Contemporary Cryptography：Principles and Practice

Chapter 4 Finite Field

Fuyou Miao,　Wenchao Huang

2024 年 9 月 27 日

Link

http://staff.ustc.edu.cn/∼huangwc/crypto.html

Motivation

- The problem of DES
- AES cipher and elliptic curve

- A **field (域)** is a set of elements on which two arithmetic operations (addition and multiplication) have been defined and which has the properties of ordinary arithmetic, such as closure, associativity, commutativity, distributivity, and having both additive and multiplicative inverses.
- **Modular arithmetic (模算术)** is a kind of integer arithmetic that reduces all numbers to one of a fixed set [0, ..., n - 1] for some number n. Any integer outside this range is reduced to one in this range by taking the remainder after divi- sion by n.
- The **greatest common divisor (最大公因子)** of two integers is the largest positive integer that exactly divides both integers.
- A **finite field (有限域)** is simply a field with a finite number of elements. It can be shown that the order of a finite field (number of elements in the field) must be a power of a prime p^n, where n is a positive integer.
- Finite fields of **order (阶)** p can be defined using arithmetic mod p.
- Finite fields of **order** p^n, for $n > 1$, can be defined using arithmetic over polynomials.

Outline

1. Groups, Rings, Fields (群、环、域)
2. Euclid Algorithm (欧几里得算法)
3. Modular arithmetic (模算术)
4. Galois Fields GF(p)
5. Polynomial Arithmetic (多项式运算)
6. Galois Fields GF(2^n)

Outline

1. Groups, Rings, Fields (群、环、域)
2. Euclid Algorithm (欧几里得算法)
3. Modular arithmetic (模算术)
4. Galois Fields GF(p)
5. Polynomial Arithmetic (多项式运算)
6. Galois Fields GF(2^n)

群 Groups

Definition: 群 $\{G, \cdot\}$

A **group** G, sometimes denoted by $\{G, \cdot\}$, is a set of elements with a binary operation denoted by that associates to each ordered pair (a, b) of elements in G an element $(a \cdot b)$ in G, such that:

- A1 封闭性 Closure: If a and b belongs to G, then $a \cdot b$ belongs to G.
- A2 结合律 Associative: $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all a, b, c in G
- A3 单位元 Identity element: There is an element e in G such that $a \cdot e = e \cdot a = a$
- A4 逆元 Inverse element: For each a in G, there is an element a' in G, such that $a \cdot a = a \cdot a = e$

群 Groups

Example: Groups

Let $N_n = \{1, 2, , n\}$ denote a set of n distinct symbols.
A permutation (置换) of n distinct symbols is a one-to-one mapping from N_n to N_n.

群 Groups

Example: Groups

Let $N_n = \{1, 2, , n\}$ denote a set of n distinct symbols.
A permutation (置换) of n distinct symbols is a one-to-one mapping from N_n to N_n.
Define S_n to be the set of all permutations of n distinct symbols.
Each element of S_n is a permutation of the integers p in $\{1, 2, , n\}$.

群 Groups

Example: Groups

Let $N_n = \{1, 2, , n\}$ denote a set of n distinct symbols.
A permutation (置换) of n distinct symbols is a one-to-one mapping from N_n to N_n.
Define S_n to be the set of all permutations of n distinct symbols.
Each element of S_n is a permutation of the integers p in $\{1, 2, , n\}$.

A1 If $\pi, \rho \in S_n$, then the composite mapping $\pi \cdot \rho$ is formed by permuting the elements of ρ according to the permutation π. For example, $\{3, 2, 1\}\{1, 3, 2\} = \{2, 3, 1\}$. Clearly, $\pi \cdot \rho \in S_n$.

群 Groups

Example: Groups

Let $N_n = \{1, 2, , n\}$ denote a set of n distinct symbols.
A permutation (置换) of n distinct symbols is a one-to-one mapping from N_n to N_n.
Define S_n to be the set of all permutations of n distinct symbols.
Each element of S_n is a permutation of the integers p in $\{1, 2, , n\}$.

A1 If $\pi, \rho \in S_n$, then the composite mapping $\pi \cdot \rho$ is formed by permuting the elements of ρ according to the permutation π. For example, $\{3, 2, 1\}\{1, 3, 2\} = \{2, 3, 1\}$. Clearly, $\pi \cdot \rho \in S_n$.

