第二个同余方程得到 $5t + 1 \equiv 2 \pmod{6}$.
- 解这个方程得 $t \equiv 5 \pmod{6}$.
- 使用定理4得到 $t = 6u + 5$, 其中 u 是一个整数.
- 代入 $x = 5t + 1$, 解得 $x = 5(6u + 5) + 1 = 30u + 26$.
- 再代入第三个方程得到 $30u + 26 \equiv 3 \pmod{7}$.
- 解得 $u \equiv 6 \pmod{7}$.
- 使用定理4得到 $u = 7v + 6$, 其中 v 是一个整数.
- 将 $u = 7v + 6$ 代入 $x = 30u + 26$ 中得到, $x = 30(7v + 6) + 26 = 210v + 206$.

将这个结果转化为同余方程, 我们得到解 $x \equiv 206 \pmod{210}$.

Wilson Theorem (威尔逊定理)

Wilson定理: P 是素数当且仅当 $(p-1)! \equiv -1 \pmod{p}$.

引理: 如 p 为素数且 $x^2 \equiv 1 \pmod{p}$, 则 $x \equiv 1 \pmod{p}$ 或 $x \equiv -1 \pmod{p}$.

引理证明: $p \mid (x^2 - 1)$, 即 $p \mid (x+1)(x-1)$, 因 p 为素数, 则 $p \mid (x+1)$ 或 $p \mid (x-1)$, 得证。

Wilson定理证明:

① 先证当 $(p-1)! \equiv -1 \pmod{p}$ 时, p 为素数:

设 $n \mid p$ 且 $n \neq p$, 则 $n \in \{1, 2, \dots, p-1\}$, 能推导出 $n \mid (p-1)!$; 根据条件,

$p \mid ((p-1)! + 1)$, 因为 $n \mid p$, 则 推出 $n \mid ((p-1)! + 1)$; 又已推出 $n \mid (p-1)!$, 得 $n \mid 1$, 则 $n=1$, 推出 p 的因子只有 p 和 1 , 则 p 为素数.

② 再证当 p 为素数时, $(p-1)! \equiv -1 \pmod{p}$ 成立:

$(p-1)! = 1 * (p-1) * (2 * 3 * \dots * (p-2)) = (p-1) * ((p-3)/2 \text{对互逆的数相乘}) \pmod{p} = (p-1) \pmod{p} = -1 \pmod{p}$, 得证.

引例: $6! \pmod{7} = 1 * (2 * 4) * (3 * 5) * 6 \pmod{7} = 6 \pmod{7} = -1 \pmod{7}$;

欧拉函数与欧拉定理

欧拉函数 $\phi(n)$: 比n小且与n互素的正整数的个数.

例: $\phi(6)=2$ [1,5]; $\phi(7)=6$ [1,2,3,4,5,6]

欧拉函数 $\phi(n)$ 的计算: 如果 p 是素数, 则 $\phi(p)=p-1$. 如果 $n=pq$, 且 p 和 q 是两个不同的素数, 则 $\phi(n)=(p-1)(q-1)$.

欧拉定理 : 如果 $\gcd(a,n)=1$, 则 $a^{\phi(n)} \equiv 1 \pmod{n}$.

证明 : 设 $X=\{x \mid x\text{为小于}n\text{的正整数且}\gcd(x,n)=1\}$, 则 $|X|=\phi(n)$;

对元素 a 且 $\gcd(a,n)=1$, 有集合 $aX=\{ax \pmod{n} \mid x \in X\}$;

因为 $\gcd(a, n)=\gcd(x, n)=1$, 可知 $\gcd(ax, n)=1$; 再看 aX 集合中任意两个元素 $ax_i \equiv ax_j \pmod{n}$ 是否成立; 如成立, 则推出 $n \mid (x_i - x_j)$, 即 $x_i = x_j$, 矛盾 \Rightarrow 不成立
因此 $aX=X$, 即两个集合相等;

将集合 X 和集合 aX 中所有元素相乘, 得: $\prod_{x \in X} x = \prod_{y \in aX} y = \prod_{x \in X} ax \pmod{n}$

再将以上等式两边都乘 $x \pmod{n}$ 的逆, 等式最左边为1, 等式最右边为 $a^{\phi(n)}$,
因此 $a^{\phi(n)} \equiv 1 \pmod{n}$, 得证.

