1.Security of communications

When communicate on a insecure network,we can encrypt messages to preserve secrets(**cryptography 密码学**)

To protect sensitive informamtion,we must ensure the following goals are met:

- Data Integrity 数据完整性 Information should not be altered without detection.
- Authentication 身份验证 Agentsinvolved in the communication must be identified.
- Authorization 授权 Agents operating onsensitive information must be authorised to perform those operations.
- Nonrepudiation 不可抵赖性 If binded by contracts, agents cannot drop out of their obligations without being detected.
- Confidentiality 机密性 Sensitive information must be kept secret from agents that are not authorised to access it.

为了保护敏感信息，产生了一些基于算法和沟通协议的技术，一些技术需要用到**数论(number theory)**，例如RSA scheme，除此之外一些安全协议还要用到形式化方法。

2.Review of math basics

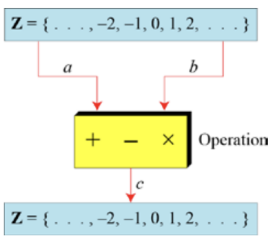
2.1Integer arthimetric

Z表示整数集合，表示从正无穷到负无穷的不包含分数的所有整数。

Z={....−2,−1,0,1,2....}

在密码学中，通常采用三种二元运算方法，所以有两个输入值一个输出值：

Figure *Three binary operations for the set ofintegers*



2.1.1Set of integers

The following shows the results of the three binary operations on two integers. Because each input can be either positive or negative, we can have four cases for each operation.

Add:	5 + 9 = 14	(–5) + 9 = 4	5 + (–9) = –4	(–5) + (–9) = –14
Subtract:	5 – 9 = –4	(–5) – 9 = –14	5 – (–9) = 14	(–5) – (–9) = +4
Multiply:	5 × 9 = 45	(–5) × 9 = –45	5 × (–9) = –45	(–5) × (–9) = 45

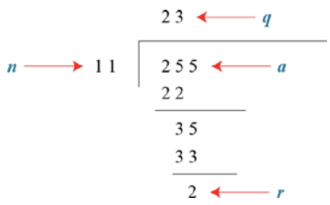
2.1.2Binary operations

为什么没有➗：有些情况下除法得到的不是整数。

2.1.3Integer division

定义：a=q*n+r 当用n去除a时，得到q以及r，如图：

Example *finding the quotient and the remainder*

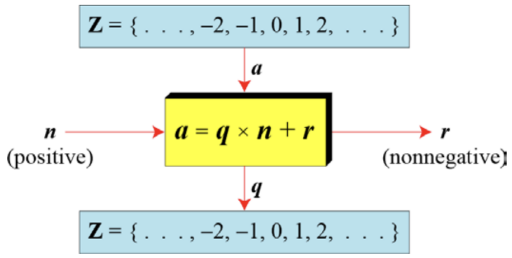


$0 \le r < |n|$

Assume that a = 255 and n = 11. We can find q = 23 and r = 2 using the division algorithm.

所以可以得到整数的除法算法：

Figure *Division algorithm for integers*



当a是负数时，r和q也是负数，当根据上述中的限制条件，r需要是正数，所以在此处应用：

将q减去1，r的位置则会变为n+r，例如：

–255=(–23*11)+(–2) 此时显而易见有a=–255，q=–23，n=11，r=–2。将–23减去1，则会得到：–255=(–24*11)+(–2)+11，此时r变为9，满足r是正数的要求。

2.1.4Divisibility

如果在上述问题中，a不为0，那么此时令r=0，则会得到：

a=q*n，如果余数(reminder)为0，记作 n | a 即a能被n整除，若不为0，记作的符号为：

$$n \nmid a$$

Example

a. The integer 4 divides the integer 32 because 32 = 8 ×4. We show this as

$$4 | 32$$

b. The number 8 does not divide the number 42 because 42 = 5 × 8 + 2. There is a remainder, the number 2, in the equation. We show this as

$$8 \nmid 42$$

a. We have $13|78$, $7|98$, $-6|24$, $4|44$, and $11|(-33)$.

b. We have $13 \nmid 27$, $7 \nmid 50$, $-6 \nmid 23$, $4 \nmid 41$, and $11 \nmid (-32)$.