A2 The composition of mappings is also easily seen to be associative

群 Groups

Example: Groups

Let $N_n = \{1, 2, , n\}$ denote a set of n distinct symbols.
A permutation (置换) of n distinct symbols is a one-to-one mapping from N_n to N_n.
Define S_n to be the set of all permutations of n distinct symbols.
Each element of S_n is a permutation of the integers p in $\{1, 2, , n\}$.

A1 If $\pi, \rho \in S_n$, then the composite mapping $\pi \cdot \rho$ is formed by permuting the elements of ρ according to the permutation π. For example, $\{3, 2, 1\}\{1, 3, 2\} = \{2, 3, 1\}$. Clearly, $\pi \cdot \rho \in S_n$.

A2 The composition of mappings is also easily seen to be associative

A3 The identity mapping is the permutation that does not alter the order of the n elements. For S_n, the identity element is $\{1, 2, , n\}$.

群 Groups

Example: Groups

Let $N_n = \{1, 2, , n\}$ denote a set of n distinct symbols.
A permutation (置换) of n distinct symbols is a one-to-one mapping from N_n to N_n.
Define S_n to be the set of all permutations of n distinct symbols.
Each element of S_n is a permutation of the integers p in $\{1, 2, , n\}$.

A1 If $\pi, \rho \in S_n$, then the composite mapping $\pi \cdot \rho$ is formed by permuting the elements of ρ according to the permutation π. For example, $\{3, 2, 1\}\{1, 3, 2\} = \{2, 3, 1\}$. Clearly, $\pi \cdot \rho \in S_n$.

A2 The composition of mappings is also easily seen to be associative

A3 The identity mapping is the permutation that does not alter the order of the n elements. For S_n, the identity element is $\{1, 2, , n\}$.

A4 For any $\pi \in S_n$, the mapping that undoes the permutation defined by p is the inverse element for p. There will always be such an inverse. For example $\{2, 3, 1\}\{3, 1, 2\} = \{1, 2, 3\}$

群 Groups

Definition：Finite Group (有限群) and Infinite Group (无限群)

- If a group has a finite number of elements, it is referred to as a **finite group**. The **order (阶)** of the group is equal to the number of elements in the group
- **Otherwise**, the group is an **infinite group**.

Definition：Abelian Group （阿贝尔群, 交换群)

A5 交换律 Commutative: $a \cdot b = b \cdot a$ for all a, b in G

Definition：Cyclic Group (循环群)

- A group G is **cyclic** if every element of G is a power a^k (k is an integer) of a fixed element a ($a \in G$)
- The element a is said to generate the group G or to be a **generator (生成元)** of G.

环 Rings

Definition: Rings (环) $\{R, +, \times\}$

A ring R, sometimes denoted by $\{R, +, \times\}$, is a set of elements with two binary operations, called addition and multiplication, such that:

- [A1-A5] R is an abelian group with respect to addition. Denote the identity element as 0 and the inverse of a as a.

M1 Closure 乘法封闭性 If a and b belongs to R, then ab belongs to R.

M2 Associativity 乘法结合律 $a(bc) = (ab)c$ for all a, b, c in R

M3 Distributive laws 分配律 $a(b + c) = ab + ac$ for all a, b, c in R

Definition: Commutative Ring (交换环)

M4 Commutativity of multiplication 乘法交换律 $ab = ba$ for all a, b in R

Definition: Integral Domain (整环)

M5 Multiplicative identity 1 乘法单位元 $a1 = 1a = a$ for all a in R

M6 No zero divisors 无零因子 $a, b \in R \wedge ab = 0 \Rightarrow a = 0 \vee b = 0$.

域 Fields

Definition: Fields (域) $\{F, +, \times\}$

A set of elements (F) with two binary operations, called addition and multiplication

- [A1-M6] F is an integral domain

M7 Multiplicative inverse (乘法逆元) For each a in F, except 0, there is an element a^{-1} in F such that $aa^{-1} = (a^{-1})a = 1$.

