第十一章 欧拉函数与中国剩余定理

> φ**函数公式**：
>
> 1. 如果p是素数且$k \geq 1$，则$\varphi(p^k) = p^k - p^{k-1}$
> 2. 如果$gcd(m,n) = 1$，则$\varphi(mn) = \varphi(m)\varphi(n)$

> **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
>
> $$x \equiv b \pmod{m} \text{ 与 } x \equiv c \pmod{n}$$
>
> 恰有一个解$0 \leq x < mn$

> **习题**

- *11.1*
  - *(a)* 求$\varphi(97)$的值。
  - *(b)* 求$\varphi(8800)$的值。

  (a) 因为97是素数且$k=1$，所以$\varphi(97) = 97-1 = 96$

  (b) 因为$8800 = 2^5 \times 5^2 \times 11$，所以，根据$\varphi$函数公式$\varphi(8800) = \varphi(2^5) \cdot \varphi(5^2) \cdot \varphi(11)$

  $\varphi(2^5) = 2^5 - 2^4 = 16$，$\varphi(5^2) = 5^2 - 5 = 20$，$\varphi(11) = 10$

  所以$\varphi(8800) = 3200$

- *11.2*
  - *(a)* 如果$m \geq 3$，解释为什么$\varphi(m)$总是偶数。
  - *(b)* $\varphi(m)$"经常"被4整除。叙述$\varphi(m)$不能被4整除的所有m。

  (a) 首先因数分解$m = p_1^{k_1} \cdot p_2^{k_2} \cdot \cdots \cdot p_r^{k_r} = (p_1^{k_1} - p_1^{k_1-1})(p_2^{k_2} - p_2^{k_2-1}) \cdots (p_r^{k_r} - p_r^{k_r-1})$

  无论p为何值，$p^k - p^{k-1}$均为偶数，又因为偶数的乘积为偶数，所以$\varphi(m)$总是偶数

  (b) 根据(a)当m能分解为两个不同素数乘积时，$\varphi(m)$一定被4整除

  所以当$m=1$，$m=2$，$m=4$，$m=p^k$时，$\varphi(m)$不能被4整除

- *11.3* 假设p_1，p_2，$\cdots$，p_r是整除m的不同素数。证明$\varphi(m)$的下述公式成立：

$$\varphi(m) = m(1 - \frac{1}{p_1})(1 - \frac{1}{p_2}) \cdots (1 - \frac{1}{p_r})$$

使用这个公式计算$\varphi(1000000)$。

11.3 首先因数分解$m = p_1^{k_1} \cdot p_2^{k_2} \cdot \cdots \cdot p_r^{k_r}$

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

$= m(1 - \frac{1}{p_1})(1 - \frac{1}{p_2}) \cdots (1 - \frac{1}{p_r})$，得证

$1000000 = 10^6 = (2 \times 5)^6 = 2^6 \times 5^6$

所以$\varphi(1000000) = 1000000 \times (1 - \frac{1}{2}) \times (1 - \frac{1}{5}) = 400000$

- *11.4* 编写程序计算欧拉函数$\varphi(n)$的值。应该使用n的素因数分解来计算$\varphi(n)$，而不是通过求与n互素1与n之间的所有a来计算$\varphi(n)$。

```go
package main

import (
	"fmt"
)

func primeFactors(n int) map[int]bool {
	factors := make(map[int]bool)

	for n%2 == 0 {
		factors[2] = true
		n = n / 2
	}

	for i := 3; i*i <= n; i += 2 {
		for n%i == 0 {
			factors[i] = true
			n = n / i
		}
	}

	if n > 2 {
		factors[n] = true
	}
	return factors
}

func eulerPhi(n int) int {
	if n < 1 {
		return 0
	}
	if n == 1 {
		return 1
	}

	factors := primeFactors(n)
	result := n
	//euler函数公式
	for p := range factors {
		result = result * (p - 1) / p
	}
	return result
}

func main() {
	var n int
```

```go
    fmt.Print("请输入整数: ")
    _, err := fmt.Scan(&n)
    if err != nil {
        return
    }

    fmt.Printf("φ(%d) = %d\n", n, eulerPhi(n))
}
```