Properties：

1. 如果a | 1，那么a的值是正负1。
2. 如果a | b且b | a，那么有a等于正负b。
3. 如果b | a且c | b，那么则有c | a。

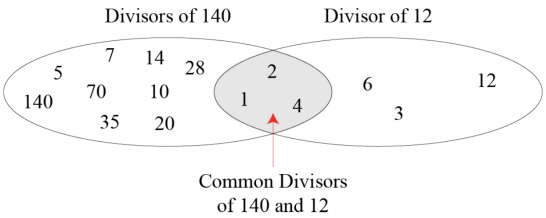
4. 如果a | b且a | c，那么a | (m*b+n*c)，且m和n是任意整数。

Example

a. Since 3|15 and 15|45,
according to the third property, 3|45.

b. Since 3|15 and 3|9,
according to the fourth property,
3|(15 × 2 + 9 × 4), which means 3|66.

Common divisors of twointegers



定义：**最大共同除数 Greatest common divisor**

两个正整数的最大共同除数就是能够整除这两个正整数的最大的除数。

定义：**欧几里得算法 Euclidean algorithm**

首先有：gcd(a,0)=a

引理：存在四个整数a,b,q,r使得a=bq+r且b不等于0，那么gcd(a,b)=gcd(b,r)

证明：

令d=gcd(a,b) e=gcd(b,r) 则需要证明d=e

又根据上述可知，d | a，d | b，

根据**Property4（如果a | b且a | c，那么a | (m*b+n*c)，且m和n是任意整数。）**

在上述式子中可得出d | a–bq（取参数为b和–q），又已知a–bq是r，

则可以得到d | r，所以d是b和r共有的一个divisor，而又根据假设e是b和r共有的divisor中最大的一个，

所以易得：d<=e ，后续证明过程同理：

e | b，e | r可以得出e | bq+r，即e | a，所以e是a和b共有的一个divisor，而a和b共有的divisor中最大的是d，所以易得：e<=d，

综上所述，d=e，即gcd(a,b)=gcd(b,r)，证毕。

定义：**互质（relatively prime）**

若g(c,d)=1，则a和b是互质的。

例子：

1. Given $26 = 6 \times 4 + 2$,find the gcd (26,6) and gcd(6,2).

Solution

a=26 , b=6, r=2

common divisors of 26 and 6: (1,2)

common divisors of 6 and 2: (1,2)

gcd(a,b)=2, gcd(b,r)=2

2. Given $60 = 24 \times 2 + 12$ Find the gcd (60,24) and gcd (24,12).

Solution

a=60 , b=24, r=12

common divisors of 60 and 24: (1,2,3,4,6,12)

common divisors of 24 and 12: (1,2,3,4,6,12)

gcd(a,b)=12, gcd(b,r)=12

定义：**基本欧几里得算法 Basic Euclidean algorithm**（即辗转相除法）

Basic Euclidean algorithms

```
def gcd(a,b)
    assert a>=b and b>=0 and a+b>0
    return gcd(b, a%b) if b>0 else a
```

modulo

欧几里得算法用来计算两个正整数的最大公约数，计算公式为gcd(a,b) = gcd(b,a mod b)，一直重复上述操作，例子如下：

Find the greatest common divisor of 25 and 60.

Solution

$r1 = 60, r2 = 25, \quad q = 2, \quad r = 10$
$r1 = 25, r2 = 10, \quad q = 2, \quad r = 5$
$r1 = 10, r2 = 5, \quad q = 2, \quad r = 0$
$r1 = 5, r2 = 0$

We have gcd (25, 60) = 5.

首先题目要求是求出gcd(60,25)，根据欧几里得算法，60mod25=10，所以问题变为gcd(25,10)，下一步中等式转换为gcd(10,5)➡️gcd(5,0)=5，因此60和25的最大公约数是5。

定义：**扩展欧几里得算法 Extended Euclidean algorithm**

Given two integers a and b, we often need to find other two integers, s and t, such that

$$s \times a + t \times b = \gcd(a, b)$$

The extended Euclidean algorithm can calculate the gcd (a, b) and at the same time calculate the value of s and t.

