1.5 Tutorial in Number Theory and Algebra

In this section, we collect various definitions and facts from number theory and algebra that are essential in the design and analysis of cryptographic algorithms. We do not provide proofs; proofs can be found in appropriate mathematics textbooks.

Modular arithmetic

Suppose a and b are integers, and m is a positive integer. Then we write $a \equiv b \pmod{m}$ if m divides $b - a$. The phrase $a \equiv b \pmod{m}$ is called a *congruence*, and it is read as "a is *congruent* to b modulo m." The integer m is called the *modulus*.

For example, we have $11 \equiv -25 \pmod{6}$ because $6 \mid (11 - (-25))$.

We will use the notation $a \bmod m$ (without parentheses) to denote the non-negative remainder when a is divided by m. Thus, $a \equiv b \pmod{m}$ if and only if $a \bmod m = b \bmod m$. If we replace a by $a \bmod m$, we say that a is *reduced* modulo m.

For example, to compute $101 \bmod 7$, we write $101 = 7 \times 14 + 3$. Since $0 \leq 3 \leq 6$, it follows that $101 \bmod 7 = 3$. As another example, suppose we want to compute $(-101) \bmod 7$. In this case, we have $-101 = 7 \times (-15) + 4$. Since $0 \leq 4 \leq 6$, it follows that $(-101) \bmod 7 = 4$.

$\mathbb{Z}_n$ is is used to denote the set $\{0, \ldots, n-1\}$, which is usually equipped with the two operations $+$ and $\times$. Addition and multiplication in $\mathbb{Z}_n$ work exactly like real addition and multiplication, except that the results are reduced modulo n.

For example, suppose we want to compute 11×13 in $\mathbb{Z}_{16}$. As integers, we have $11 \times 13 = 143$. Then we reduce 143 modulo 16 as described above: $143 = 8 \times 16 + 15$, so $143 \bmod 16 = 15$, and hence $11 \times 13 = 15$ in $\mathbb{Z}_{16}$.

Finite Abelian Groups

A *finite abelian group* is a pair $(X, \star)$, where X is a finite set and $\star$ is a binary operation defined on X, that satisfies the following properties:

1. the operation $\star$ is *closed*, i.e., $a \star b \in X$ for any $a, b \in X$
2. the operation $\star$ is *commutative*, i.e., $a \star b = b \star a$ for any $a, b \in X$
3. the operation $\star$ is *associative*, i.e., $(a \star b) \star c = a \star (b \star c)$ for any $a, b, c \in X$

4. There is an element $id \in X$ called the *identity*, such that $a \star id = id \star a = a$ for any $a \in X$
5. for every $a \in X$, there exists an element $b \in X$ called the *inverse* of a, such that $a \star b = b \star a = id$ for any $a \in X$

The *order* of the group $(X, \star)$ is $|X|$.

We give some examples of finite groups:

1. Let $n \geq 2$ be an integer. Then $(\mathbb{Z}_n, +)$ is an abelian group of order n, where $+$ denotes addition mudulo n. The identity element is 0, and the inverse of a, usually denoted $-a$, is $(-a) \bmod n$.
2. Let $p \geq 2$ be a prime. Define $\mathbb{Z}_p{}^* = \mathbb{Z}_p \backslash \{0\}$. Then $(\mathbb{Z}_p{}^*, \cdot)$ is an abelian group of order $p - 1$, where $\cdot$ denotes multiplication mudulo p. The identity element is 1, and the inverse of a, usually denoted a^{-1}, is computed using the EUCLIDEAN ALGORITHM.
3. We generalize the previous example. Let $n \geq 2$ be an integer. Define

$$\mathbb{Z}_n{}^* = \mathbb{Z}_n \backslash \{d \in \mathbb{Z}_n : \gcd(d, n) > 1\}.$$

Then $(\mathbb{Z}_n{}^*, \cdot)$ is an abelian group where $\cdot$ denotes multiplication mudulo n. The identity element is 1, and the inverse of a, usually denoted a^{-1}, is computed using the EUCLIDEAN ALGORITHM.

