费马小定理

观察

取整数 a，考虑它的幂 a，a^2，a^3，… 模 m. 在这些幂中存在什么模式吗？我们先看素数模 $m=p$ 的情形，因为这时容易识别出模式. 这种现象在数论中(尤其在同余理论中)很普遍. 当寻求同余模式时最好先从素数模入手.

对每个素数 $p=3$，$p=5$，$p=7$，列出整数 $a=0$，1，2，…和一些幂模 p. 在进一步阅读之前，应该停下来去考察这些表，并设法列出猜测的模式公式. 然后通过制作 $p=11$ 的类似表格验证你的猜测，看你的模式是否仍是正确的.

a	a^2	a^3	a^4
0	0	0	0
1	1	1	1
2	1	2	1

$a^k \pmod 3$

a	a^2	a^3	a^4	a^5	a^6
0	0	0	0	0	0
1	1	1	1	1	1
2	4	3	1	2	4
3	4	2	1	3	4
4	1	4	1	4	1

$a^k \pmod 5$

a	a^2	a^3	a^4	a^5	a^6	a^7	a^8
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	4	1	2	4	1	2	4
3	2	6	4	5	1	3	2
4	2	1	4	2	1	4	2
5	4	6	2	3	1	5	4
6	1	6	1	6	1	6	1

$a^k \pmod 7$

许多令人感兴趣的模式可从这些表看出. 在本章，相关模式在列

$$a^2 \pmod 3 \quad a^4 \pmod 5 \quad 与 \quad a^6 \pmod 7$$

中可观察出. 除去顶端一个，这些列中的每一项等于 1. 这一模式对较大的素数继续成立吗？可检查 $p=11$ 的表格，会发现

$$1^{10} \equiv 1 \pmod{11}, 2^{10} \equiv 1 \pmod{11}, 3^{10} \equiv 1 \pmod{11} \cdots$$

$$9^{10} \equiv 1 \pmod{11} \quad 与 \quad 10^{10} \equiv 1 \pmod{11}.$$

猜想

由此得到下述猜想：

$$a^{p-1} \equiv 1 \pmod p, \quad 1 \leqslant a < p.$$

当然，不需将 a 限制在 1 与 $p-1$ 之间. 如果 a_1 与 a_2 是 p 的不同倍数，则其幂对模 p 相同. 所以关于 a 的真正条件为它不是 p 的倍数. 这个结果由费马在1640 年给 Bessy 的信中首先提出，但费马没有给出证明的细节. 第一个证明应归于莱布尼茨[⊖].

定理 9.1(费马小定理) 设 p 是素数，a 是任意整数且 $a \not\equiv 0 \pmod p$，则

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

作用：简化计算

在给出费马小定理证明之前，我们要指出它的幂并说明如何用其进行简化计算. 作为特例，考虑同余式

$$6^{22} \equiv 1 \pmod{23}.$$

这说明数 $6^{22}-1$ 是 23 的倍数. 如果不用费马小定理验证这个事实，则必须算出 6^{22} 减 1 再除以 23. 下面是所得结果：

$$6^{22} - 1 = 23 \cdot 5\,722\,682\,775\,750\,745.$$

类似地，为直接验证 $73^{100} \equiv 1 \pmod{101}$，必须计算 $73^{100}-1$. 不幸的是，$73^{100}-1$ 有 187 位数. 注意这个例子仅使用 $p=101$，这是比较小的素数. 因此，费马小定理描述了有关大数的一个令人惊讶的事实.

我们可使用费马小定理简化计算. 例如，为计算 $2^{35} \pmod 7$，可利用 $2^6 \equiv 1 \pmod 7$. 所以记 $35 = 6 \cdot 5 + 5$，使用指数律计算

$$2^{35} = 2^{6\cdot 5+5} = (2^6)^5 \cdot 2^5 \equiv 1^5 \cdot 2^5 \equiv 32 \equiv 4 \pmod 7.$$

类似地，假设要解同余式 $x^{103} \equiv 4 \pmod{11}$. 肯定有 $x \not\equiv 0 \pmod{11}$，因此由费马小定理得

$$x^{10} \equiv 1 \pmod{11}.$$

两边自乘 10 次得 $x^{100} \equiv 1 \pmod{11}$，然后乘以 x^3 得 $x^{103} \equiv x^3 \pmod{11}$. 要解原同余式，正好需要解 $x^3 \equiv 4 \pmod{11}$. 通过连续尝试 $x=1$，$x=2$，…，可解这个同余式. 这样

$x \pmod{11}$	0	1	2	3	4	5	6	7	8	9	10
$x^3 \pmod{11}$	0	1	8	5	9	4	7	2	6	3	10

因此同余式 $x^{103} \equiv 4 \pmod{11}$ 有解 $x \equiv 5 \pmod{11}$.