即计算出上述式子的整数解，具体方法如下：

注意：q=r1/r2这一步q的取值必须向下取整

Extended Euclidean Algorithm

$$s \times a + t \times b = \gcd(a, b)$$

$r_1 \leftarrow a;$ $r_2 \leftarrow b;$
$s_1 \leftarrow 1;$ $s_2 \leftarrow 0;$
$t_1 \leftarrow 0;$ $t_2 \leftarrow 1;$ (Initialization)

while $(r_2 > 0)$
{
$q \leftarrow r_1 / r_2;$

$r \leftarrow r_1 - q \times r_2;$
$r_1 \leftarrow r_2;$ $r_2 \leftarrow r;$ (Updating r's)

$s \leftarrow s_1 - q \times s_2;$
$s_1 \leftarrow s_2;$ $s_2 \leftarrow s;$ (Updating s's)

$t \leftarrow t_1 - q \times t_2;$
$t_1 \leftarrow t_2;$ $t_2 \leftarrow t;$ (Updating t's)

}
$\gcd(a, b) \leftarrow r_1;$ $s \leftarrow s_1;$ $t \leftarrow t_1$

首先将上述式子取出：sa+tb=gcd(a,b) 本质是求该二元一次不定方程的整数解，例子如下：

1.Given a = 161 and b = 28, find gcd (a, b) and the values of s and t.

Solution

We get gcd (161, 28) = 7, s = −1 and t = 6.

2. Given a = 391 and b = 299 , find gcd (a, b) and the values of s and t.

Solution

We get gcd (391, 299) = 23, s = -3, and t = 4.

Solution:

Example 1:

首先要找到gcd(161,28)，然后求出161s+28t=gcd(161,28) 的整数解。

根据辗转相除法：gcd(161,28) ➡️gcd(28,21) ➡️gcd(21,7)➡️gcd(7,0)=7

根据扩展欧几里得算法：计算161s+28t=gcd(161,28)，求出s和t，根据上述提到的算法：

s*r1+t*r2=gcd(r1,r2) s1=1，s2=0，t1=0，t2=1 此时r2=28>0，

q=r1/r2=5，r=r1−q*r2=161−140=21， r1=28， r2=21（更新r）

s=s1−q*s2=1−0=1，s1=0，s2=1 （更新s）

t=t1−q*t2=0−5=−5，t1=1，t2=−5 （更新t）

继续上述操作：

q=r1/r2=1 r=r1−q*r2=28−21=7，r1=21，r2=7

s=s1−q*s2=0−1=−1 s1=1，s2=−1

t=t1−q*t2=6 t1=−5，t2=6

此时r2=7仍大于0，继续上述操作：

q=r1/r2=3 r=r1−q*r2=21−21=0，r1=7，r2=0

s=s1−q*s2=4 s1=−1，s2=−4

t=t1−q*t2=−23 t1=6，t2=−23

此时条件不满足，跳到下一步操作：

则结果为gcd(a,b)=7，s=s1=−1，t=t1=6

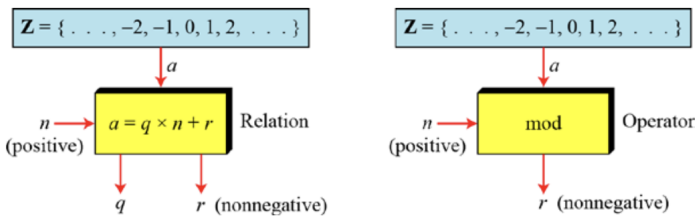
2.2Modular arithmetic

本部分只考虑在除法操作中一个输出值：余数。

2.2.1Modulo operator

The modulo operator is shown as mod (or % in some languages). The second input (n) is called the modulus. The output r is called the residue.

