Number Theory 1

整除

$b|a$: 设 $a,b \in Z$, 若 $\exists q \in Z$, 使 $a = qb$, 则称 "b 整除 a"

- 例 : $3|9$，$3|6$，$5|25...$

1. (自反性) $a|a$
2. (传递性) $b|a$ 且 $a|c$, 则 $b|c$
3. (相乘性) $b|a$, 则 $bc|ac$
4. (消去性) $bc|ac$ 且 $c \neq 0$, 则 $b|a$
5. (线性性) $b|a$ 且 $b|c$,对于所有的 $s,t \in Z$, 都有 $b|(sa \pm tc)$
6. (比较性) 如果 $a,b \in N$ 且 $b|a$, 则 $b \leq a$

Prime 素数

定义: 设 $n \in Z$ 且 $n \geq 2$ ，除了 `1` 和 `n` 以外，没有其他正整数整除 n，则 `n` 称作“素数”(通常用 `p` 表示) ;
否则， n 称作“合数”

引理 : 任何 > 1 的整数都有**素因子(素数因子)**

- 素数: 当然没问题, 因为素数本身就是他自己的 素因子 ;
- 合数: 稍微有点麻烦, 可以用 `反证法 + 突破边界` 来证明 :
  - 假设有个集合 S，S 内整数没有素因子，然后设立 m 为集合中最小的整数（下界）
  - 由于 m 不是素数（因为集合中的数不能有素因子），则 m 为合数，既然是合数，则 m 可以表示成 $m = ab$ 的形式，且 a 和 b 都比 m 小，既然比 m 小，也就是突破了集合的边界，a 和 b 必有一个素因子，根据传递性 m 也有素因子，则不满足集合情况，反证法成功

引理 : 任何合数 n 都至少有一个不大于 $\sqrt{n}$ 的素因子
即: 若 $n \geq 1$ 是合数, 则存在 Prime p , 使得 $p <= \sqrt{n}$

此引理可用来精简判断 n 是否是素数 :

- 如果所有的素数 $p <= \sqrt{n}$ 都不能整除 `n`, 则 `n` 是素数
- 比如整数 97, $\because 9 < \sqrt{97} < 10$, 即不超过 10 的素数有 2, 3, 5 ,7, 它们都不能整除 97, 所以 97 是素数。

算术基本定理

算术基本定理，又称为正整数的唯一分解定理，即：

- 每个大于 1 的自然数，要么本身就是质数，要么可以写为 2 个或以上的质数的积，而且这些质因子按大小排列之后，写法仅有一种方式。
- 如 : $1200 = 2^4 \times 3 \times 5^2$; $6936 = 2^3 \times 3 \times 17^2$; $5207 = 41 \times 127$ 都只有一种表示形式

算术基本定理的内容由两部分构成：

- 分解的存在性：
- 分解的唯一性，即不考虑排列的顺序，正整数分解为素数乘积的方式是唯一的。

利用"算数基本定理" 证明 素数有无穷多个 ：

证明 1 :

- 假设素数的个数是有限的，总共有 n 个，分别为 $p_1, p_2, \cdots, p_n$
- 根据算数基本定理，P 能表示成一系列素数的乘积即 $\Rightarrow \exists \mathrm{k} \in \{1, 2, \cdots, \mathrm{n}\}$, $\mathrm{s.t.\ p_k} \mid \mathrm{P}$

$$\begin{cases} \mathrm{p_k} \mid \mathrm{P} = \mathrm{p_1 p_2 \cdots p_n} + 1 \\ \mathrm{p_k} \mid \mathrm{p_1 p_2 \cdots p_n} \end{cases} \Rightarrow \mathrm{p_k} \mid 1$$

证明 1 :

证明: 假设素数只是有限个: $P_1, \ldots, P_k$ 。

> 设 $M = p_1 \cdot p_m \cdot \ldots \cdot p_k$， 设 $N = M + 1$ ，很明显 $N >= 2$
> 根据算术基本定理，N 可以表示成一系列素数乘积的形式，
> 则存在一个素数 p，满足 $p|N$
> 且有 $p \neq P_1, \ldots, P_k$, (否则 $p|M$ & $p|N$, 导致 $p|(N - M)$ ⇒ $p|1$，这不可能)
> 所以，p 不在 $P_1, \ldots, P_k$ 之中，这与假设相矛盾。

Mod 模运算

负数求模 (类似把时钟向回拨)

> 如 $-40 \ mod \ 11$ 只需要不断将 -40 加 11 , 直到该数变成正数为止 ;

模运算的性质 :

模运算与基本四则运算有些相似，但是除法例外。其规则如下：

1. $(a + b) \ \% \ p = (a \ \% \ p + b \ \% \ p) \ \% \ p$ (1)
2. $(a - b) \ \% \ p = (a \ \% \ p - b \ \% \ p) \ \% \ p$ (2)
3. $(a * b) \ \% \ p = (a \ \% \ p * b \ \% \ p) \ \% \ p$ (3)
4. $(a^b) \ \% \ p = ((a \ \% \ p)^b) \ \% \ p$ (4)