欧拉函数与欧拉定理

欧拉定理：如果 $\gcd(a,n)=1$, 则 $a^{\phi(n)} \equiv 1 \pmod{n}$.

欧拉定理应用：求解 a 在模 n 下的逆元，即 $a^{\phi(n)-1} a \equiv 1 \pmod{n}$

例：求 3 在模 7 下的逆元

解：

$$\gcd(3, 7) = 1$$

$$3^{\phi(7)-1} 3 \equiv 1 \pmod{7}$$

$$\text{逆元是 } 3^{\phi(7)-1} = 3^{(7-1)-1} = 243$$

存在唯一小于 7 的正整数 $243 \pmod{7} = 5$, 5 也是 3 在模 7 下的逆元 5

练习：求 3 在模 7 下的逆元

解：逆元是 3

欧拉定理推论

欧拉定理推论：如果 $\gcd(a,n)=1$, 则 $a^x \equiv a^{x \pmod{\phi(n)}} \pmod{n}$.

证明： $x = q * \phi(n) + r$, $r = x \pmod{\phi(n)}$,

则 $a^x = a^{q\phi(n)+r} = (a^{\phi(n)})^q a^r \equiv a^r \equiv a^{x \pmod{\phi(n)}} \pmod{n}$.

欧拉定理推论应用：计算 $2^{999} \pmod{21}=?$

因 $n=21=3*7$, 则 $\phi(21)=2*6=12$, 又因为 $\gcd(2,21)=1$,

则 $2^{999} \pmod{21}=2^{999 \pmod{12}}=2^3=8 \pmod{21}$

在RSA公钥密码算法正确性证明中也会用到上述推论.

Fermat's Little Theorem (费马小定理)

定理 3:如果 p 是一个素数且 a 是一个不被 p 整除的整数，则有

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

此外，对于每个整数 a ，都有

$$a^p \equiv a \pmod{p}$$



Pierre de Fermat
(1601-1665)

费马小定理应用: 计算整数的大次幂的模 p 的余数

例: 计算 $7^{222} \pmod{11}$.

根据费马小定理，我们知道 $7^{10} \equiv 1 \pmod{11}$ ，所以 $(7^{10})^k \equiv 1 \pmod{11}$ ，对于每个正整数 k . 因此，

$$7^{222} = 7^{22 \cdot 10 + 2} = (7^{10})^{22} 7^2 \equiv (1)^{22} \cdot 49 \equiv 5 \pmod{11}.$$

因此， $7^{222} \pmod{11} = 5$.

Pseudoprimes(伪素数)¹

根据费马小定理，对于素数 $n > 2$ ，有

$$2^{n-1} \equiv 1 \pmod{n}.$$

但如果这个同余方程成立， n 不一定是素数. 合成整数 n 满足 $2^{n-1} \equiv 1 \pmod{n}$ 被称为该基数2下的伪素数.

例: 整数 341 是以2为基数的伪素数.

$$341 = 11 \cdot 31$$

$$2^{340} \equiv 1 \pmod{341}$$

我们可以将 2 替换为任意整数 $b \geq 2$.

定义: 设 b 是一个正整数。如果 n 是一个合数，并且 $b^{n-1} \equiv 1 \pmod{n}$ ，那么 n 被称为以 b 为基数的伪素数

伪素数₂

给定一个正整数 n , 如果 $2^{n-1} \equiv 1 \pmod{n}$:

- 如果 n 满足该同余方程, 它可能是素数, 也可能是以2为基数的伪素数
- 如果 n 不满足该同余方程, 它是合数

使用额外的基数 b 进行类似的测试, 可以提供更多关于 n 是否为素数的证据

与素数相比, 在不超过正实数 x 的正整数中, 基数 b 的伪素数相对较少

- 例如, 在小于 10^{10} 的正整数中, 有 455,052,512 个素数, 但只有 14,884 个基数 2 的伪素数.