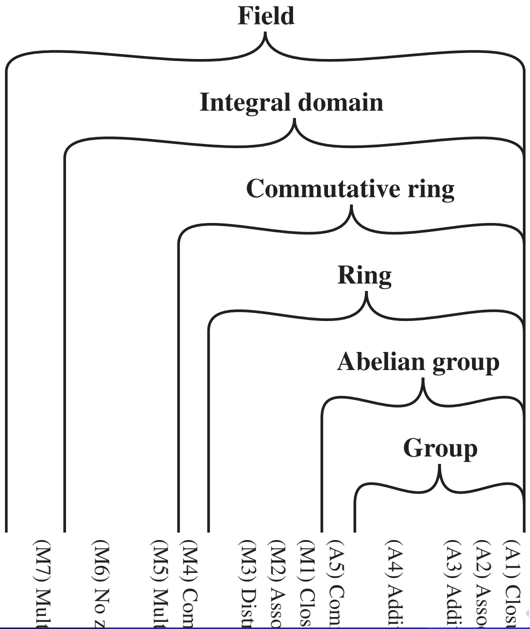
Examples

- rational numbers (有理数集合)
- the real numbers (实数集合)
- the complex number (复数集合)

Counter-Examples

The set of all integers (整数集合) is not a field, because not every element of the set has a multiplicative inverse.

群、环、域的关系


Field
Integral domain
Commutative ring
Ring
Abelian group
Group
(M7) Mult
(M6) No z
(M5) Mult
(M4) Com
(M3) Distr
(M2) Asso
(M1) Clos
(A5) Com
(A4) Addi
(A3) Addi
(A2) Asso
(A1) Clos

Outline

1 Groups, Rings, Fields (群、环、域)

2 Euclid Algorithm (欧几里得算法)

3 Modular arithmetic (模算术)

4 Galois Fields GF(p)

5 Polynomial Arithmetic (多项式运算)

6 Galois Fields GF(2^n)

Euclid Algorithm (欧几里得算法)

Definition: Divisibility (整除)

We say that a nonzero b **divides** a if $a = mb$ for some m, where a, b and m are integers

Definition: $b \mid a$

The notation $b \mid a$ is commonly used to mean b divides a

Definition: Divisor (因数)

If $b \mid a$, we say that b is a **divisor** of a

Euclid Algorithm (欧几里得算法)

The Division Algorithm

Given any positive integer n and any nonnegative integer a, if we divide a by n, we get an integer **quotient (商)** q and an integer **remainder (余数)** r that obey the following relationship:

$$a = qn + r$$

where $0 \leq r < n; q = \lfloor a/n \rfloor$, and $\lfloor x \rfloor$ is the largest integer less than or equal to x

Euclid Algorithm (欧几里得算法)

The Division Algorithm

Given any positive integer n and any nonnegative integer a, if we divide a by n, we get an integer **quotient (商)** q and an integer **remainder (余数)** r that obey the following relationship:

$$a = qn + r$$

where $0 \leq r < n; q = \lfloor a/n \rfloor$, and $\lfloor x \rfloor$ is the largest integer less than or equal to x

Definition: Greatest Common Divisor (最大公因子) $\gcd(a, b)$

The positive integer c is said to be the **greatest common divisor** of a and b if

1. c is a divisor of a and of b
2. Any divisor of a and b is a divisor of c

Euclid Algorithm (欧几里得算法)

EUCLID(a, b)

```
1 $A \leftarrow a; B \leftarrow b$
2 if $B = 0$ return
  $A = \gcd(a, b)$
3 $R = A \mod B$
4 $A \leftarrow B$
5 $B \leftarrow R$
6 goto 2
```

Example：Compute $\gcd(1970, 1066)$

$$\begin{array}{rcl}
1970 & = 1 \times 1066 + 904 & \gcd(1066, 904) \\
1066 & = 1 \times 904 + 162 & \gcd(904, 162) \\
904 & = 5 \times 162 + 94 & \gcd(162, 94) \\
162 & = 1 \times 94 + 68 & \gcd(94, 68) \\
94 & = 1 \times 68 + 26 & \gcd(68, 26) \\
68 & = 2 \times 26 + 16 & \gcd(26, 16) \\
26 & = 1 \times 16 + 10 & \gcd(16, 10) \\
16 & = 1 \times 10 + 6 & \gcd(10, 6) \\
10 & = 1 \times 6 + 4 & \gcd(6, 4) \\
6 & = 1 \times 4 + 2 & \gcd(4, 2) \\
4 & = 2 \times 2 + 0 & \gcd(2, 0)
\end{array}$$

Modular arithmetic (模算术)

The Modulus (模)

If a is an integer and n is a positive integer,

- We define $\underline{a \mod n}$ to be the remainder when a is divided by n.
- The integer n is called the **modulus**