栗子 :

$$
\begin{aligned}
&(152 + 131 * 81 - 3) \ mod \ 5 \\
=\ &(152 \ mod \ 5 + (131 \ mod \ 5) * (81 \ mod \ 5) - 3) \ mod \ 5
\end{aligned}
$$

辗转相除法

$gcd(12, 18) == 6$

互素 :
设 $a, \ b \in Z$, 若 $gcd(a, b) == 1$ 则称 a, b 互素

辗转相除法 :

$$
\begin{aligned}
&gcd(100, \ 35) \\
&= gcd(35, \ 30) \\
&= gcd(30, \ 5) \\
&= gcd(5, \ 0) = 5
\end{aligned}
$$

原理 :

$$
\begin{aligned}
&a = r_0, \quad b = r1 \\
&r_0 = r_1 \cdot q_1 \ + \ r_2 \\
&r_1 = r_2 \cdot q_2 \ + \ r_3 \\
&\ldots \\
&r_{i-1} = r_i \cdot q_i \ + \ r_{i+1} \\
&\ldots \\
&r_{n-2} = r_{n-1} \cdot q_{n-1} \ + \ r_n \\
&r_{n-1} = r_n \cdot q_n \\
\Longrightarrow \ &r_n = gcd(a, b) \quad r_n \text{ 就是最大公约数}
\end{aligned}
$$

举例 :

$$
\begin{aligned}
&gcd(100, 35) \\
&100 \ = \ r_0, \ 35 = r_1 \\
\\
&r_0 = r_1 \cdot q_1 \ + \ r_2 \\
&\text{即 } 100 = q_1 \cdot 35 \ + \ r_2 \quad \Longrightarrow q_1 == 2, \ r_2 == 30 \\
&- - - - - - - - \\
&r_1 = r_2 \cdot q_2 \ + \ r_3 \\
&\text{即 } 35 = q_2 \cdot 30 \ + \ r_3 \quad \Longrightarrow q_2 == 1, \ r_3 == 5 \\
&\ldots \\
&r_2 = r_3 \cdot q_3 \\
&\text{即 } 30 = r3 * q_n \quad \text{发现能整除了} \\
\\
\Longrightarrow \ &r_n = gcd(a, b) \quad r_n == 5 \text{ 就是最大公约数}
\end{aligned}
$$

欧几里得算法 Euclidean Algorithm

欧几里得算法 就是 辗转相除法（Euclidean Algorithm）

下面是一个例子，说明如何使用欧几里得算法求解 30 和 21 的最大公约数：

1. 30 除以 21 的余数为 9 。
2. 21 除以 9 的余数为 3。
3. 9 除以 3 的余数为 0 ，所以 3 即为最大公约数。

> 欧几里得算法的效率非常高，对于任意两个正整数，最多只需要执行log(a+b)次运算即可求出它们的最大公约数。因此，欧几里得算法被广泛应用于计算机科学和数学领域。

贝祖等式/定理

贝祖等式是数论中的一个重要定理，它表明对于任意两个大于 1 的正整数 a 和 b，它们的**最大公约数（GCD）与最小公倍数（LCM）的乘积等于这两个数的乘积** 。即：

$$GCD(a,b) \times LCM(a,b) = a \times b$$

以下是一个贝祖等式的例子：

假设我们要求 18 和 24 的最大公约数和最小公倍数。首先，我们可以使用欧几里得算法来计算它们的最大公约数：

$18 \div 24 = 0 ... 18 (24 > 18)$
$24 \div 18 = 1 ... 6 (18 < 24)$
$18 \div 6 = 3 ... 0 (24 > 6)$
因此，$GCD(18,24) = 6$。

接下来，我们可以使用以下公式来计算它们的最小公倍数：

$$LCM(18,24) = |18 \times 24| \div GCD(18,24) = 432 \div 6 = 72$$

现在，我们可以将贝祖等式应用于这个例子中：

$$GCD(18,24) \times LCM(18,24) = 6 \times 72 = 432 18 \times 24 = 432 \quad == 18 \times 24$$

这个例子证明了贝祖等式的正确性。

拓展欧几里得

> 没看懂, 先不管了

拓展欧几里得算法是一种求解两个数的 **最大公约数**（Greatest Common Divisor，简称 GCD）以及它们的 **贝祖等式**（Bézout's identity）的算法。

拓展欧几里得算法的基本思想是通过 **递归** 计算两个数的最大公约数，并在每一步中计算出一个关于 x 和 y 的线性组合，以便最终得到贝祖等式中的 x 和 y 值。