Division algorithm and modulo operator

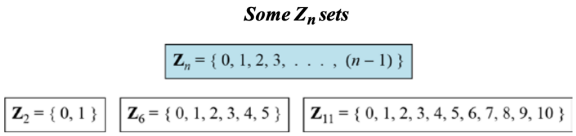


记作mod，输出的值即为余数，n被称作**模量(modulus)**，输出的r被称作**余数(residue)**。

2.2.2Set of residues

The modulo operation creates a set, which in modular arithmetic is referred to as the set of least residues modulo n, or Z_n.

Some Z_n sets



即模量n的最小余数集为从0到n−1。

2.2.3Congruence

证明两个整数是全等的(congruent)：

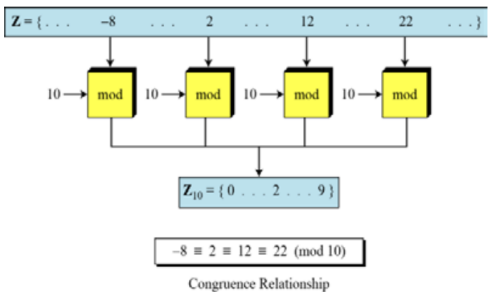
To show that two integers are congruent, we use the congruence operator (≡): a≡b(mod m)

A is congruent to B modulo m

For example, wewrite:

$2 \equiv 12 \pmod{10}$　　$13 \equiv 23 \pmod{10}$

$3 \equiv 8 \pmod{5}$　　$8 \equiv 13 \pmod{5}$



这里全等的概念指：对两个整数取模操作，得到的值是一致的。

定义：剩余类(Residue classes)

A residue class [a] or $[a]_n$ is the set of integers congruent modulo n:

$\{\ldots, a-3n, a-2n, a-n, a, a+n, a+2n, a+3n, \ldots\}$.

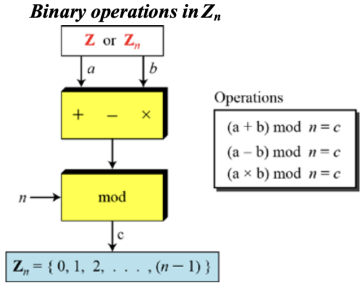
$$
\begin{aligned}
[0] &= \{\ldots, -15, -10, -5, 0, 5, 10, 15, \ldots\} \\
[1] &= \{\ldots, -14, -9, -4, 1, 6, 11, 16, \ldots\} \\
[2] &= \{\ldots, -13, -8, -3, 2, 7, 12, 17, \ldots\} \\
[3] &= \{\ldots, -12, -7, -5, 3, 8, 13, 18, \ldots\} \\
[4] &= \{\ldots, -11, -6, -1, 4, 9, 14, 19, \ldots\}
\end{aligned}
$$

即一个剩余类的数对取模得到的值相同。

2.2.4Operation in Zn

The three binary operations that we discussed for the set Z can also be defined for the set Z_n. The result may need to be mapped to Z_n using the mod operator

Binary operations in Z_n



Properties of mod operator

First Property:	$(a+b) \bmod n = [(a \bmod n) + (b \bmod n)] \bmod n$
Second Property:	$(a-b) \bmod n = [(a \bmod n) - (b \bmod n)] \bmod n$
Third Property:	$(a \times b) \bmod n = [(a \bmod n) \times (b \bmod n)] \bmod n$

Example

The following shows the application of the above properties:

(37 + 99) mod 6 = 136 mod 6 = 4
[(37 mod 6) + (99 mod 6)] mod 6 = (1+3) mod 6 = 4

另外，有时需要找出10的幂除以整数的余数：

$10^n \bmod x = (10 \bmod x)^n$ Applying the third property n times.

$$
\begin{aligned}
10 \bmod 3 = 1 &\rightarrow 10^n \bmod 3 = (10 \bmod 3)^n = 1 \\
10 \bmod 9 = 1 &\rightarrow 10^n \bmod 9 = (10 \bmod 9)^n = 1 \\
10 \bmod 7 = 3 &\rightarrow 10^n \bmod 7 = (10 \bmod 7)^n = 3^n \bmod 7
\end{aligned}
$$

此外，一个整数除以3的余数，与他十进制的数相加后得到的余数一样。

例子：将一个整数写成不同的十进制的位数乘10的不同幂数。

$$a = a_n \times 10^n + \cdots + a_1 \times 10^1 + a_0 \times 10^0$$
$$\text{For example: } 6371 = 6 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 1 \times 10^0$$