卡米切尔数

定义: 一个合数 n , 如果对所有满足 $\gcd(b,n)=1$ 的正整数 b 都满足 $b^{n-1} \equiv 1 \pmod{n}$, 则称 n 为卡米切尔数

例子: 整数 561 是一个卡米切尔数。证明如下:

- 561 是合数, 因为 $561 = 3 \cdot 11 \cdot 13$.
- 如果 $\gcd(b, 561) = 1$, 那么 $\gcd(b, 3) = 1$, 那么 $\gcd(b, 11) = \gcd(b, 17) = 1$.
- 根据费马小定理: $b^2 \equiv 1 \pmod{3}$, $b^{10} \equiv 1 \pmod{11}$, $b^{16} \equiv 1 \pmod{17}$.
- 则

$$b^{560} = (b^2)^{280} \equiv 1 \pmod{3},$$

$$b^{560} = (b^{10})^{56} \equiv 1 \pmod{11},$$

$$b^{560} = (b^{16})^{35} \equiv 1 \pmod{17}.$$

- 由此可知, 对于所有满足 $\gcd(b, 561) = 1$ 的正整数 b , 都有 $b^{560} \equiv 1 \pmod{561}$ 。因此, 561 是一个卡米切尔数。

Primitive Roots (原根)

定义: 模素数 p 的原根是 \mathbf{Z}_p 中的一个整数 r , 使得 \mathbf{Z}_p 中的每一个非零元素都是 r 的某个幂次 (即 $r^e \bmod p$)

例 1: 对于素数 11, 2 是一个原根, 因为 \mathbf{Z}_{11} 中的每个元素都可以表示为 2 的某个幂。计算如下:

$$2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 5, 2^5 = 10,$$

$$2^6 = 9, 2^7 = 7, 2^8 = 3, 2^9 = 6, 2^{10} = 1.$$

例 2: 对于质数 11, 3 不是一个原根, 因为并非所有 \mathbf{Z}_{11} 中的元素都是 3 的某个幂。计算如下:

$3^1 = 3, 3^2 = 9, 3^3 = 5, 3^4 = 4, 3^5 = 1$, 当计算 3 的更高幂次时, 这个模式会重复, 并不能生成所有 \mathbf{Z}_{11} 中的元素.

重要事实: 对于每个素数 p , 总存在一个原根.

Discrete Logarithms (离散对数)

假设 p 是一个素数， r 是模 p 的一个原根。如果 a 是位于 1 和 $p-1$ 之间的整数，即 a 是 \mathbf{Z}_p 的一个元素，那么存在一个唯一的指数 e 使得 $r^e \equiv a \pmod{p}$ 成立，也就是 $r^e \bmod p = a$.

定义：设 p 是一个素数， r 是模 p 的一个原根，且 a 是位于 1 到 $p-1$ 之间的整数。如果 $r^e \bmod p = a$ ，且 $1 \leq e \leq p-1$ ，我们称 e 是以 r 为底数 a 模 p 的离散对数，写作 $\log_r a = e$ (其中素数 p 是默认的)。

例：试找出以 2 为底 3 和 5 模 11 的离散对数

因为以 2 为底 3 模 11 的离散对数是 8，即 $2^8 \bmod 11 = 3$. 我们写作 $\log_2 3 = 8$

因为以 2 为底 5 模 11 的离散对数是 4，即 $2^4 \bmod 11 = 5$. 我们写作 $\log_2 5 = 4$

同余式的应用

小节概要

Hashing Functions (哈希函数)

Pseudorandom Numbers (伪随机数)

Check Digits (数字校验位)

Hashing Functions (哈希函数)

定义: 哈希函数 h 将具有键 k 的记录分配到内存位置 $h(k)$.