具体来说，要求解 a 和 b $(a \geq b)$ 的最大公约数 `gcd(a,b)`，可以执行以下步骤：

1. 若 b = 0，则 `gcd(a,b) = a`，此时 x = 1，y = 0，返回结果；
2. 否则，我们用 a 除以 b 得到商 q 和余数 r，即构造 $a = bq + r$。根据欧几里得(辗转相除)算法，$gcd(a,b) = gcd(b,r)$。 比如 $gcd(32,18) = gcd(18,4) = gcd(4,2) = 2$
3. 然后我们可以递归地计算 `gcd(b,r)` 并得到相应的 x_1 和 y_1 值，即：
   $gcd(b,r) = bx_1 + ry_1$ 如 $gcd(18,4) = 18 \cdot x_1 + 4 \cdot y_1$
4. 现在我们将 x 和 y 表示为 x = y1 和 y = x1 - qy1 - ry1，即： (啥啥啥???)

$x = y1$
$y = x1 - qy1 - ry1$

5. 最后返回 $gcd(a,b)$ 和对应的 x 和 y 值，即：

$gcd(a,b) = gcd(b,r)$
$x = y1$
$y = x1 - qy1 - ry1$

这样我们就得到了 a 和 b 的最大公约数以及对应的 x 和 y 值，满足贝祖等式 ax + by = gcd(a,b)。

最小公倍数 (Least common Multiple)

比如 $lcm(6,8)=24, lcm(3,5)=15$

奇妙的定理：

$$lcm(a,b) * gcd(a,b) = ab$$

比如

$$\begin{aligned} & lcm(4,6) * gcd(4,6) \\ & = 12 * 2 \\ & = 24 == 4 * 6 \end{aligned}$$

证明: 略.

等价关系 equivalence relation

例如, 对于实数集合 R , 二元关系 $=$, 举个超级简单的例子：

- 自反性：对于所有的 $a \in R$, 都有 $a = a$
- 对称性：对于所有的 $a, b \in R$, 都有 $a = b => b = a$
- 传递性：对于所有的 $a, b, c \in R$, 都有 $a = b, b = c \ \ => \ \ a = c$

定义等价关系：设集合 S, 定义在 S 上的二元关系 Rls , 如果 R 满足以下性质 , 则称它为 "等价关系"：

- 自反性：对于所有 $a \in S$, 都有 $(a, a) \in Rls$, 常写作 $X \sim X$
- 对称性：对于所有 $a, b \in S$, 都有 $(a, b) \in Rls \ => \ (b, a) \in Rls$, 常写作 $X \sim Y, \ \ => \ \ Y \sim X$
- 传递性：对于所有的 $a, b, c \in S$, 都有 $(a, b) \in Rls, \ (b, c) \in Rls \ \ => \ \ (a, c) \in Rls$
  - 常写作 $X \sim Y, \ \ Y \sim Z, \ \ => \ \ X \sim Z$

同余 congruence

设 n 为正整数，整数 a 和 b 分别模 n ,如果得到相同的余数，就称 a 和 b 在模 n 下满足同余关系(congruence relation)， 简称同余。 同余可以写作如下形式：

$$\begin{aligned} & a \equiv b \pmod{m} \\ \text{or} \quad & a \equiv_m b \end{aligned}$$

- Reflexivity 自反性: $a \equiv_m a$
- Symmetry 对称性: If $a \equiv_m b$, then $b \equiv_m a$
- Transitivity 传递性: If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$

定义 同余关系：设 $a, b, n \in Z, \ n > 0$, 如果 $n | a - b$, 就称 a 和 b 在模 n 下同余, 记作 $a \equiv_m b$

证明：

$$\begin{aligned} & 令 \quad a = q_1 n + r_1, \\ & \quad\ \ b = q_2 n + r_2 \end{aligned}$$

$$则有 \quad a - b = (q_1 - q_2)n + (r_1 - r_2)$$

$$IF \quad a、b \text{ 余数相同，即 } r1 == r2$$

$$==> \quad a - b = (q_1 - q_2)n \quad 即 \quad n | a - b$$

重要性质:
$a \equiv_m b \quad <==> \quad a = qn + b, \quad \exists q \in Z$

---

congruence 关系是一种 `equivalent relation`

- 自反性：对于所有的 $a \in Z$, 都有 $a \equiv a \pmod{n}$
- 对称性：对于所有的 $a, b \in Z$ 都有：
  - $a \equiv b \pmod{n} \quad => \quad b \equiv a \pmod{n}$
- 传递性：对于所有的 $a, b, c \in Z$, 都有：
  - $a \equiv b$, $b \equiv c$, $\quad => \quad a \equiv c \pmod{n}$

同余的运算性质

$$
\begin{aligned}
&\text{对于} \quad 15 \equiv_6 8 \\
\\
&15 + m \equiv_6 9 + m \\
&15 - m \equiv_6 9 - m \\
&15 \times m \equiv_6 9 \times m \\
&15^m \equiv_6 9^m
\end{aligned}
$$