Rewrite r in Equations

$$a = \lfloor a/n \rfloor \times n + (a \mod n)$$

Modular arithmetic (模算术)

Definition: Congruent Modulo n (同余)

Two integers a and b are said to be **congruent modulo** n, if $(a \mod n) = (b \mod n)$

- written as $a \equiv b \pmod{n}$

Properties of Congruences

1. $a \equiv b \pmod{n}$, if $n \mid (a - b)$
2. $a \equiv b \pmod{n}$ implies $b \equiv a \pmod{n}$
3. $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$ imply $a \equiv c \pmod{n}$

Modular arithmetic (模算术)

Definition : Modular arithmetic (模算术)

Note that the $\mod n$ operator maps all integers into the set of integers $\{0, 1, \ldots, (n-1)\}$.
Thus, **modular arithmetics** performs arithmetic operations within the set.

Properties of modular arithmetics

- $[(a \mod n) + (b \mod n)] \mod n = (a + b) \mod n$
- $[(a \mod n) - (b \mod n)] \mod n = (a - b) \mod n$
- $[(a \mod n) \times (b \mod n)] \mod n = (a \times b) \mod n$

Modular arithmetic (模算术)

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

表: Addition Modulo 8

×	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

表: Multiplication modulo 8

Modular arithmetic (模算术)

Definition: additive inverse

If there exists z, such that

$$w + z = 0 \mod n$$

then z is **additive inverse** of w, denoted as $-w$

Definition: multiplicative inverse

If there exists z, such that

$$w * z = 1 \mod n$$

then z is **multiplicative inverse** of w, denoted as w^{-1}

表: Additive and multiplicative inverses modulo 8

w	0	1	2	3	4	5	6	7
$-w$	0	7	6	5	4	3	2	1
w^{-1}	–	1	–	3	–	5	–	7

Modular arithmetic (模算术)

Definition：set of residues, or residue classes (剩余类集 剩余集)

Define the set Z_n as the set of nonnegative integers less than n:

$$Z_n = \{0, 1, , (n-1)\}$$

Definition: residue class (剩余类)

We can label the residue classes (mod n) as $[0], [1], [2], \ldots, [n-1]$, where

$$[r] = \{a \mid a \in Z \wedge (a \equiv r \mod n)\}$$

Example: the residue classes (mod 4)

$$[0] = \{\ldots, -8, -4, 0, 4, 8, \ldots, \}, \quad [1] = \{\ldots, -7, -3, 1, 5, 9, \ldots, \}$$

$$[2] = \{\ldots, -6, -2, 2, 6, 10, \ldots, \}, \quad [3] = \{\ldots, -5, -1, 3, 7, 11, \ldots, \}$$

Modular arithmetic (模算术)

表: Properties of Modular Arithmetic for Integers in Z_n

Property	Expression
Commutative Laws	$(w+x) \mod n = (x+w) \mod n$ $(w \times x) \mod n = (x \times w) \mod n$
Associative Laws	$[(w+x)+y] \mod n = [w+(x+y)] \mod n$ $[(w \times x) \times y] \mod n = [w \times (x \times y)] \mod n$
Distributive law	$[w \times (x+y)] \mod n = [(w \times x)+(w \times y)] \mod n$
Identities	$(0+w) \mod n = w \mod n$ $(1 \times w) \mod n = w \mod n$
Additive inverse	$\forall w \in Z_n$, there exists z such that $w+z=0 \mod n$

Outline

1. Groups, Rings, Fields (群、环、域)
2. Euclid Algorithm (欧几里得算法)
3. Modular arithmetic (模算术)
4. Galois Fields GF(p)
5. Polynomial Arithmetic (多项式运算)
6. Galois Fields GF(2^n)

Galois Fields GF(p)

Definition: GF(p)

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_p$$

together with the arithmetic operations modulo p.

Definition: GF(p^n) ????

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_{p^n}$$

together with the arithmetic operations ????.

Galois Fields GF(p)

Definition: GF(p)

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_p$$

together with the arithmetic operations modulo p.