- *11.5* 对每个同余式组求其解x。
  - *(a)* $x \equiv 3 \pmod{7}$，$x \equiv 5 \pmod{9}$
  - *(b)* $x \equiv 3 \pmod{37}$，$x \equiv 1 \pmod{87}$
  - *(c)* $x \equiv 5 \pmod{7}$，$x \equiv 2 \pmod{12}$，$x \equiv 8 \pmod{13}$

(a) 根据 $x \equiv 3 \pmod 7$，首先设 $x = 7k+3$，将 $x=7k+3$ 代入 $x \equiv 5 \pmod 9$

$7k \equiv 2 \pmod 9$ 得 $7k - 9l = 2$.

利用扩展欧几里得原理 $9 = 1\times 7 + 2$　　$2 = a - b \ (a=9, b=7)$

$7 = 3\times 2 + 1 \Rightarrow 1 = b - 3(a-b) = 4b - 3a$

$2 = 2\times 1 + 0$　　即 $7\times 4 - 9\times 3 = 1$

得 $k \equiv 8 \pmod 9$，代入 $x = 7k+3$ 得 $x = 59$

(b) 根据 $x \equiv 3 \pmod{37}$，首先设 $x = 37k+3$，将 $x = 37k+3$ 代入 $x \equiv 1 \pmod{87}$

$37k \equiv -2 \pmod{87}$ 得 $37k - 87l = -2$

利用扩展欧几里得原理 $87 = 2\times 37 + 13$　　$13 = a - 2b \ (a=87, b=37)$

$37 = 2\times 13 + 11$　　$11 = b - 2(a-2b) = 5b - 2a$

$13 = 1\times 11 + 2 \Rightarrow 2 = a - 2b - (5b - 2a) = 3a - 7b$

$11 = 5\times 2 + 1$　　$1 = 5b - 2a - 5(3a - 7b) = 40b - 17a$

$2 = 2\times 1 + 0$　　即 $37\times 40 - 87\times 17 = 1$

得 $k \equiv 7 \pmod{87}$，代入 $x = 37k+3$ 得 $x = 262$

(c) 根据 $x \equiv 5 \pmod 7$，首先设 $x = 7k+5$，将 $x = 7k+5$ 代入 $x \equiv 2 \pmod{12}$ 和

$x \equiv 8 \pmod{13}$ 得 $7k \equiv -3 \pmod{12}$ 和 $7k \equiv 3 \pmod{13}$

首先求解第一个式子，利用扩展欧几里得原理，解得 $k = 3 + 12m$

将k代入第二个式子得 $7(3+12m) \equiv 3 \pmod{13}$ 转化为解 $84m - 13n = -18$

利用扩展欧几里得原理，$84 = 6\times 13 + 6$　　$6 = a - 6b \ (a=84, b=13)$

$13 = 2\times 6 + 1 \Rightarrow 1 = b - 2(a - 6b) = 13b - 2a$

$6 = 6\times 1 + 0$　　即 $84\times(-2) + 13\times 13 = 1$

得 $m \equiv 10 \pmod{13}$，代入 $k = 3 + 12m = 123$，$x = 7k + 5 = 866$

- *11.6* 解“历史插曲”提到的《孙子算经》中已有1700年历史的中国剩余定理。

11.6 孙子算经中提到的中国剩余定理为解 $x\equiv 2 \pmod 3$，$x\equiv 3 \pmod 5$，$x\equiv 2 \pmod 7$

根据 $x\equiv 2 \pmod 3$，首先设 $x=3k+2$，将 $x=3k+2$ 代入 $x\equiv 3 \pmod 5$ 和 $x\equiv 2 \pmod 7$

得 $3k\equiv 1 \pmod 5$ 和 $3k\equiv 0 \pmod 7$

首先求解第一个式子，利用扩展欧几里得原理，解得 $k=2+5m$

将 k 代入第二个式子得 $3(2+5m)\equiv 0 \pmod 7$ 转化为解 $15m-7n=-6$

利用扩展欧几里得原理 $15=2\times 7+1$，$7=1\times 1+0$ $\Rightarrow$ $1=a-2b$ $(a=15, b=7)$ 即 $15-7\times 2=1$

得 $m\equiv 1 \pmod 7$，代入 $k=2+5m=7$，$x=3k+2=23$

- *11.7* 一个农夫在去集市卖鸡蛋的路上流星打中了他的小货车，击碎了他的鸡蛋。为申请保险索赔，他需要知道打碎了多少鸡蛋。他知道两两数之余一，三三数之余一，四四数之余一，五五数之余一，六六数之余一，七七数之余零。问小货车里鸡蛋的最少个数是多少?