乘法逆元 Modular Inverse

在数学中，给定一个元素 a 和一个数 n，如果存在一个元素 b，使得 a 和 b 的乘积模 n 等于 1，则我们称 b 是 a 关于模 n 的乘法逆元，通常用 a 的倒数表示，即 $b = a^{-1}$ 。

举个例子：计算 `14/4 mod 5`：

$$
\begin{aligned}
&14/4 \mod 5 \\
&= 7/2 \\
&= 7 \times \frac{1}{2} \mod 5 \\
&= \dots
\end{aligned}
$$

计算到这里，模运算下最后的计算结果必然是一个整数, 所以 $\frac{1}{2}$ 必然要进行转化：
" $7 \times \frac{1}{2} \mod 5$ " 就可以理解为 " 7×2的乘法逆元 $\mod 5$ "

> 可以用倒数来类比

因为 $2 \times 3 \mod 5 == 1$, 所以 3 是 2 在模 5 下的乘法逆元 (倒数)
即 $2^{-1} == 3 \quad (mod\ 5)$

$$
\begin{aligned}
&= 7 \times \frac{1}{2} \mod 5 \\
&= 7 \times 2^{-1} \mod 5 \\
&= 7 \times 3 \mod 5 \\
&= 21 \mod 5 == 1
\end{aligned}
$$

为了加深记忆, 请计算 $2^{-1} \mod 7$??

$$
\begin{aligned}
&2^{-1} \mod 7 \\
&<= 2 \times [\text{谁}] \mod 7 = 1 \ ? \\
&<= 2 \times 4 \mod 7 = 1 \\
&\therefore 2^{-1} \mod 7 = 4
\end{aligned}
$$

再来看几个

$$
\begin{aligned}
&2^{-1} \mod 7 = 4 \\
&2^{-1} \mod 5 = 3 \\
&3^{-1} \mod 7 = 5
\end{aligned}
$$

现在，我们来详细解释下乘法逆元的定义和性质。

定义: 设 $a \in Z,\ z \in N,\ If \quad az \equiv 1 \mod n$, 则称 z 是模 n 下 a 的乘法逆元, 记作 $a^{-1} = z$

- a 的乘法逆元是 $z = a^{-1}$
- $z^{-1} = (a^{-1})^{-1} = a$

首先，我们要注意到，如果 n 不是质数，那么可能存在某些元素 a 没有乘法逆元。例如，如果 $n = 6$，那么元素 2 没有乘法逆元，因为 $2 \times \ ? \ = 1 \mod 6$, 这个数不存在。因此，我们在定义乘法逆元时，通常要求 n 是一个质数。

一次同余方程

首先看一个错误例子：

$$
\begin{aligned}
&9 \equiv 3 \mod 6 \\
&9/3 \equiv 3/3 \mod 6 \qquad \text{为什么同时除3不行?} \\
&9 \times 3^{-1} \equiv 3 \times 3^{-1} \mod 6 \qquad \text{3和6不互质, 所以 3 在 } mod\ 6 \text{没有逆元} \\
&\qquad \text{所以} \uparrow \text{式是乘了一个不存在的东西, 当然不行}
\end{aligned}
$$

把 $9 \equiv 3 \mod 6$ 变换一下, 变成等式的形式：

$$
\begin{aligned}
9 &\equiv 3 \mod 6 \\
9 &= q*6+3 \qquad \text{同时除3} \\
9/3 &= q*6/3+3/3 \\
3 &= 2q+1
\end{aligned}
$$

即 $3 \equiv 1 \mod 2$
即 $(9/3 \equiv 3/3 \mod 6/3)$

如例：$21x \equiv 14 \mod 35$

- 提取 a = 21，b = 14，m = 35
- `gcd(a, m) = gcd(21, 35) = 7`。因为 7 整除 14，所以我们可以将方程两边同时除以 7，
- 得到新的同余方程 $3x \equiv 2 \pmod 5$
- 由于 3 和 5 互质，所以它们的最大公约数是 1，求得 x == 4

引出：同余下的消去律
设 $a,n \in Z$，$n>0$，如果 $gcd(a,n)=d$，则有：

$$
az \equiv az' \pmod n \Rightarrow z=z' \pmod{n/d}
$$

> - 如果 a 和 n 的最大公约数是 d，就可以把 a 从式子两边直接消掉；
> - 并把模数变成 n 除以 d

再看一个例子：

$$
\begin{aligned}
105x &\equiv 63 \mod 6 \\
105x/21 &\equiv 63/21 \mod 6/3 \\
&\uparrow \ 21 \text{ 和 } 6 \text{ 的最大公因数是 } 3 \\
5x &\equiv 3 \mod 2 \\
x &\equiv 3/5 \mod 2 \\
&\text{计算 } 5 \text{ 在 } mod \ 2 \text{ 下的乘法逆元，是 } 1 \text{，因为} (5\times 1)\%2 == 1 \\
x &\equiv 3\times 1 \mod 2 \\
\text{化简得} \quad x &\equiv 1 \mod 2
\end{aligned}
$$