Property: Multiplicative inverse (w^{-1})

For each $w \in Z_p$, $w \neq 0$, there exists $z \in Z_p$, such that

$$w \times z \equiv 1 \mod p$$

表: Multiplicative inverses modulo 8

w	0	1	2	3	4	5	6	7
w^{-1}	–	1	–	3	–	5	–	7

表: Multiplicative inverses modulo 7

w	0	1	2	3	4	5	6
w^{-1}	–	1	4	5	2	3	6

Galois Fields GF(p)

Finding the Multiplicative Inverse in GF(p): Extended EUCLID(m,b)

```
1. $(A_1, A_2, A_3) \leftarrow (1, 0, m); (B_1, B_2, B_3) \leftarrow (0, 1, b)$
2. if $B_3 = 0$ return "There exists no multiplicative inverse"
3. if $B_3 = 1$ return $B_2 = b^{-1} \mod m$
4. $Q = \lfloor \frac{A_3}{B_3} \rfloor$
5. $(T_1, T_2, T_3) \leftarrow (A_1 - QB_1, A_2 - QB_2, A_3 - QB_3)$
6. $(A_1, A_2, A_3) \leftarrow (B_1, B_2, B_3)$
7. $(B_1, B_2, B_3) \leftarrow (T_1, T_2, T_3)$
8. goto 2
```

Property: Loop Invariant — And finally $bB_2 \equiv 1 \mod m$

The property that holds at every beginning of the loop:

$$mT_1 + bT_2 = T_3, mA_1 + bA_2 = A_3, mB_1 + bB_2 = B_3$$

Galois Fields GF(p)

Example： Compute the multiplicative inverse of 550 in GF(1759)

Q	A_1	A_2	A_3	B_1	B_2	B_3
–	1	0	1759	0	1	550
3	0	1	550	1	-3	109
5	1	-3	109	-5	16	5
21	-5	16	5	106	-339	4
1	106	-339	4	-111	355	1

Outline

1 Groups, Rings, Fields (群、环、域)

2 Euclid Algorithm (欧几里得算法)

3 Modular arithmetic (模算术)

4 Galois Fields GF(p)

5 **Polynomial Arithmetic (多项式运算)**

6 Galois Fields GF(2^n)

Polynomial Arithmetic (多项式运算)

Three classes of polynomial arithmetic

1. Ordinary polynomial arithmetic, using the basic rules of algebra.

Polynomial Arithmetic (多项式运算)

Three classes of polynomial arithmetic

1. Ordinary polynomial arithmetic, using the basic rules of algebra.
2. Polynomial arithmetic in which the arithmetic on the coefficients (系数) is performed modulo p ; that is, the coefficients are in GF(p).

Polynomial Arithmetic (多项式运算)

Three classes of polynomial arithmetic

1. Ordinary polynomial arithmetic, using the basic rules of algebra.
2. Polynomial arithmetic in which the arithmetic on the coefficients (系数) is performed modulo p ; that is, the coefficients are in GF(p).
3. Polynomial arithmetic in which the coefficients are in GF(p), and the polynomials are defined modulo a polynomial $m(x)$ whose highest power is some integer n

Polynomial Arithmetic (多项式运算)

Three classes of polynomial arithmetic

1. Ordinary polynomial arithmetic, using the basic rules of algebra.
2. Polynomial arithmetic in which the arithmetic on the coefficients (系数) is performed modulo p ; that is, the coefficients are in GF(p).
3. Polynomial arithmetic in which the coefficients are in GF(p), and the polynomials are defined modulo a polynomial $m(x)$ whose highest power is some integer n

Arithmetic 1: Ordinary polynomial arithmetic

A polynomial of degree n (integer $(n \geq 0)$) is an expression of the form

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 = \sum_{i=0}^{n} a_i x^i$$

where the a_i are elements of some designated set of numbers S, called the coefficient set, and $a_n \neq 0$.
We say that such polynomials are defined over the coefficient set S.

Polynomial Arithmetic (多项式运算)

Arithmetic 1: Addition and Multiplication

If $f(x) = \sum_{i=0}^{n} a_i x^i$, $g(x) = \sum_{i=0}^{m} b_i x^i$, $n \geq m$,
then addition is defined as:

$$f(x) + g(x) = \sum_{i=0}^{m} (a_i + b_i) x^i + \sum_{i=m+1}^{n} a_i x^i$$

then multiplication is defined as:

$$f(x) \times g(x) = \sum_{i=0}^{n+m} c_i x^i, \quad c_k = a_0 b_k + a_1 b_{k-1} + \cdots + a_{k-1} b_1 + a_k b_0$$

Polynomial Arithmetic (多项式运算)