特例证明

现在我们准备证明费马小定理. 为说明证明方法，首先证明 $3^6 \equiv 1 \pmod 7$. 当然，无需给出这个事实的奇特证明，这是因为 $3^6 - 1 = 728 = 7 \cdot 104$. 不过，当尝试理解证明或构造证明时，通常可使用特定的数. 当然，思路是设计不是真正利用我们考察特定数的结果的证明，然后这种证明能够应用到一般情况.

为证明 $3^6 \equiv 1 \pmod 7$，我们由数 1，2，3，4，5，6 分别乘以 3 开始再模 7 化简. 结果列入下表：

$x \pmod 7$	1	2	3	4	5	6
$3x \pmod 7$	3	6	2	5	1	4

注意每个数 1，2，3，4，5，6 在第二行恰好重新出现一次. 所以，如果将第二行的所有数乘起来就得到与第一行所有数乘积相同的结果. 当然必须模 7. 因此，

$$\underbrace{(3\cdot 1)(3\cdot 2)(3\cdot 3)(3\cdot 4)(3\cdot 5)(3\cdot 6)}_{\text{第二行的数}} \equiv \underbrace{1\cdot 2\cdot 3\cdot 4\cdot 5\cdot 6}_{\text{第一行的数}} \pmod 7.$$

为节省空间，使用数 n 的阶乘的标准符号 $n!$（1，2，3，…，n 的乘积）. 换句话说，

$$n! = 1 \cdot 2 \cdot 3 \cdots (n-1) \cdot n.$$

同余式左边提出 6 个因数 3 得

$$3^6 \cdot 6! \equiv 6! \pmod 7.$$

注意到 $6!$ 与 7 互素，所以，可从两边消去 $6!$ 得 $3^6 \equiv 1 \pmod 7$，这正好是费马小定理.

一般性证明

现在我们准备证明一般性的费马小定理. $3^6 \pmod 7$ 证明的关键是乘以 3 再重排数 1，2，3，4，5，6(mod 7). 所以，我们首先验证下述断言：

断言 9.2 设 p 是素数，a 是任何整数且 $a \not\equiv 0 \pmod p$，则数

$$a, 2a, 3a, \cdots, (p-1)a \pmod p$$

与数

$$1, 2, 3, \cdots, (p-1) \pmod p$$

相同，尽管它们的次序不同.

证明 数列 a，$2a$，$3a$，…，$(p-1)a$ 包含 $p-1$ 个数，显然没有一个数被 p 整除. 假设从数列中取两个数 ja 与 ka，并假设它们同余，

$$ja \equiv ka \pmod p.$$

则 $p \mid (j-k)a$，因为假设 p 不整除 a，所以 $p \mid (j-k)$. 注意我们使用了第 7 章的素数整除性定理，该定理说明如果素数整除乘积则它整除一个因数. 另一方面，已知 $1 \leqslant j, k \leqslant p-1$，则 $|j-k| < p-1$. 仅有一个数的绝对值小于 $p-1$ 且被 p 整除，这个数是 0. 从而 $j=k$. 这表明 a，$2a$，$3a$，…，$(p-1)a$ 中的不同乘积对模 p 不同.

现在我们已知数列 a，$2a$，$3a$，…，$(p-1)a$ 包含 $p-1$ 个不同的非零值(mod p). 但仅有 $p-1$ 个不同的非零值(mod p)，即数 1，2，3，…，$(p-1)$. 因此，尽管这些数可能以不同次序出现，但数列 a，$2a$，$3a$，…，$(p-1)a$ 与数列 1，2，3，…，$(p-1)$ 必包含相同的数(mod p). 这就完成了断言的验证.