The order of $(\mathbb{Z}_n{}^*, \cdot)$ is denoted $\phi(n)$ (note that $\phi(n)$ is just the number of positive integers less than n that are relatively prime to n). $\phi(n)$ can be computed from the following formula: suppose that n has prime power factorization

$$n = \prod_{i=1}^{\ell} p_i{}^{e_i}$$

(the p_i's are distinct primes and $e_i \geq 1$ for $1 \leq i \leq \ell$). Then

$$\phi(n) = \prod_{i=1}^{\ell} p_i{}^{e_i - 1}(p_i - 1).$$

Order of Group Elements

For a finite group $(X, \star)$, define the *order* of an element $a \in X$ (denoted $\mathsf{ord}(a)$) to be the smallest positive integer m such that

$$\underbrace{a \star a \star \cdots \star a}_{m} = id.$$

If the group operation is multiplication, then

$$\underbrace{a \star a \star \cdots \star a}_{m}$$

is written as an exponentiation, a^m. If the group operation is addition, then the same expression is written as a multiplication, ma.

The identity element is defined to have order 1. It can be shown that the order of any $a \in X$ is a divisor of the order of the group. It is also possible to show, for any $a \in X$, that the order of $b = a^i$ (where, for concreteness, we assume that the group operation is written multiplicatively) is

$$\mathsf{ord}(b) = \frac{\mathsf{ord}(a)}{\gcd(\mathsf{ord}(a), i)}.$$

For example, if $\mathsf{ord}(a) = 100$ and $b = a^{35}$, then

$$\mathsf{ord}(b) = \frac{100}{\gcd(100, 35)} = \frac{100}{5} = 20.$$

Cyclic Groups

A finite abelian group $(X, \star)$ is a *cyclic group* if there exists an element $a \in X$ having order equal to $|X|$. Such an element is called a *generator* of the group.

Here are some examples of cyclic groups.

1. Let $n \geq 2$ be an integer. Then $(\mathbb{Z}_n, +)$ is a cyclic group, and 1 is a generator. Further, any element $a \in \mathbb{Z}_n$ is a generator of $(\mathbb{Z}_n, +)$ if and only if $\gcd(a, n) = 1$. The number of generators of $(\mathbb{Z}_n, +)$ is $\phi(n)$.
2. Let $p \geq 2$ be a prime. Then $({\mathbb{Z}_p}^*, \cdot)$ is cyclic group, and a generator of this group is called a *primitive element*.
   It can be shown that $\alpha \in {\mathbb{Z}_p}^*$ is a primitive element if and only if
   $$\alpha^{(p-1)/q} \not\equiv 1 \pmod{p}$$
   for all primes q such that $q|(p-1)$. The number of generators of $({\mathbb{Z}_p}^*, \cdot)$ is $\phi(p-1)$.
3. It is known that $({\mathbb{Z}_n}^*, \cdot)$ is cyclic group if and only if $n = 2, 4, p^e$ or $2\,p^e$, where p is an odd prime and e is a positive integer.

The result mentioned as item 2. provides a very useful way to test whether a given element $\alpha \in {\mathbb{Z}_p}^*$ is a primitive element (where p is an odd prime, provided that the factorization of $p-1$ is known. We illustrate with a small example, $p = 13$ and $\alpha = 2$. The factorization of 12 is $12 = 2^2 3^1$. Therefore, to verify that 2 is a primitive element modulo 13, it is sufficient to check that $2^6 \not\equiv 1 \pmod{13}$ and $2^4 \not\equiv 1 \pmod{13}$. This is much faster than checking all 12 powers of α.

Quadratic Residues

Suppose p is an odd prime and a is an integer. a is defined to be a *quadratic residue* modulo p if $a \not\equiv 0 \pmod{p}$ and the congruence $y^2 \equiv a \pmod{p}$ has a solution $y \in \mathbb{Z}_p$.

It can be shown that a is a quadratic residue modulo p if and only if

$$a^{(p-1)/2} \equiv 1 \pmod{p}.$$

Every quadratic residue modulo p has exactly two square roots in $\mathbb{Z}_p$, which are negatives of each other modulo p. If $p \equiv 3 \pmod{4}$ is prime and a is a quadratic residue modulo p, then it can be shown that the two square roots of a modulo p are $\pm a^{(p+1)/4} \bmod p$.