- 一个常见的哈希函数是 $h(k)=k \bmod m$, 其中 m 是内存位置的总数.
- 由于这个哈希函数是满射的, 所有内存位置都是可能的.

例子: 设哈希函数为 $h(k)=k \bmod 111$ 。这个哈希函数将具有社会安全号码作为键的客户记录分配到内存位置, 如下所示:

$$h(064212848) = 064212848 \bmod 111 = 14$$

$$h(037149212) = 037149212 \bmod 111 = 65$$

$h(107405723) = 107405723 \bmod 111 = 14$, 但由于位置 14 已经被占用, 因此记录被分配到下一个可用的位置, 即 15.

哈希函数不是一一对应的, 因为可能的键比内存位置要多得多。当多个记录被分配到同一位置时, 我们说发生了冲突。这里, 通过将记录分配到第一个空闲位置解决了冲突.

对于冲突解决, 我们可以使用**线性探测函数**: $h(k,i) = (h(k) + i) \bmod m$, 其中 i 从 0 到 $m-1$ 变化.

还有许多其他处理冲突的方法.

伪随机数₁

随机选择的数字在许多应用中都是必需的，包括计算机模拟。
随机数并不是真正的随机，因为它们是通过系统化的方法生成的。
线性同余法是生成伪随机数的一种常用方法。

生成伪随机数需要四个整数：模数 m 、乘数 a 、增量 c 、以及种子 x_0 ，其中 $2 \leq a < m, 0 \leq c < m, 0 \leq x_0 < m$.

我们通过递归定义的函数生成伪随机数序列 $\{x_n\}$ ，其中 $0 \leq x_n < m$ 对于所有 n 成立。

递归定义的函数为： $x_{n+1} = (ax_n + c) \bmod m$.

如果需要介于 0 和 1 之间的伪随机数，可以将生成的数字除以模数，即 x_n/m .

伪随机数₂

例: 找到由线性同余法生成的伪随机数序列, 模数为 $m=9$, 乘数为 $a=7$, 增量为 $c=4$, 种子为 $x_0 = 3$.

解:通过依次使用同余关系计算序列的项

$$x_1 = 7x_0 + 4 \bmod 9 = 7 \cdot 3 + 4 \bmod 9 = 25 \bmod 9 = 7,$$

$$x_2 = 7x_1 + 4 \bmod 9 = 7 \cdot 7 + 4 \bmod 9 = 53 \bmod 9 = 8,$$

$$x_3 = 7x_2 + 4 \bmod 9 = 7 \cdot 8 + 4 \bmod 9 = 60 \bmod 9 = 6,$$

$$x_4 = 7x_3 + 4 \bmod 9 = 7 \cdot 6 + 4 \bmod 9 = 46 \bmod 9 = 1,$$

$$x_5 = 7x_4 + 4 \bmod 9 = 7 \cdot 1 + 4 \bmod 9 = 11 \bmod 9 = 2,$$

$$x_6 = 7x_5 + 4 \bmod 9 = 7 \cdot 2 + 4 \bmod 9 = 18 \bmod 9 = 0,$$

$$x_7 = 7x_6 + 4 \bmod 9 = 7 \cdot 0 + 4 \bmod 9 = 4 \bmod 9 = 4,$$

$$x_8 = 7x_7 + 4 \bmod 9 = 7 \cdot 4 + 4 \bmod 9 = 32 \bmod 9 = 5,$$

$$x_9 = 7x_8 + 4 \bmod 9 = 7 \cdot 5 + 4 \bmod 9 = 39 \bmod 9 = 3.$$

生成的序列是: 3,7,8,6,1,2,0,4,5,3,7,8,6,1,2,0,4,5,3,... 它在生成 9 项后重复.