Arithmetic 1: Example

If $f(x) = x^3 + x^2 + 2, g(x) = x^2 - x + 1$ then:

$$\begin{array}{rcl} f(x) + g(x) & = & x^3 + 2x^2 - x + 3 \\ f(x) - g(x) & = & x^3 + x + 1 \\ f(x) \times g(x) & = & x^5 + 3x^2 - 2x + 2 \\ f(x) \ / \ g(x) & = & ?? \end{array}$$

Polynomial Arithmetic (多项式运算)

Arithmetic 2: Polynomial Arithmetic with Coefficients in Z_p

The coefficients are elements of some field Z_p

Example :$p = 2$， such that coefficients are 0 or 1

If $f(x) = x^3 + x^2, g(x) = x^2 + x + 1$， then

$$f(x) + g(x) = x^3 + x + 1$$

$$f(x) \times g(x) = x^5 + x$$

If $f(x) = x^7 + x^5 + x^4 + x^3 + x + 1$, $g(x) = x^3 + x + 1$, then

$$f(x) + g(x) = f(x) - g(x) = x^7 + x^5 + x^4$$

$$f(x) \times g(x) = x^{10} + x^4 + x^2 + 1$$

$$f(x)/g(x) = x^4 + 1$$

Polynomial Arithmetic (多项式运算)

Definition: Irreducible

A polynomial $f(x)$ over a field F is called **irreducible (不可约)** if and only if:

- $f(x)$ cannot be expressed as a product of two polynomials, both over F, and both of degree lower than that of $f(x)$.

Example: reducible

The polynomial $f(x) = x^4 + 1$ over GF(2) is reducible, because

$$x^4 + 1 = (x+1)(x^3 + x^2 + x + 1)$$

Polynomial Arithmetic (多项式运算)

Definition：Greatest Common Divisor $c(x)$ 最大公因式

The polynomial $c(x)$ is said to be the greatest common divisor of $a(x)$ and $b(x)$, if

- $c(x)$ divides both $a(x)$ and $b(x)$
- Any divisor of $a(x)$ and $b(x)$ is a divisor of $c(x)$

Property

$$\gcd[a(x), b(x)] = \gcd[b(x), a(x) \mod b(x)]$$

Polynomial Arithmetic (多项式运算)

EUCLID(a, b) (最大公因子)

```
1. $A \leftarrow a; B \leftarrow b$
2. if $B = 0$ return $A = \gcd(a, b)$
3. $R = A \mod B$
4. $A \leftarrow B$
5. $B \leftarrow R$
6. goto 2
```

EUCLID$(a(x), b(x))$ (最大公因式)

```
1. $A(x) \leftarrow a(x); B(x) \leftarrow b(x)$
2. if $B(x) = 0$ return $A(x) = \gcd[a(x), b(x)]$
3. $R(x) = A(x) \mod B(x)$
4. $A(x) \leftarrow B(x)$
5. $B(x) \leftarrow R(x)$
6. goto 2
```

Polynomial Arithmetic (多项式运算)

Example： Compute $\gcd[a(x), b(x)]$

其中， $a(x) = x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$, $b(x) = x^4 + x^2 + x + 1$

1. $A(x) = a(x), B(x) = b(x), R(x) = x^3 + x^2 + 1$
2. $A(x) = x^4 + x^2 + x + 1, B(x) = x^3 + x^2 + 1, R(x) = 0$
3. $\gcd[a(x), b(x)] = x^3 + x^2 + 1$

Outline

1. Groups, Rings, Fields (群、环、域)
2. Euclid Algorithm (欧几里得算法)
3. Modular arithmetic (模算术)
4. Galois Fields GF(p)
5. Polynomial Arithmetic (多项式运算)
6. Galois Fields GF(2^n)

Galois Fields GF(2^n)

Recall Definition: GF(p)

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_p$$

together with the arithmetic operations modulo p.

Definition: GF(p^n) ????

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_{p^n}$$

together with the arithmetic operations ????.

Galois Fields GF(2^n)

Recall Definition: GF(p)

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_p$$

together with the arithmetic operations $\underline{\text{modulo } p}$.

Definition: GF(p^n) ????

For a given prime, p, we define the finite field of order p, GF(p), as the set

$$Z_{p^n}$$

together with the arithmetic operations ????.

Motivation

Enable Block cipher:

- Finite fields
- multiplicative inverse
- bit block

Galois Fields GF(2^n)

Recall: Three classes of polynomial arithmetic

1. Ordinary polynomial arithmetic, using the basic rules of algebra.
2. Polynomial arithmetic in which the arithmetic on the coefficients (系数) is performed modulo p ; that is, the coefficients are in GF(p).