利用该断言，容易完成费马小定理的证明. 断言说明数列

$$a, 2a, 3a, \cdots, (p-1)a \pmod p \text{ 与数列 } 1, 2, 3, \cdots, (p-1) \pmod p$$

相同，所以第一个数列中数的乘积等于第二个数列中数的乘积：

$$a \cdot (2a) \cdot (3a) \cdots ((p-1)a) \equiv 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod p.$$

下面从左边提出 $p-1$ 个 a 得

$$a^{p-1} \cdot (p-1)! \equiv (p-1)! \pmod p.$$

最后我们看到 $(p-1)!$ 与 p 互素，因此从两边消去 $(p-1)!$ 就得到费马小定理

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

作用：判断素数

无需真正分解一个数，可用费马小定理证明这个数不是素数. 例如，有

$$2^{1\,234\,566} \equiv 899\,557 \pmod{1\,234\,567}.$$

这意味着 1 234 567 不是素数，因为如果它是的话，费马小定理告诉我们 $2^{1\,234\,566}$ 必同余于 1(mod 1 234 567). (如果你想了解如何计算 $2^{1\,234\,566} \pmod{1\,234\,567}$，不必犯愁，我们将在第 16 章讲述如何计算.) 因为 $1\,234\,567 = 127 \cdot 9721$，所以在这种情况下，我们实际上可求出一个因数. 考察数

$$m = 10^{100} + 37.$$

当计算 $2^{m-1} \pmod m$ 时可得

$$2^{m-1} \equiv 3\,626\,360\,327\,545\,861\,062\,487\,760\,199\,633\,583\,910\,836\,873\,253\,019\,151\,380\,128\,320\,824$$
$$091\,124\,859\,463\,579\,459\,059\,730\,070\,231\,844\,397 \pmod m.$$

由费马小定理我们再次推出 $10^{100} + 37$ 不是素数，但不知道如何求其因数. 在台式计算机上可快速验证它没有小于 200 000 的素因数. 有点奇怪的是容易写出这样的合数，但还找不出其(真)因数.

欧拉公式

依赖模m的指数

在前一章我们证明了费马小定理：如果 p 是素数且 $p \nmid a$，则 $a^{p-1} \equiv 1 \pmod{p}$. 如果 p 换成合数，结论就不正确了. 例如，$5^5 \equiv 5 \pmod{6}$，$2^8 \equiv 4 \pmod{9}$. 因此，我们问是否有依赖模 m 的指数使得

$$a^{???} \equiv 1 \pmod{m}.$$

首先，观察到：如果 $\gcd(a, m) > 1$，则这是不可能的. 为了说明原因，假设 $a^k \equiv 1 \pmod{m}$. 则对某整数 y，$a^k = 1 + my$，所以 $\gcd(a, m)$ 整除 $a^k - my = 1$. 换句话说，如果 a 的某个幂模 m 余 1，则必有 $\gcd(a, m) = 1$. 这提示我们观察与 m 互素的数的集合

$$\{a : 1 \leqslant a \leqslant m, \quad \gcd(a, m) = 1\}.$$

例如，

m	$\{a: 1 \leqslant a \leqslant m,\ \gcd(a,\ m) = 1\}$
1	$\{1\}$
2	$\{1\}$
3	$\{1, 2\}$
4	$\{1, 3\}$
5	$\{1, 2, 3, 4\}$
6	$\{1, 5\}$
7	$\{1, 2, 3, 4, 5, 6\}$
8	$\{1, 3, 5, 7\}$
9	$\{1, 2, 4, 5, 7, 8\}$
10	$\{1, 3, 7, 9\}$

欧拉函数

在 0 与 m 之间且与 m 互素的整数个数是个重要的量，我们赋予这个量一个名称：

$$\phi(m) = \#\{a : 1 \leqslant a \leqslant m, \quad \gcd(a, m) = 1\}.$$

函数 ϕ 叫做欧拉函数. 由前表可求得 $1 \leqslant m \leqslant 10$ 时的 $\phi(m)$ 值.

m	1	2	3	4	5	6	7	8	9	10
$\phi(m)$	1	1	2	2	4	2	6	4	6	4

注意 p 是素数时每个整数 $1 \leqslant a \leqslant p$ 都与 p 互素. 所以，对素数 p 有公式

$$\phi(p) = p - 1.$$

我们设法模拟费马小定理的证明. 例如，假设要求 7 的幂次模 10 余 1. 不取所有数 $1 \leqslant a < 10$，而是恰好取与 10 互素的数. 它们是

$$1, 3, 7, 9 \pmod{10}.$$

如果用 7 乘每个数得

$$7 \cdot 1 \equiv 7 \pmod{10}, \quad 7 \cdot 3 \equiv 1 \pmod{10},$$
$$7 \cdot 7 \equiv 9 \pmod{10}, \quad 7 \cdot 9 \equiv 3 \pmod{10}.$$

注意重排后取相同的数. 如果将它们乘起来就得到相同的乘积

$$(7 \cdot 1)(7 \cdot 3)(7 \cdot 7)(7 \cdot 9) \equiv 1 \cdot 3 \cdot 7 \cdot 9 \pmod{10}$$
$$7^4(1 \cdot 3 \cdot 7 \cdot 9) \equiv 1 \cdot 3 \cdot 7 \cdot 9 \pmod{10}.$$

消去 $1 \cdot 3 \cdot 7 \cdot 9$ 得 $7^4 \equiv 1 \pmod{10}$.