通常, 计算机使用增量 $c=0$ 的线性同余生成器。这种生成器称为纯乘法生成器
使用模数 $2^{31} - 1$ 和乘数 $7^5 = 16,807$ 的生成器在重复之前会生成 $2^{31} - 2$ 个数字

校验位: UPCs

常见的检测数字串中错误的方法是添加一个额外的数字在末尾，该数字通过一个函数进行评估。如果最后的数字不正确，则假定该字符串不正确。

例子:零售产品通过其通用产品代码 (UPC) 进行标识。这些代码通常有12位十进制数字，其中最后一位是校验位。校验位通过以下同余式确定：

$$3x_1 + x_2 + 3x_3 + x_4 + 3x_5 + x_6 + 3x_7 + x_8 + 3x_9 + x_{10} + 3x_{11} + x_{12} \equiv 0 \pmod{10}.$$

- 假设UPC的前11位数字是 79357343104。那么校验位是多少？
- 041331021641 是否是一个有效的UPC?

解:

a. $3 \cdot 7 + 9 + 3 \cdot 3 + 5 + 3 \cdot 7 + 3 + 3 \cdot 4 + 3 + 3 \cdot 1 + 0 + 3 \cdot 4 + x_{12} \equiv 0 \pmod{10}$

$$21 + 9 + 9 + 5 + 21 + 3 + 12 + 3 + 3 + 0 + 12 + x_{12} \equiv 0 \pmod{10}$$

$$98 + x_{12} \equiv 0 \pmod{10}$$

$x_{12} \equiv 2 \pmod{10}$ 所以校验位是2.

b. $3 \cdot 0 + 4 + 3 \cdot 1 + 3 + 3 \cdot 3 + 1 + 3 \cdot 0 + 2 + 3 \cdot 1 + 6 + 3 \cdot 4 + 1 \equiv 0 \pmod{10}$

$$0 + 4 + 3 + 3 + 9 + 1 + 0 + 2 + 3 + 6 + 12 + 1 = 44 \not\equiv 0 \pmod{10}$$

所以, 041331021641 不是有效UPC.

校验位:ISBNs

图书通过国际标准图书编号 (ISBN-10) 来识别，这是一个10位数字的代码。前9位数字用来标识语言、出版社和图书。第十位数字是校验位，通过以下的同余关系确定

$$x_{10} \equiv \sum_{i=1}^9 ix_i \pmod{11}.$$

ISBN-10号码的有效性可以通过以下等式来评估

$$\sum_{i=0}^{10} ix_i \equiv 0 \pmod{11}$$

- 假设 ISBN-10 的前 9 位数字是 007288008。校验数字是多少?
- 084930149X 是有效的 ISBN-10 吗? X is used for the digit 10.

解:

- $$X_{10} \equiv 1 \cdot 0 + 2 \cdot 0 + 3 \cdot 7 + 4 \cdot 2 + 5 \cdot 8 + 6 \cdot 8 + 7 \cdot 0 + 8 \cdot 0 + 9 \cdot 8 \pmod{11}.$$
$$X_{10} \equiv 0 + 0 + 21 + 8 + 40 + 48 + 0 + 0 + 72 \pmod{11}.$$
$$X_{10} \equiv 189 \equiv 2 \pmod{11}. \text{ 因此, } X_{10} = 2.$$
- $$1 \cdot 0 + 2 \cdot 8 + 3 \cdot 4 + 4 \cdot 9 + 5 \cdot 3 + 6 \cdot 0 + 7 \cdot 1 + 8 \cdot 4 + 9 \cdot 9 + 10 \cdot 10 =$$
$$0 + 16 + 12 + 36 + 15 + 0 + 7 + 32 + 81 + 100 = 299 \equiv 2 \not\equiv 0 \pmod{11}$$

因此, 084930149X 不是有效ISBN-10.

单一错误是指识别号码中的一个数字错误，而置换错误是指两个数字意外交换的位置。这两种错误都可以通过 ISBN-10 的检查数字进行检测.