证明一个正整数除以3的余数与它各位相加得到的数除以3得到的余数相同。

证明：设这个正整数为a

a mod 3=(an*10^n+...+a1*10^1+a0*10^0)mod 3

根据First property：

上式可转化为：

(an*10^n mod 3)+...+(a1*10^1 mod 3)+(a0*10^0 mod 3)]mod 3

根据Third property：

上式可转化为：

{[(an mod 3)*(10^n mod 3)]mod3+...+[(a1 mod 3)*(10^1 mod 3)]+[(a0 mod 3)*(10^0 mod 3)]}mod 3

={[(an mod 3)mod 3]+...+[(a1 mod 3)mod 3]+[(a0 mod 3)mod 3]}mod 3

=(an mod 3+...a1 mod 3+a0 mod 3)mod 3

=[(an+...+a1+a0)mod 3]mod 3

=(an+...+a1+a0)mod 3 证毕

2.2.5Inverse

定义：Additive inverse 加法逆元

In Z_n, two numbers a and b are additive inverses of each other if

$$a + b \equiv 0 \pmod{n}$$

Note

In modular arithmetic, each integer has an additive inverse. The sum of an integer and its additive inverse is congruent to 0 modulon.

定义：Multiplicative inverse 乘法逆元

In Z_n, two numbers a and b are the multiplicative inverse of each other if

$$a \times b \equiv 1 \pmod{n}$$

Note

In modular arithmetic, an integer may or may not have a multiplicative inverse. When it does, the product of the integer and its multiplicative inverse is congruent to 1 modulo n.

例子如下：

Example 1

Find the multiplicative inverse of 8 in Z_{10}.

• Solution

There is no multiplicative inverse because gcd (10, 8) = 2 ≠ 1.

Example 2

Find all multiplicative inverses in Z_{10}

• Solution

(1, 1), (3, 7) and (9, 9).

在上述第一个例子中，判断一个数是否存在乘法逆元用扩展欧几里得算法：b在Zn中且满足gcd(b,n)=1，

此时该乘法逆元的值就是上述算法的t。且b的乘法逆元为**b^–1(mod n)**，具体算法如下：

Using extended Euclidean algorithm to find multiplicativeinverse

$r_1 \leftarrow n; \quad r_2 \leftarrow b;$
$t_1 \leftarrow 0; \quad t_2 \leftarrow 1;$
while $(r_2 > 0)$
{
$q \leftarrow r_1 / r_2;$
$r \leftarrow r_1 - q \times r_2;$
$r_1 \leftarrow r_2; \quad r_2 \leftarrow r;$
$t \leftarrow t_1 - q \times t_2;$
$t_1 \leftarrow t_2; \quad t_2 \leftarrow t;$
}
if $(r_1 = 1)$ then $b^{-1} \leftarrow t_1$

注：不要忘记给q进行**向下取整**操作。

例子：

找到Z26中11的乘法逆元：

n	b	q	r	t1	t2
26	11	2	4	0	1
11	4	2	3	1	−2
4	3	1	1	−2	5
3	1	3	0	5	−7

首先填写n和b的值，注意在上图的算法中没有提到n和b的更新，但其实就是上一部分中辗转相除法的过程，因此可以将n，b对应的值。

然后将最后一行的−7 去mod 26得到的值是19，故答案是19。

2.2.6Different sets

Some Z_n and Z_{n^*} sets

$$\mathbb{Z}_n^* = \{[a]_n \in \mathbb{Z}_n : \gcd(a, n) = 1\}$$

$\mathbf{Z}_6 = \{0, 1, 2, 3, 4, 5\}$ $\quad$ $\mathbf{Z}_6^* = \{1, 5\}$

$\mathbf{Z}_7 = \{0, 1, 2, 3, 4, 5, 6\}$ $\quad$ $\mathbf{Z}_7^* = \{1, 2, 3, 4, 5, 6\}$