指数 4 从何而来呢？它等于 0 与 10 之间且与 10 互素的整数个数；即由于 $\phi(10) = 4$，所以指数等于 4. 这意味着下述公式成立.

欧拉公式

定理 10.1（欧拉公式） 如果 $\gcd(a, m) = 1$，则

$$a^{\phi(m)} \equiv 1 (\mathrm{mod}\ m).$$

证明 至此，我们已经确定了考察的数的正确集合，所以欧拉公式的证明几乎与费马小定理的证明一致. 令

$$1 \leqslant b_1 < b_2 < \cdots < b_{\phi(m)} < m$$

是 0 与 m 之间且与 m 互素的 $\phi(m)$ 个整数.

断言 10.2 如果 $\gcd(a, m) = 1$，则数列

$$b_1 a, b_2 a, b_3 a, \cdots, b_{\phi(m)} a (\mathrm{mod}\ m)$$

与数列

$$b_1, b_2, b_3, \cdots, b_{\phi(m)} (\mathrm{mod}\ m)$$

相同，尽管它们可能次序不同.

注：b_i与m互素，则b_i加上m的整数倍（即b_i+km）也与m互素。

a与m互素，b与m互素，则ab与m也互素，与m互素的数字模m的结果都在b_1到$b_{\varphi(m)}$之中，因而有——

断言的证明 注意到如果 b 与 m 互素，则 ab 也与 m 互素. 从而数列

$$b_1 a, b_2 a, b_3 a, \cdots, b_{\phi(m)} a (\mathrm{mod}\ m)$$

中的每个数同余于数列

$$b_1, b_2, b_3, \cdots, b_{\phi(m)} (\mathrm{mod}\ m)$$

中的一个数. 进而，每个数列都有 $\phi(m)$ 个数. 因此，如果能够证明第一个数列中的数对于模 m 不同，则就得到两个数列（重排后）相同.

假设从第一个数列中取两个数 $b_j a$ 与 $b_k a$，并假设它们同余，

$$b_j a \equiv b_k a (\mathrm{mod}\ m).$$

则 $m \mid (b_j - b_k) a$. 但是 m 与 a 互素，因而得到 $m \mid b_j - b_k$. 另一方面，b_j，b_k 在 1 与 m 之间，这蕴涵 $\mid b_j - b_k \mid \leqslant m - 1$. 仅有一个数的绝对值严格小于 m 且被 m 整除，这个数是 0. 从而 $b_j = b_k$. 这说明数列

$$b_1 a, b_2 a, b_3 a, \cdots, b_{\phi(m)} a (\mathrm{mod}\ m)$$

中的数模 m 不同，这就完成了断言成立的证明.

使用上述断言容易完成欧拉公式的证明. 断言说明数列

$$b_1 a, b_2 a, b_3 a, \cdots, b_{\phi(m)} a (\mathrm{mod}\ m)$$

与数列

$$b_1, b_2, b_3, \cdots, b_{\phi(m)} (\mathrm{mod}\ m)$$

是相同的，所以，第一个数列中数的乘积等于第二个数列中数的乘积：

$$(b_1 a) \cdot (b_2 a) \cdot (b_3 a) \cdots (b_{\phi(m)} a) \equiv b_1 \cdot b_2 \cdot b_3 \cdots b_{\phi(m)} (\mathrm{mod}\ m).$$

左边提出 $\phi(m)$ 个 a 得到

$$a^{\phi(m)} B \equiv B (\mathrm{mod}\ m), \quad \text{其中 } B = b_1 b_2 b_3 \cdots b_{\phi(m)}.$$

最后由于每个 b_i 与 m 互素，我们得 B 与 m 互素. 这表明可从两边消去 B 得到欧拉公式

$$a^{\phi(m)} \equiv 1 (\mathrm{mod}\ m).$$

φ函数公式

欧拉公式

$$a^{\phi(m)} \equiv 1 (\mathrm{mod}\ m)$$

是既优美又有力的结果. 但是，除非能找到计算 $\phi(m)$ 的有效方法，否则它的用途不能发挥出来. 显然，我们不想列出 1 到 $m-1$ 的所有整数来检查每个数是否与 m 互素. 例如，如果 $m \approx 1000$，会耗费许多时间，对 $m \approx 10^{100}$ 则是不可能的. 正像我们在前一章看到的，容易计算 $\phi(m)$ 的一种情况是 $m=p$ 为素数，这是因为每个整数 $1 \leqslant a \leqslant p-1$ 与 m 互素. 因此 $\phi(p)=p-1$.