但是最后的答案并不是 $x \equiv 1$，因为这个式子表示的是：x 和 1 在模 2 下同余，任何一个➗ 2 == 1 的整数都是方程的解

- 解集：`{..., -3, -1, 1, 3, 5, 7 ...}`
- 1 ,3 ,5 < 模数 6；1,3,5 是 3 个数；6/3 == 2 (2 是最终模数)

规律：
原模数为 n，最终模数为 n'，设 n/n' = d，有：0 ~ n-1 之间解的数量恰好 == d

剩余类

剩余类，也叫模意义下的剩余系，是指对于给定的正整数模数 n，所有模 n 意义下不同的整数构成的集合。其中，模 n 意义下的整数是指与n的除数关系相同的整数，即模 n 意义下等价的整数。

举个例子，假设我们要考虑模5意义下的剩余类。那么，对于任意一个整数 x，它与 5 的除数关系可以表示为：

$$
x \equiv a \pmod 5
$$

其中 a 是一个整数且满足 0 ≤ a < 5。也就是说，x 与 a 在模 5 意义下是等价的，即它们属于同一个剩余类。

因此，模5意义下的剩余类可以表示为5 组 {0, 1, 2, 3, 4}，分别对应着

- x ≡ 0 (mod 5)，模 5 余数为 0：{... -10, -5, 0, 5, 10 ,15 ...}
- x ≡ 1 (mod 5)，模 5 余数为 1：{... -9, -4, 1, 6, 11 ,16 ...}
- x ≡ 2 (mod 5)，模 5 余数为 2：{... -8, -3, 2, 7, 12 ,17 ...}
- x ≡ 3 (mod 5)，模 5 余数为 3：{... -7, -2, 3, 8, 13 ,18 ...}
- 和 x ≡ 4 (mod 5) 这五个同余方程的解集。

比如模 5 下余数为 0 的剩余类，就记作 `[0] / [5]`

中国剩余定理

如果你有一些关于同余方程的问题，例如：

$$
\begin{aligned}
x &\equiv a1 (mod m1) \\
x &\equiv a2 (mod m2) \\
x &\equiv a3 (mod m3) \\
&\cdots \\
x &\equiv ak (mod mk)
\end{aligned}
$$

那么中国剩余定理告诉你，只要这些模数 m1, m2, m3, ..., mk 互质（也就是没有共同的因数），那么一定存在一个解 x，而且这个解是唯一的，且在模数 m1 × m2 × m3 × ... × mk 的意义下。

解决这个问题的方法是，首先用扩展欧几里得算法计算出每个模数 mi 对于所有其他模数的乘积的逆元 ti，然后计算出

```
x = (a1 * t1 * M/m1) + (a2 * t2 * M/m2) + ... + (ak * tk * M/mk) (mod M)
```

其中 M = m1 × m2 × m3 × ... × mk 是所有模数的乘积，ti 是 mi 对于 M/mi 的逆元，也就是满足 ti × mi ≡ 1 (mod M/mi) 的数。这个公式就是中国剩余定理的核心。

需要注意的是，如果这些模数不互质，那么这个公式可能没有解，或者有多个解。因此在使用中国剩余定理的时候，必须要先检查这些模数是否互质。

中国剩余定理（Chinese Remainder Theorem）是一个用于解决同余方程组的定理，其基本思想是将一个大的同余方程组分解为若干个小的同余方程组，然后分别求解再合并起来，从而得到整个方程组的解。

下面以一个简单的例子来介绍中国剩余定理：

假设我们要解决以下同余方程组：

$$
\begin{cases}
x \equiv 2 \pmod{3} \\
x \equiv 3 \pmod{5} \\
x \equiv 2 \pmod{7}
\end{cases}
$$

根据中国剩余定理，我们可以将它分解为三个小的同余方程组：

$x \equiv 2 \pmod{3}$
$x \equiv 3 \pmod{5}$
$x \equiv 2 \pmod{7}$

然后，我们可以分别求解这三个方程组。例如，

1. 对于第一个方程组，其解是 $\{x \in \mathbb{Z} \mid x = 3k + 2$，其中 $k \in \mathbb{Z}\}$ 如 $x = 2, 5, 8, 11, \ldots$ 找到一个符合条件的解，即 $x = 2$。
2. 同理，我们可以得到第二个方程组的解为 $x = 23$，
3. 第三个方程组的解为 $x = 16$。

最后，根据中国剩余定理，我们可以将这三个解合并起来，得到整个方程组的解：

$x \equiv 2 \cdot 35 \cdot 16 + 3 \cdot 21 \cdot 16 + 2 \cdot 15 \cdot 23 \pmod{3 \cdot 5 \cdot 7}$