$\mathbf{Z}_{10} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ $\quad$ $\mathbf{Z}_{10}^* = \{1, 3, 7, 9\}$

Zn用来计算加法逆元，Zn*用来计算乘法逆元。密码学同样用到另外两个集合：Zp和Zp*，和前两个不同到地方在于这里到模量是质数(primes)。例子：

$$Z_{13} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$$
$$Z_{13}* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$$

2.3Matrices

线性代数中的矩阵在密码学中同样很重要。

2.3.1Definition

A matrix of size $l \times m$

m columns

Matrix **A**: l rows

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{lm} \end{bmatrix}$$

Examples of matrices

$\begin{bmatrix} 2 & 1 & 5 & 11 \end{bmatrix}$ Row matrix

$\begin{bmatrix} 2 \\ 4 \\ 12 \end{bmatrix}$ Column matrix

$\begin{bmatrix} 23 & 14 & 56 \\ 12 & 21 & 18 \\ 10 & 8 & 31 \end{bmatrix}$ Square matrix

$\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$ **0**

$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ **I** Identity matrix

行矩阵，列矩阵，方阵，0矩阵，单位矩阵（方阵，主对角线/从左上到右下到对角线元素均为1）

2.3.2Operations and relations

加法，减法，数量乘，乘法

2.3.3Determinant（行列式）

定义：对于一个m*m的方阵A，其行列式记作det(A)的值如下：

1. If $m = 1$, $\det(\mathbf{A}) = a_{11}$
2. If $m > 1$, $\det(\mathbf{A}) = \sum_{i=1\ldots m} (-1)^{i+j} \times a_{ij} \times \det(\mathbf{A}_{ij})$

where A_{ij} is a matrix obtained from A by deleting the i-th row and j-th column.

注意只有方阵存在行列式。

以任何一行或者一列为基准计算行列式得到的都是行列式的值，在INT202中只考察2*2方阵的行列式：即左上至右下对角线的值相乘减去另一条对角线。

2.3.4Residue matrices

Cryptography uses residue matrices:
In Zn, all operations on residue matrices are performed the same as for the integer matrices except that the operations are done in modular arithmetic.
A residue matrix has a multiplicative inverse if the determinant of the matrix has a multiplicative inverse in Zn => gcd(det(A), n) = 1.

2.4Linear congruence

2.4.1Single-variable linear equations 单变量线性方程

Equations of the form ax ≡ b (mod n) might have no solution or a limited number of solutions.

Assume that the gcd $(a, n) = d$.

If $d \nmid b$, there is no solution.

If $d|b$, there are d solutions.

关于一次同余方程的解法和性质有下述定理：

1.设(a, m) = 1，m>0，则同余式ax≡b(mod m)恰有一个解；

2.设(a, m) = d，m>0，则同余式ax≡b(mod m)有解的充分必要条件是d|b，此时恰有d个解。

根据以上两个定理，同余方程ax≡b (mod m)在a≠0且(a,m)|b的条件下，必有(a,m)个关于模m互不同余的解。又根据最大公约数的性质，必有二整数x、y，能使ax+my=(a,m)。由于(a,m)|b，所以有 $x_0 = bx/(a, m)$ ，$y_0 = by/(a, m)$ ，使 $ax_0 + my_0 = b$ ，由此即可得到原方程的(a,m)个关于模m互不同余的解为 $x_0 = x + mt/(a, m), t = 0, 1, 2, \cdots, (a, m) = 1$ 。

Examples：

1.10x≡2(mod 15)

2.14x≡12(mod 18)

3.3x+4≡6(mod 13)

Solutions:

1.gcd(10,15)=5，5不能整除2，所以无解。

2.首先，gcd(14,18)=2能整除12，所以有两个解。原式子简化14x≡12(mod 18)➡️7x≡6(mod 9)，然后左右同时乘以7的乘法逆元也就是4，x≡24(mod 9)，第一个解即为24(mod 9)的值为6，第二个解为6+9=15。

3.左右同时减去4的加法逆元：3x≡2(mod 13)，然后可计算出gcd(3,13)=1，有一个解，两件=边乘以3的乘法逆元9，x≡18(mod 13)，所以解为x=5.