当 $m=p^k$ 是素数幂次时，可容易推出 $\phi(p^k)$ 的类似公式. 不用设法计数 1 到 p^k 之间的与 p^k 互素的整数个数，而是由满足 $1 \leqslant a \leqslant p^k$ 的所有整数开始，然后丢弃与 p^k 不互素的整数.

数 a 什么时候与 p^k 不互素呢？p^k 仅有的因数是 p 的幂次，所以，当 a 被 p 整除时 a 不与 p^k 互素. 换句话说，

$$\phi(p^k) = p^k - \#\{a : 1 \leqslant a \leqslant p^k, p \mid a\}.$$

因此，必须计数 1 与 p^k 之间有多少个整数被 p 整除. 这是容易的，下述数是 p 的倍数：

$$p,\ 2p,\ 3p,\ 4p,\ \cdots,\ (p^{k-1}-2)p,\ (p^{k-1}-1)p,\ p^k.$$

它们有 p^{k-1} 个，这就给出公式

$$\phi(p^k) = p^k - p^{k-1}.$$

例如

$$\phi(2401) = \phi(7^4) = 7^4 - 7^3 = 2058.$$

这表明在 1 与 2401 之间有 2058 个整数与 2401 互素.

当 m 是素数幂次时，我们已知如何计算 $\phi(m)$. 下面假设 m 是两个素数幂次的乘积 $m=p^jq^k$. 要将猜测公式化，我们对一些小的值计算 $\phi(p^jq^k)$ 并将它与 $\phi(p^j)$ 和 $\phi(p^k)$ 的值进行比较.

p^j	q^k	p^jq^k	$\phi(p^j)$	$\phi(q^k)$	$\phi(p^jq^k)$
2	3	6	1	2	2
4	5	20	2	4	8
3	7	21	2	6	12
8	9	72	4	6	24
9	25	225	6	20	120

这个表揭示了 $\phi(p^jq^k)=\phi(p^j)\phi(q^k)$. 我们也可试一些不是素数幂次的数的例子，如

$$\phi(14) = 6,\quad \phi(15) = 8,\quad \phi(210) = \phi(14 \cdot 15) = 48.$$

这使得我们猜测下述断言成立：

如果 $\gcd(m,n) = 1$，则 $\phi(mn) = \phi(m)\phi(n)$.

在设法证明这个乘法公式之前，对任意的 m，或更确切地对能够分解成素数乘积的任意 m，我们说明利用它能够多么容易地计算 $\phi(m)$.

假设已知整数 m，且假设已将 m 分解成素数乘积，即

$$m = p_1^{k_1} \cdot p_2^{k_2} \cdots p_r^{k_r},$$

其中 $p_1, p_2, \cdots, p_r$ 是不同的素数. 首先使用乘法公式计算

$$\phi(m) = \phi(p_1^{k_1}) \cdot \phi(p_2^{k_2}) \cdots \phi(p_r^{k_r}).$$

然后使用素数幂公式 $\phi(p^k)=p^k-p^{k-1}$ 得

$$\phi(m) = (p_1^{k_1} - p_1^{k_1-1}) \cdot (p_2^{k_2} - p_2^{k_2-1}) \cdots (p_r^{k_r} - p_r^{k_r-1}).$$

这个公式看起来复杂，但实际上计算 $\phi(m)$ 的过程很简单. 例如，

$$\begin{aligned}\phi(1512) &= \phi(2^3 \cdot 3^3 \cdot 7) = \phi(2^3) \cdot \phi(3^3) \cdot \phi(7) \\ &= (2^3 - 2^2) \cdot (3^3 - 3^2) \cdot (7-1) = 4 \cdot 18 \cdot 6 = 432.\end{aligned}$$

所以，在 1 与 1512 之间有 432 个数与 1512 互素.

现在准备证明欧拉函数的乘法公式. 我们也重述素数幂公式以使两个公式方便地列在一起.

定理 11.1（ϕ 函数公式） （a）如果 p 是素数且 $k \geqslant 1$，则

$$\phi(p^k) = p^k - p^{k-1}.$$

（b）如果 $\gcd(m, n) = 1$，则 $\phi(mn) = \phi(m)\phi(n)$.