化简得 $x \equiv 2338 \pmod{105}$，因此 $x = 2338 + 105k$，其中 k 是任意整数。这样，我们就求出了该同余方程组的所有解。

$$
\begin{cases}
1a + 1 \\
2
\end{cases}
$$

质因数分解

要分解一个数的质因数，可以按照以下步骤进行：

1. 从小到大找到该数的最小质因数，可以从2开始，一直试除到这个数的平方根，如果都不能整除，则该数为质数，否则就找到了最小质因数。
2. 将该数除以最小质因数得到一个商和一个余数，如果余数为 0 ，则商为下一次要分解的数，否则用下一个质因数试除。
3. 重复第1、2步，直到商为1为止。此时，所有的质因数都已经找到。

举个例子，假设要分解的数为 24 ：

1. 从 2 开始试除，发现 2 可以整除 24，因此 24 的最小质因数是2。
2. 将 24 除以 2 得到 12，继续试除2，发现 2 还能整除12，将12再次除以2得到6，继续试除2，得到3，此时 3 是质数，因此24的质因数分解为2 × 2 × 2 × 3。

需要注意的是，如果要分解的数是质数，那么其质因数分解只有一个因数，即它本身。

欧拉函数

欧拉函数，又称欧拉-φ 函数（Euler's totient function），是一种重要的数论函数。
对于任意正整数 n，欧拉函数 φ(n) 定义为小于或等于 n 的正整数中与 n 互质的数的个数，即：

$\varphi(n) = k | 1 \le k \le n, \quad gcd(k,n) = 1$ 的个数。

其中 $gcd(a,b)$ 表示 a 和 b 的最大公约数。

例如，当 n = 6 时，小于或等于 6 的正整数中，与 6 互质的数有 1、5，因此 $\varphi(6) = 2$。

欧拉函数的一些性质如下：

1. **如果 p 是质数，则** $\varphi(p) = p - 1$。
    1. 很显然 , 如 $\varphi(7) = 6$, 分别是 1, 2, 3, 4, 5 , 6 ;
    2. 如 $\varphi(11) = 10$, 分别是 1, 2, 3, 4, ... 8, 9, 10
2. **如果 p 和 q 是两个不同的质数，则** $\varphi(pq) = (p-1) \times (q-1)$。
    1. 如 2 *3 : $\varphi(pq) = \varphi(6) = (2-1)$ (3-1) = 2$*
3. 对于任意正整数 n，有 $\varphi(n) = n \times \Pi(1 - 1/p)$，其中 Π 的范围是质因数分解中的所有不同质因数，p 是质因子。
4. 如果 a 和 n 互质，则 $a^{\varphi(n)} \equiv 1 \pmod{n}$，其中 ≡ 表示同余。
    1. 比如 3 和 5 互质 :
    2. $3^{\varphi(n)} = 3^{\varphi(5)} = 3^{5-1} = 81 \pmod{5} == 1$
5. $\varphi(p^k) = p^{k-1}\varphi(p)$ 即 $\varphi(p^k) = p^{k-1}(p-1) \leftarrow \ If\ p\ is\ prime$
    1. 证明在下面, 很简单
6. 设两两互素的正整数 $n_1, n_2, \ldots, n_m \in N$ 则 :
7. $$\varphi(\prod_{i=1}^{m} n_i) = \prod_{i=1}^{m} \varphi(n_i)$$

    如 :

$$\begin{aligned} \varphi(2 \times 3 \times 5) &= \varphi(2) \times \varphi(3) \times \varphi(5) \\ \varphi(30) &= (2-1) \times (3-1) \times (5-1) \\ &= 8 \end{aligned}$$

性质 3 可能比较复杂 :

如果 $n = p_1^{k_1} p_2^{k_2} \cdots p_r^{k_r}$ 是 n 的质因数分解式(见 质因数分解)，那么

$$\varphi(n) = n \cdot \prod_{i=1}^{r} \left(1 - \frac{1}{p_i}\right)$$

这个公式告诉我们，欧拉函数可以用 n 的质因数分解式来计算。下面是一个例子：

假设 $n = 210 = 2 \cdot 3 \cdot 5 \cdot 7$，那么

$$\varphi(210) = 210 \cdot \left(1 - \frac{1}{2}\right) \cdot \left(1 - \frac{1}{3}\right) \cdot \left(1 - \frac{1}{5}\right) \cdot \left(1 - \frac{1}{7}\right) = 48$$

因为小于等于 210 的正整数中，与 210 互质的数共有 48 个。

性质 5 :

$$\varphi(p^k) = p^{k-1} \cdot \varphi(p) = p^{k-1} \cdot (p-1)$$

> 重复一遍: 欧拉函数 $\varphi(n)$ 是指小于或等于 n 的正整数中与 n 互质的数的个数。
> 如何理解上式: 质数 p 的 k 次方, 等于 $p^{k-1} \cdot (p-1)$?