Galois Fields GF(2^n)

Recall: Three classes of polynomial arithmetic

1. Ordinary polynomial arithmetic, using the basic rules of algebra.
2. Polynomial arithmetic in which the arithmetic on the coefficients (系数) is performed modulo p ; that is, the coefficients are in GF(p).
3. Polynomial arithmetic in which the coefficients are in GF(p), and the polynomials are defined modulo a polynomial $m(x)$ whose highest power is some integer n

Galois Fields GF(2^n)

Arithmetic 3: Modular Polynomial Arithmetic

Consider the set S of all polynomials of degree $n-1$ or less over the field Z_p. Thus, each polynomial has the form

$$f(x) = a_{n-1}x^{n-1} + \cdots + a_1 x + a_0 = \sum_{i=0}^{n-1} a_i x^i$$

where $\underline{a_i \in \{0, 1, \ldots, p-1\}}$.

Arithmetic 3: Modular Polynomial Arithmetic

Consider the set S of all polynomials of degree $n-1$ or less over the field Z_p. Thus, each polynomial has the form

$$f(x) = a_{n-1}x^{n-1} + \cdots + a_1 x + a_0 = \sum_{i=0}^{n-1} a_i x^i$$

where $\underline{a_i \in \{0, 1, \ldots, p-1\}}$.

- Arithmetic follows the ordinary rules of polynomial arithmetic using the basic rules of algebra, with the following two refinements.

Galois Fields GF(2^n)

Arithmetic 3: Modular Polynomial Arithmetic

Consider the set S of all polynomials of degree $n-1$ or less over the field Z_p. Thus, each polynomial has the form

$$f(x) = a_{n-1}x^{n-1} + \cdots + a_1x + a_0 = \sum_{i=0}^{n-1} a_i x^i$$

where $\underline{a_i \in \{0, 1, \ldots, p-1\}}$.

- Arithmetic follows the ordinary rules of polynomial arithmetic using the basic rules of algebra, with the following two refinements.
- Arithmetic on the coefficients is performed modulo p. That is, we use the rules of arithmetic for the finite field Z_p.

Galois Fields GF(2^n)

Arithmetic 3: Modular Polynomial Arithmetic

Consider the set S of all polynomials of degree $n-1$ or less over the field Z_p. Thus, each polynomial has the form

$$f(x) = a_{n-1}x^{n-1} + \cdots + a_1 x + a_0 = \sum_{i=0}^{n-1} a_i x^i$$

where $\underline{a_i \in \{0, 1, \ldots, p-1\}}$.

- Arithmetic follows the ordinary rules of polynomial arithmetic using the basic rules of algebra, with the following two refinements.
- Arithmetic on the coefficients is performed modulo p. That is, we use the rules of arithmetic for the finite field Z_p.
- If multiplication results in a polynomial of degree greater than $n-1$, then the polynomial is reduced modulo some **irreducible polynomial** $m(x)$ of degree n. That is, we divide by $m(x)$ and keep the remainder. For a polynomial $f(x)$, the remainder is expressed as $\underline{r(x) = f(x) \mod m(x)}$.

Galois Fields GF(2^n)

Example

The Advanced Encryption Standard (AES) uses arithmetic in the finite field GF(2^8), with the irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$. Consider the two polynomials $f(x) = x^6 + x^4 + x^2 + x + 1$ and $g(x) = x^7 + x + 1$. Then

Galois Fields GF(2^n)

Example

The Advanced Encryption Standard (AES) uses arithmetic in the finite field GF(2^8), with the irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$. Consider the two polynomials $f(x) = x^6 + x^4 + x^2 + x + 1$ and $g(x) = x^7 + x + 1$. Then

$$\begin{aligned} f(x) + g(x) &= x^6 + x^4 + x^2 + x + 1 + x^7 + x + 1 \\ &= x^7 + x^6 + x^4 + x^2 \end{aligned}$$

Galois Fields GF(2^n)