证明 我们在本章前面证明了素数幂公式（a），因此余下的便是要证明乘法公式（b）. 我们使用数论中的

计数

这个最有用的工具之一来进行证明. 简言之，我们找一个包含 $\phi(mn)$ 个元素的集合，再找一个包含 $\phi(m)\phi(n)$ 个元素的第二个集合，然后证明这两个集合包含个数相同的元素.

第一个集合是

$$\{a : 1 \leqslant a \leqslant mn, \quad \gcd(a, mn) = 1\}.$$

显然，这个集合包含 $\phi(mn)$ 个元素，因为这正好是 $\phi(mn)$ 的定义. 第二个集合是

$$\{(b,c) : 1 \leqslant b \leqslant m, \quad \gcd(b,m) = 1, \quad 1 \leqslant c \leqslant n, \quad \gcd(c,n) = 1\}.$$

第二个集合含有多少个序对 (b, c) 呢？正好对 b 有 $\phi(m)$ 个选择，因为这是 $\phi(m)$ 的定义. 对 c 有 $\phi(n)$ 个选择，因为这是 $\phi(n)$ 的定义. 所以，对第一个坐标 b 有 $\phi(m)$ 个选择，对第二个坐标 c 有 $\phi(n)$ 个选择，从而对序对 (b, c) 总共有 $\phi(m)\phi(n)$ 个选择.

例如，假设取 $m = 4$ 与 $n = 5$. 则第一个集合由与 20 互素的数

$$\{1,3,7,9,11,13,17,19\}$$

组成. 第二个集合由序对

$$\{(1,1),(1,2),(1,3),(1,4),(3,1),(3,2),(3,3),(3,4)\}$$

组成，其中每个序对的第一个数与 4 互素，第二个数与 5 互素.

回到一般情形，我们取第一个集合的每个元素，按照下述方法

$$\left\{a : \begin{array}{l} 1 \leqslant a \leqslant mn \\ \gcd(a,mn) = 1 \end{array}\right\} \longrightarrow \left\{(b,c) : \begin{array}{l} 1 \leqslant b \leqslant m, \quad \gcd(b,m) = 1 \\ 1 \leqslant c \leqslant n, \quad \gcd(c,n) = 1 \end{array}\right\}$$

$$a \bmod mn \qquad \longmapsto \qquad (a \bmod m, a \bmod n)$$

将它与第二个集合的序对对应. 这指的是取第一个集合的整数 a 并把它指派到序对 (b, c)，满足

$$a \equiv b \pmod m \quad \text{与} \quad a \equiv c \pmod n.$$

如果再看一下 $m = 4$，$n = 5$ 的例子，也许比较清楚. 例如，第一个集合的数 13 与第二个集合的序对 $(1, 3)$ 对应，因为 $13 \equiv 1 \pmod 4$ 且 $13 \equiv 3 \pmod 5$. 对第一个集合的其他数采用同样的做法.

$$\{1,3,7,9,11,13,17,19\} \longrightarrow \{(1,1),(1,2),(1,3),(1,4),(3,1),(3,2),(3,3),(3,4)\}$$

$$1 \longmapsto (1,1) \quad 3 \longmapsto (3,3) \quad 7 \longmapsto (3,2) \quad 9 \longmapsto (1,4)$$

$$11 \longmapsto (3,1) \quad 13 \longmapsto (1,3) \quad 17 \longmapsto (1,2) \quad 19 \longmapsto (3,4)$$

在这个例子中，可以看到第二个集合的每个序对恰好与第一个集合的一个元素匹配. 这表明两个集合有相同的元素个数. 我们要证明一般情形下的同样匹配出现.

我们需要证明下面两个陈述是正确的：

（1）第一个集合的不同数对应第二个集合的不同序对.

（2）第二个集合的每个序对适合第一个集合的某个数.

一旦验证了这两条，我们就知道两个集合有相同的元素个数. 但是，已知第一个集合有 $\phi(mn)$ 个元素，第二个集合有 $\phi(m)\phi(n)$ 个元素. 所以，为了完成 $\phi(mn)=\phi(m)\phi(n)$ 的证明，只需验证(1)与(2).

要验证(1)，我们取第一个集合的两个数 a_1 与 a_2，假设它们在第二个集合有相同的象. 这意味着

$$a_1 \equiv a_2 \pmod{m} \quad 与 \quad a_1 \equiv a_2 \pmod{n}.$$

因此，a_1-a_2 被 m 与 n 整除. 然而，m 与 n 互素，因此 a_1-a_2 一定被 mn 整除. 换句话说，

$$a_1 \equiv a_2 \pmod{mn},$$

这表明 a_1 与 a_2 是第一个集合的相同元素. 这就完成了第一个陈述的证明.