看一下 $\varphi(p^k)$ 的定义 :

- **小于或等于 p^k 的正整数中，与 p^k 互质的数，必须不包含 p 这个质因数。**
- 因此，可以把问题转化为：小于或等于 p^k 的正整数中，有多少个数包含 p 这个质因数?

我们知道，小于或等于 p^k 的正整数中，包含 p 这个质因数的数的个数是 p^{k-1}。这是因为，这些数可以表示为 $p \times m$ 的形式 :

如 $\underbrace{p \times 0,\ p \times 1,\ p \times 2, \ldots, p \times (p^{k-1} - 1)}_{\text{共 } p^{k-1} \text{ 个}}$

其中 m 是小于或等于 p^{k-1} 的正整数，因此有 p^{k-1} 个选项。而这些数不与 p^k 互质，因此它们不能被计入 $\varphi(p^k)$。

因此，小于或等于 p^k 的正整数中，与 p^k 互质的数的个数就是 $p^k - p^{k-1}$。因为 p 是一个质数，所以小于 p 的正整数中，与 p 互质的数的个数是 $p-1$。因此，我们可以得到：

$$\begin{aligned}\varphi(p^k) &= p^k - p^{k-1} \\ &= p^k \cdot \left(1 - \frac{1}{p}\right) \\ &= p^{k-1} \cdot p \cdot \left(1 - \frac{1}{p}\right) \\ &= p^{k-1} \cdot (p-1)\end{aligned}$$

这就证明了 $\varphi(p^k) = p^{k-1}\varphi(p)$。

举个栗子：

$$\begin{aligned}2^5 &= 2 \times 2 \times 2 \times 2 \times 2 \\ \varphi(2^5) &= 2 \times 2 \times 2 \times 2 \times 2 - 2 \times 2 \times 2 \times 2 = 32 - 16 = 16 \\ &= p^{k-1} \cdot (p-1) \\ &= 2^{5-1} \cdot (2-1)\end{aligned}$$

试计算 $\varphi(30000)$：

$$\begin{aligned}\varphi(30000) &= \varphi(3 \times 2^4 \times 5^4) \\ &= \varphi(3) \cdot \varphi(2^4) \cdot \varphi(5^4) \quad // \because \text{两两互素} \quad \varphi(\prod_{i=1}^{m} n_i) = \prod_{i=1}^{m} \varphi(n_i) \\ &= \varphi(3) \cdot 2^3\varphi(2) \cdot 5^3\varphi(5) \quad // \because \varphi(p^k) = p^{k-1} \cdot (p-1) \\ &= (3-1) \cdot 8 \cdot (2-1) \cdot 5^3 \cdot (5-1) \\ &= 8000\end{aligned}$$

现代计算机与因子分解

对于现代计算机而言，当待分解的数变得非常大时，因子分解问题就变得极其困难，需要进行大量的计算才能找到其因子。

如 $n = pq$, 其中 p 和 q 是不同的大素数 , 如果这 2 个素数的长度达到 1024 bit, 就目前计算机的水平而言, 除非这 2 个素数有特殊的结构, 否接就基本不可能分解 n , 要计算 $\varphi(n)$ 也就不现实, 这个困难问题就被著名的 RSA 算法所应用

然而，对于量子计算机而言，比如应用 Shor 算法，由于其量子并行性的优势，因子分解问题的难度可以得到大幅降低，这使得一些现有的密码算法 (如 RSA) 在量子计算机的攻击下变得不安全。

不过量子计算机的研制还很远, 更重要的问题是快速找到因子分解的算法

乘法阶

对于一个正整数 a 和正整数 n，它们的乘法阶是指最小的正整数 k，使得 $a^k \equiv 1 \pmod{n}$。

换句话说，乘法阶是指将 a 不断自乘，直到得到 1（模 n 下），所需要自乘的次数。

如果这样的 k 不存在，则a在模n下没有乘法逆元，即a和n不互质。

乘法阶有以下性质：

1. 如果 a 和 n 互质，则 a 在模 n 意义下的乘法阶存在。
2. a 在模 n 意义下的乘法阶等于 a 在模 n 意义下的幂次的周期性。也就是说，如果 $a^k \equiv 1 \pmod{n}$，那么对于任意整数 m，都有 $a^{k+m\phi(n)} \equiv a^k \pmod{n}$。
3. 如果 a 在模 n 意义下的乘法阶为 k，则 $a^k \equiv 1 \pmod{n}$。反之，如果 $a^k \equiv 1 \pmod{n}$，则 a 在模 n 意义下的乘法阶必定是 k 的因数。
4. 如果 a 和 b 在模 n 意义下的乘法阶都存在，则 ab 在模 n 意义下的乘法阶也存在，并且有 $\operatorname{lcm}(k_a, k_b) \mid k_{ab}$，其中 lcm 表示最小公倍数。

欧拉定理 :

(欧拉定理) : 设 $a \in {Z_n}^*$, 则 :