11.7 题目中要求解 $x\equiv 1 \pmod 2$，$x\equiv 1 \pmod 3$，$x\equiv 1 \pmod 4$，$x\equiv 1 \pmod 5$

$x\equiv 1 \pmod 6$，$x\equiv 0 \pmod 7$

根据中国剩余定理可以转化为解 $x\equiv 1 \pmod{12}$，$x\equiv 1 \pmod 5$，$x\equiv 0 \pmod 7$

根据 $x\equiv 1 \pmod{12}$，首先设 $x=12k+1$，将 $x=12k+1$ 代入 $x\equiv 1 \pmod 5$ 和 $x\equiv 0 \pmod 7$

得 $12k\equiv 0 \pmod 5$ 和 $12k\equiv -1 \pmod 7$

首先求解第一个式子，利用扩展欧几里得原理，解得 $k=5+12m$

将 k 代入第二个式子得 $12(5+12m)\equiv -1 \pmod 7$ 转化为解 $144m-7n=-61$

利用扩展欧几里得原理 $144=20\times 7+4$，$7=1\times 4+3$，$4=1\times 3+1$，$3=3\times 1+0$ $\Rightarrow$ $4=a-20b$ $(a=144, b=7)$，$3=b-(a-20b)=21b-a$，$1=2a-41b$ 即 $144\times 2-7\times 41=1$

得 $m=4 \pmod 7$，代入 $k=5+12m=25$，$x=12k+1=301$

- *11.8* 编写程序，取四个整数$(b, m, c, n)(gcd(m, n)=1)$作为输入，计算满足

$$x\equiv b \pmod m，x\equiv c \pmod n，0\le x< mn$$

的整数x。

```
package main

import (
    "fmt"
)

func extendedGCD(a, b int) (int, int, int) {
    if b == 0 {
        return a, 1, 0
    }
    g, x1, y1 := extendedGCD(b, a%b)
```

```go
    x := y1
    y := x1 - (a/b)*y1
    return g, x, y
}

func chineseRemainderTheorem(b, m, c, n int) int {
    g, x1, x2 := extendedGCD(m, n)
    if g != 1 {
        fmt.Println("需要m和n互质。")
        return -1
    }

    x := (b*n*x2 + c*m*x1) % (m * n)
    if x < 0 {
        x += m * n
    }

    return x
}

func main() {
    var b, m, c, n int
    fmt.Print("请输入整数b, m, c, n: ")
    _, err := fmt.Scan(&b, &m, &c, &n)
    if err != nil {
        return
    }

    x := chineseRemainderTheorem(b, m, c, n)
    if x != -1 {
        fmt.Printf("满足条件的解为: x = %d\n", x)
    }
}
```

- *11.9* 在本题中将证明三个同余式的中国剩余定理。设m_1，m_2，m_3是两两互素的正整数，即

  $$gcd(m_1, m_2) = 1，gcd(m_1, m_3) = 1，gcd(m_2, m_3) = 1$$

  设a_1，a_2，a_3是任意三个整数。证明同余式组

  $$x \equiv a_1 \pmod{m_1}，x \equiv a_2 \pmod{m_2}，x \equiv a_3 \pmod{m_3}$$

  在区间$0 \leq x < m_1 m_2 m_3$恰有一个整数解x。你能找出将这个问题推广到处理多个同余式

  $$x \equiv a_1 \pmod{m_1}，x \equiv a_2 \pmod{m_2}，\cdots，x \equiv a_r \pmod{m_r}$$