要验证陈述(2)，需要证明对 b 与 c 的任何已知值，至少可求得一个整数 a 满足

$$a \equiv b \pmod{m} \quad 与 \quad a \equiv c \pmod{n}.$$

这个同余式组有解的事实是很重要的，足以保证它有自己的名称. □

定理 11.2（中国剩余定理） 设 m 与 n 是整数，$\gcd(m, n)=1$，b 与 c 是任意整数. 则同余式组

$$x \equiv b \pmod{m} \quad 与 \quad x \equiv c \pmod{n}$$

恰有一个解 $0 \leqslant x \leqslant mn$.

证明 像通常一样，我们由例子开始. 假设要解

$$x \equiv 8 \pmod{11} \quad 与 \quad x \equiv 3 \pmod{19}.$$

第一个同余式的解由形如 $x=11y+8$ 的所有整数组成. 将它代入第二个同余式，化简并求解. 因此，

$$11y+8 \equiv 3 \pmod{19}$$
$$11y \equiv 14 \pmod{19}.$$

我们知道怎样解这种类型的线性同余式组（见第 8 章线性同余式定理）. 解是 $y_1 \equiv 3 \pmod{19}$，然后可用 $x_1=11y_1+8=11\cdot 3+8=41$ 求得原来同余式的解. 最后验证答案：$(41-8)/11=3$ 与 $(41-3)/9=2$ 是正确的.

对一般情况，由解第一个同余式 $x \equiv b \pmod{m}$ 开始. 其解由形如 $x=my+b$ 的所有数组成. 将此代入第二个同余式得

$$my \equiv c-b \pmod{n}.$$

已知 $\gcd(m, n)=1$，第 8 章线性同余式定理告诉我们恰有一个解 y_1，$0 \leqslant y_1 < n$. 则

$$x_1 = my_1 + b$$

给出了原来同余式组的解，这是唯一解 x_1，$0 \leqslant x_1 < mn$，因为在 0 与 n 之间有唯一解 y_1，且用 m 乘 y_1 得 x_1. 这就完成了中国剩余定理的证明及公式 $\phi(mn)=\phi(m)\phi(n)$ 的证明. □

历史插曲 中国剩余定理的第一个有记载的实例出现在 3 世纪末 4 世纪初的中国数学著作中. 令人惊讶的是，它涉及解由三个同余式构成的同余式组这样的难题.

“今有物不知其数. 三三数之剩二，五五数之剩三，七七数之剩二，问物几何？”

——《孙子算经》(孙子的数学著作)，大约公元 300 年，第 3 卷问题 26

补充

在数论中，对正整数n，**欧拉函数φ(n)**是<=n的正整数中与n互质的数的数目。

φ念fai，四声。

此函数以其首名研究者欧拉命名，它又称为φ函数（由高斯所命名）。

例如，φ(8)=4，因为1,3,5,7均与8互质。

φ(1)被定义为1，但是并没有任何实质的意义。

当m = p为素数时，φ(p) = p – 1；

当m = p^k为素数幂次时，φ(p) = p^k – p^{k-1}。

更普遍地，如果gcd(m, n) = 1，φ(mn) =φ(m)φ(n)。

$$\varphi(x) = \varphi(p_1^{k1}p_2^{k2}\dots p_n^{kn})$$

$$= \varphi(p_1^{k1})\varphi(p_2^{k2})\dots\varphi(p_n^{kn})$$

$$= (p_1^{k1} - p_1^{k1-1})(p_2^{k2} - p_2^{k2-1})\dots(p_n^{kn} - p_n^{kn-1})$$

$$= p_1^{k1-1}(p_1 - 1)p_2^{k2-1}(p_2 - 1)\dots p_n^{kn-1}(p_n - 1) \quad \text{式子①}$$

$$= p_1^{k1}(1 - 1/p_1)p_2^{k2}(1 - 1/p_2)\dots p_n^{kn}(1 - 1/p_n)$$

$$= x(1 - 1/p_1)(1 - 1/p_2)\dots(1 - 1/p_n) \quad \text{式子②}$$

$$= x(\frac{p_1 - 1}{p_1})(\frac{p_2 - 1}{p_2})\dots(\frac{p_n - 1}{p_n}) \quad \text{式子③}$$