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

且 $k \mid \varphi(n)$, 其中 k 是 a 在模 n 下的阶

> 利用欧拉定理的前提是 : a 和 n 互素

举例 : $2^{\varphi(5)} \equiv 1 \pmod{5}$, 即 2 的 (5-1) 次方 模 5 = 1 。

欧拉定理的证明很有技巧性 :

首先，我们要证明一个引理：对于任意正整数 a 和 b，如果 a 和 b 互质，则存在正整数 k 和 l ，使得$ka+lb=1$。

证明：我们可以用扩展欧几里得算法来证明这个引理。根据扩展欧几里得算法，我们可以找到两个数 x 和 y，满足 $\gcd(a,b)=ax+by$。因为 a 和 b 互质，所以 $\gcd(a,b)=1$。因此我们可以写出：

$$1=ax+by$$

因此，$k=x$，$l=y$，我们得证。

现在，我们考虑欧拉定理。对于任意正整数 a 和模数 n，如果 a 和 n 互质，那么我们可以找到一个整数k，使得$ak\equiv 1 \pmod n$，根据引理，我们可以找到一个正整数l，使得 $ak+nl=1$。因此，我们可以得到：

$$\begin{aligned} a\cdot ak &\equiv a \pmod n \\ \Rightarrow a^2k &\equiv a \pmod n \\ \Rightarrow a^2k\cdot ak &\equiv ak \pmod n \\ \Rightarrow a^3k^2 &\equiv ak \pmod n \end{aligned}$$

以此类推，我们可以得到：

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

因为 $\gcd(a,n)=1$ ，所以 a 的欧拉函数 $\varphi(n)$ 等于模数 n 的欧拉函数。因此，我们可以写出：

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

因此，欧拉定理得证。

> 注意：这里的证明使用了一些数论中的 模同余，欧拉函数，扩展欧几里得算法等

欧拉定理举例

欧拉定理给出了一个关于模运算的定理，如果 a 和 m 是互质的正整数，则有

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

其中 $\varphi(m)$ 是欧拉函数，表示小于等于 m 的正整数中与 m 互质的数的个数。

对于模 8，我们知道 $\varphi(8)=4$ $(\because \varphi(8)=\varphi(2^3)=2^2\varphi(2)=4)$ ， 因此，根据欧拉定理，对于任何一个与 8 互质的正整数 a，都有

$$a^4\equiv 1 \pmod 8.$$

现在考虑 $3^{2022} \pmod 8$，我们可以将 2022 写成 $2022=4\times 505+2$，然后使用欧拉定理得到：

$$\begin{aligned} 3^{2022} &\equiv 3^{4\times 505+2} \pmod 8 \\ &\equiv (3^4)^{505}\times 3^2 \pmod 8 && \leftarrow \text{对乘法各部先求模} \\ &\equiv 1^{505}\times 9 \pmod 8 \\ &\equiv 1\times 1 \pmod 8 \\ &\equiv 1 \pmod 8. \end{aligned}$$

因此，3^{2022} 除以 8 的余数为 1。

对于这个题来说：

1. 看到 `mod 8` , 先求一下 $\varphi(8)==4$,
2. 后面就只需要看 3^{2022} 中的 2022 能不能拆成 $4\times$??? 的形式即可

费马小定理

欧拉定理：

$$\begin{aligned} a^{\varphi(p)} &\equiv 1 \pmod p \\ a^{p-1} &\equiv 1 \pmod p && \text{同}\times a \\ a^p &\equiv a \pmod p && \leftarrow \text{费马小定理} \end{aligned}$$

对于欧拉定理，特别地，当 $p\in$ 素数 时，该结论加强为 [费马小定理]

Exercise

首先，我们可以将 20212017^2 表示为 $(20212022-5)^2$，即：

$$20212017^2=(20212022-5)^2=20212022^2-2\cdot 5\cdot 20212022+5^2$$

我们想要求出 20212017^2 除以 20212022 的余数，因此我们只需要求出 $2\cdot 5\cdot 20212022$ 和 5^2 除以 20212022 的余数，然后将这两个余数相加，再用 20212022 减去这个和，就是所求的余数。

$2 \cdot 5 \cdot 20212022/20212022 = 2 \cdot 5$ 。这个计算很简单

接下来，我们来计算 5^2 除以 20212022 的余数。我们可以使用模运算的另一个性质：如果 $a \equiv b \pmod{m}$，那么 $a^k \equiv b^k \pmod{m}$，其中 k 是任意正整数。因此：

$$5^2 \equiv 25 \pmod{20212022}$$

现在我们将这两个余数相加，并用 20212022 减去和，得到：

$$10 + 25 = 35 \equiv -20211987 \pmod{20212022}$$

因此，20212017^2 除以 20212022 的余数是 -20211987，或者等价地，$20212022 - 20211987 = \boxed{35}$。

The third-smallest positive value of x for which $2^x \equiv 3 \ (mod \ 13)$ is therefore $x = 5 + 6 \cdot 2 = \boxed{17}$.