Example

The Advanced Encryption Standard (AES) uses arithmetic in the finite field GF(2^8), with the irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$. Consider the two polynomials $f(x) = x^6 + x^4 + x^2 + x + 1$ and $g(x) = x^7 + x + 1$. Then

$$\begin{aligned} f(x) \times g(x) &= x^{13} + x^{11} + x^9 + x^8 + x^7 \\ &\quad + x^7 + x^5 + x^3 + x^2 + x \\ &\quad + x^6 + x^4 + x^2 + x + 1 \\ &= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \\ &= x^7 + x^6 + 1 \end{aligned}$$

Galois Fields GF(2^n)

Finding the Multiplicative Inverse in GF(2^n)

Refer to: Extended Euclid Algorithm

Galois Fields GF(2^n) – Using a Generator

Definition: Generator g

A **generator (生成元)** g of a finite field F of order q (contains q elements) is an element whose first $q-1$ powers generate all the nonzero elements of F.

$$0, g^0, g^1, \ldots, g^{q-2}$$

Properties

Consider a field F defined by a polynomial $f(x)$. An element b contained in F is called a root of the polynomial if $f(b) = 0$.
Finally, it can be shown that a root g of an irreducible polynomial is a generator of the finite field defined on that polynomial.

Galois Fields GF(2^n)

Using a Generator

Example

Let us consider the finite field GF(2^3), defined over the irreducible polynomial $x^3 + x + 1$, discussed previously. Thus, the generator g must satisfy $f(g) = g^3 + g + 1 = 0$

$$\begin{aligned}
0 &= & & 0 \\
g^0 &= & & 1 \\
g^1 &= & & g \\
g^2 &= & & g^2 \\
g^3 &= & & g+1 \\
g^4 &= & g(g^3) = g(g+1) = & \quad g^2+g \\
g^5 &= & g(g^4) = g(g^2+g) = & \quad g^2+g+1 \\
g^6 &= & g(g^5) = g(g^2+g+1) = & \quad g^2+1 \\
g^7 &= & g(g^6) = g(g^2+1) = & \quad 1
\end{aligned}$$

Conclusion

1. Groups, Rings, Fields (群、环、域)
2. Euclid Algorithm (欧几里得算法)
3. Modular arithmetic (模算术)
4. Galois Fields GF(p)
5. Polynomial Arithmetic (多项式运算)
6. Galois Fields GF(2^n)

Homework

4.6 For each of the following equations, find an integer x that satisfies the equation.

a. $5x \equiv 4 \pmod{3}$

b. $7x \equiv 6 \pmod{5}$

c. $9x \equiv 8 \pmod{7}$

4.7 In this text, we assume that the modulus is a positive integer. But the definition of the expression $a \bmod n$ also makes perfect sense if n is negative. Determine the following:

a. $5 \bmod 3$

b. $5 \bmod -3$

c. $-5 \bmod 3$

d. $-5 \bmod -3$

4.9 In Section 4.3, we define the congruence relationship as follows: Two integers a and b are said to be congruent modulo n if $(a \bmod n) = (b \bmod n)$. We then proved that $a \equiv b \pmod{n}$ if $n \mid (a - b)$. Some texts on number theory use this latter relationship as the definition of congruence: Two integers a and b are said to be congruent modulo n if $n \mid (a - b)$. Using this latter definition as the starting point, prove that, if $(a \bmod n) = (b \bmod n)$, then n divides $(a - b)$.

Homework

4.11 Prove the following:

a. $a \equiv b \pmod{n}$ implies $b \equiv a \pmod{n}$

b. $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$ imply $a \equiv c \pmod{n}$

4.12 Prove the following:

a. $[(a \bmod n) - (b \bmod n)] \bmod n = (a - b) \bmod n$

b. $[(a \bmod n) \times (b \bmod n)] \bmod n = (a \times b) \bmod n$

4.13 Find the multiplicative inverse of each nonzero element in Z_5.

4.19 Using the extended Euclidean algorithm, find the multiplicative inverse of

a. 1234 mod 4321

b. 24140 mod 40902

c. 550 mod 1769

Homework

4.23 For polynomial arithmetic with coefficients in Z_{10}, perform the following calculations.

a. $(7x + 2) - (x^2 + 5)$

b. $(6x^2 + x + 3) \times (5x^2 + 2)$

4.24 Determine which of the following are reducible over GF(2).

a. $x^3 + 1$

b. $x^3 + x^2 + 1$

c. $x^4 + 1$ (be careful)

4.27 Determine the multiplicative inverse of $x^3 + x + 1$ in $\mathrm{GF}(2^4)$ with $m(x) = x^4 + x + 1$.