在数论中，欧拉定理（也称**费马-欧拉定理**或**欧拉φ函数定理**）是一个关于同余的性质。欧拉定理表明，若 *n*, *a*为正整数，且*n*, *a* 互素（即 gcd(a,n)=1），则

$$a^{\varphi(n)} \equiv 1 \pmod{n}$$

即$a^{\varphi(n)}$与1在模n下同余。

欧拉定理得名于瑞士数学家**莱昂哈德·欧拉**。

欧拉定理实际上是费马小定理的推广。

编码

利用式子②编码：

```
// 代码1
int phi(int n)
{
    int res = n;
    for(int i = 2; i <= n; i++)
    {
        if(n % i == 0)
        {
            n/=i;
            res = res - res / i;
        }
        while(n % i == 0)
            n/=i;
    }
    return res;
}
```

由分析可以知道，这个函数的时间复杂度为O(n)，当n达到1e9，肯定会超时。由于任何一个合数都存在着至少一个不大于sqrt(n)的质因数，所以只需遍历到sqrt(n)即可，这样时间复杂度为O(sqrt(n))。

```
// 代码2
int phi(int n)
{
    int res = n;
    for(int i = 2; i * i <= n; i++) // 降低时间复杂度
    {
        if(n % i == 0)
        {
            n/=i;
            res = res - res / i;
        }
        while(n % i == 0)
            n/=i;
    }
    if(n > 1) // 因为是遍历到sqrt(n)，所以可能存在未除尽或者n本身就为质数的情况
        res = res - res / n;
    return res;
}
```

利用式子①编码：

```
// 代码3
int eular(int n)
{
    int ret=1, i;
    for (i=2; i*i<=n; i++)
    {
        if (n%i==0)
        {
            n /= i, ret *= i-1;
            while (n%i==0)
                n/=i, ret*=i;
        }
    }
    if (n>1) ret *= n-1;// n-1 是最后一个素数
    return ret;
}
```

时间复杂度为O(sqrt(n))。

上述三份代码，从代码2或代码3中选择其一写熟即可。

求n以内所有数字的欧拉函数：

```
// 代码4
#include <iostream>
using namespace std;

const int MAX = 1024;
int N;
int p[MAX], phi[MAX];

```

```cpp
int main()
{
    cin >> N;
    for(int i = 1; i <= N; i++)// 初始化
    {
        p[i] = 1;
        phi[i] = i;
    }

    p[1] = 0;// 1不是素数

    for(int i = 2; i <= N; i++)// 筛素数
    {
        if(p[i])
        {
            for(int j = i * i; j <= N; j += i)// j的初值不是i+i，因为这种情况在i=2时已经覆盖过
                p[j] = 0;
        }
    }

    for(int i = 2; i <= N; i++)// 求欧拉函数
    {
        if(p[i])
        {
            for(int j = i; j <= N; j += i)// 处理素因子i
            {
                phi[j] = phi[j] / i * (i - 1);// 根据式子③，先除后乘，防止中间过程超出范围
            }
        }
    }

    cout << "Primes: " << endl;
    for(int i = 1; i <= N; i++)
        if(p[i])
            cout << i << " ";
    cout << endl;
    cout << "Euler Phi Function: " << endl;
    for(int i = 1; i <= N; i++)
        cout << phi[i] << " ";
    cout<<endl;

    return 0;
}
```

线性筛的写法：

```cpp
// 代码5
#define N 2000001
int phi[N];
int cnt,prime[N/10];

void Get_ol()
{
    for (int i=2;i<N;i++)
    {
```

```
        if (!p[i])
        {
            prime[++cnt]=i;
            phi[i]=i-1;// 当i是素数时，ϕ(i)=i-1
        }
        for (int j=1;j<=cnt&&i*prime[j]<N;j++)
        {
            p[i*prime[j]]=1;
            if (i%prime[j]==0)// 线性筛法
            {
                phi[i*prime[j]]=phi[i]*prime[j];// gcd(i,prime[j])=prime[j]，ϕ(p^k)=(p-1)p^(k-1)=p*(p-1)p^(k-2)=p*ϕ(p^(k-1))
                break;
            }
            phi[i*prime[j]]=phi[i]*phi[prime[j]];// 由现在推未来，i与prime[j]互素，运用ϕ(mn) =ϕ(m)ϕ(n)，phi[prime[j]]=prime[j]-1
        }
    }
}
```

其它写法：

```
// 代码6
// hdu2824
void euler()
{
    int i,j;
    for(i=1;i<=3000000;i++)
        a[i]=i;
    for(i=2;i<=3000000;i++)
        if(a[i]==i)
        {
            for(j=i;j<=3000000;j=j+i)
                a[j]=a[j]/i*(i-1);
        }
}
```