的模式吗？特别地，模m_1，m_2，$\cdots$，m_r需要满足什么条件呢？

11.9 由于m_1, m_2, m_3互素，我们可以应用中国剩余定理的构造方法。设$M=m_1m_2m_3$，由于m_1, m_2, m_3互素，我们可以分别构造每个同余式的解，首先，设$M_1=\frac{M}{m_1}$，$M_2=\frac{M}{m_2}$，$M_3=\frac{M}{m_3}$

根据中国剩余定理，存在整数y_1, y_2, y_3使得$M_1y_1\equiv 1 \pmod{m_1}$，$M_2y_2\equiv 1 \pmod{m_2}$，$M_3y_3\equiv 1 \pmod{m_3}$

构造解为 $x=a_1M_1y_1+a_2M_2y_2+a_3M_3y_3 \in [0, m_1m_2m_3)$

得证

对于多个同余式的情况，同余式组在区间$0\le x<m_1m_2m_3\cdots m_r$恰有一个整数解$x$，模$m_1$，$m_2$，$\cdots$，$m_r$需要两两互素。

- *11.10* 如果$\varphi(n)$是素数，你能说出n有什么模式吗？如果$\varphi(n)$是素数的平方，n又有什么模式呢？

11.10 从欧拉函数的公式可以得到$\varphi(n)$总是偶数，因此，如果$\varphi(n)$是素数，$\varphi(n)$只能等于2，$n=3,4,6$

如果$\varphi(n)$是素数的平方 $\varphi(n)$只能等于4，此时，$n=5,8,10,12$

- *11.11*
  - *(a)* 求$\varphi(n)=160$的至少5个不同整数。
  - *(b)* 假设整数n满足$\varphi(n)=1000$。列出可能整除n的所有素数。
  - *(c)* 由*(b)*求出所有满足$\varphi(n)=1000$的整数。

(a) $160=2^5\times 5=10\times 16=4\times 40$

当$160=4\times 40$时，此时，n可以等于5×41，8×41，10×41，12×41，即205，328，410，492

当$160=10\times 16$时，$n=11\times 17=187$

(b) 可能整除n的所有素数为2，3，5，11，41，101，251

(c) 所有满足$\varphi(n)=1000$的整数为1111，1255，2008，2510，3012，2222，2510，4016，5020，6024，2750，4518。

- *11.12* 解下述方程，求n的所有值。
  - *(a)* $\varphi(n)=\frac{n}{2}$
  - *(b)* $\varphi(n)=\frac{n}{3}$
  - *(c)* $\varphi(n)=\frac{n}{6}$

(a) 根据11.3的$\varphi(n)$公式，$\varphi(n)/n=\frac{1}{2}$ 即$(1-\frac{1}{p_1})\cdots(1-\frac{1}{p_r})=\frac{1}{2}$，此时，只有一个素因子2，

所以$n=2^k$

(b) 同(a)，$\varphi(n)/n=\frac{1}{3}$，即$(1-\frac{1}{p_1})\cdots(1-\frac{1}{p_r})=\frac{1}{3}$，此时，有2个素因子2和3，所以$n=2^k\cdot 3^l$

(c) 同(a)，$\varphi(n)/n=\frac{1}{6}$，即$(1-\frac{1}{p_1})\cdots(1-\frac{1}{p_r})=\frac{1}{6}$，此时，无解

- *11.13*

- *(a)* 对每个整数$2 \leq a \leq 10$，求a^{1000}的最末4位数。

```
[tangxianning@MacBook-Air code % go run test.go
2^1000 最末4位数是： 9376
3^1000 最末4位数是： 1
4^1000 最末4位数是： 9376
5^1000 最末4位数是： 625
6^1000 最末4位数是： 9376
7^1000 最末4位数是： 1
8^1000 最末4位数是： 9376
9^1000 最末4位数是： 1
10^1000 最末4位数是： 0
```

- *(b)* 基于*(a)*的试验，给出一种简单判别法，使得可由a的值预测a^{1000}的最末4位数。

(b) 当a以1,3,7,9结尾时，a^{1000} (mod 10000)为1

当a以2,4,6,8结尾时，a^{1000} (mod 10000)为9376

当a以5结尾时，a^{1000} (mod 10000)为625

当a以0结尾时，a^{1000} (mod 10000)为0