Suppose n is an odd integer and a is an integer. a is defined to be a *quadratic residue* modulo n if $\gcd(a, n) = 1$ and the congruence $y^2 \equiv a \pmod{n}$ has a solution $y \in \mathbb{Z}_n$. Suppose that $n > 1$ is an odd integer having factorization

$$n = \prod_{i=1}^{\ell} p_i^{\,e_i},$$

where the p_i's are distinct primes and the e_i's are positive integers. Suppose further that $\gcd(a, n) = 1$. Then a is quadratic residue modulo n if and only if

$$a^{(p_i-1)/2} \equiv 1 \pmod{p_i}$$

for all $i \in \{1, \ldots, \ell\}$. (This is equivalent to saying that a is a quadratic residue modulo p for all primes p dividing n.) If a is quadratic residue modulo n, then there are exactly 2^ℓ solutions modulo n to the congruence $y^2 \equiv a \pmod{n}$.

As an example, suppose that $p = 43$ and $a = 13$. Note that $43 \equiv 3 \pmod{4}$ is prime. It can be verified that $13^{21} \equiv 1 \pmod{43}$, so 13 is a quadratic residue modulo 43. $(p+1)/4 = 11$, so the two square roots of 13 modulo 43 are

$$\pm 13^{11} \bmod 43 = \pm 23 \bmod 43 = 20, 23.$$

1.5.1 Some Fundamental Algorithms

Important algorithms used in cryptography include the EUCLIDEAN ALGORITHM, to compute multiplicative inverses in $\mathbb{Z}_n{}^*$; the CHINESE REMAINDER THEOREM, to solve systems of linear congruences; and the SQUARE-AND-MULTIPLY algorithm, to perform efficient exponentiations in multiplicative groups. We present pseudocode descriptions of these three algorithms in this section. Then we illustrate the application of these algorithms in the decryption of ciphertexts in the *Rabin cryptosystem*.

For cryptographic purposes, we often require integers that are 1024 bits in length. In this situation, all arithmetic computations would require extended precision integers. The three algorithms described in this section are very efficient, and are practical for use with numbers of this very large size.

The Euclidean Algorithm

The multiplicative inverse $b^{-1} \bmod a$ exists if and only if $\gcd(a, b) = 1$. Algorithm 1.5 computes $\gcd(a, b)$, and if $\gcd(a, b) = 1$, then it also determines the value of $b^{-1} \bmod a$. These computations employ the the so-called EXTENDED EUCLIDEAN ALGORITHM.

Algorithm 1.5: MULTIPLICATIVE INVERSE(a, b)

$a_0 \leftarrow a$
$b_0 \leftarrow b$
$t_0 \leftarrow 0$
$t \leftarrow 1$
$q \leftarrow \lfloor \frac{a_0}{b_0} \rfloor$
$r \leftarrow a_0 - q\, b_0$
while $r > 0$
  do $\begin{cases} temp \leftarrow t_0 - q\, t \\ t_0 \leftarrow t \\ t \leftarrow temp \\ a_0 \leftarrow b_0 \\ b_0 \leftarrow r \\ q \leftarrow \lfloor \frac{a_0}{b_0} \rfloor \\ r \leftarrow a_0 - q\, b_0 \end{cases}$
if $b_0 \neq 1$
  then return ("the gcd of b and a is b_0")
  else $\begin{cases} t \leftarrow t \bmod a \\ \textbf{return} \text{ ("}b^{-1} \bmod a = t\text{")} \end{cases}$

To illustrate the application of Algorithm 1.5, we compute $28^{-1} \bmod 75$. Here are the steps performed in the computation:

a_0	b_0	t_0	t	q	r
75	28	0	1	2	19
28	19	1	−2	1	9
19	9	−2	3	2	1
9	1	3	−8	9	0

Therefore, $28^{-1} \bmod 75 = -8 \bmod 75 = 67$.

The Chinese Remainder Theorem

Suppose $m_1, \dots, m_r$ are pairwise relatively prime positive integers, and suppose $a_1, \dots, a_r$ are integers. Then the system of r congruences $x \equiv a_i \pmod{m_i}$

$(1 \leq i \leq r)$ can be shown to have a unique solution modulo $M = m_1 \times \cdots \times m_r$. The Chinese remainder theorem provides a formula to compute this solution:

$$x = \sum_{i=1}^{r} a_i M_i y_i \bmod M,$$

where $M_i = M/m_i$ and $y_i = {M_i}^{-1} \bmod m_i$, for $1 \leq i \leq r$. The following algorithm evaluates this formula.

Algorithm 1.6: CRT$(m_1, \ldots, m_r, a_1, \ldots, a_r)$

external MULTINV
$M \leftarrow 1$
for $i \leftarrow 1$ **to** r
  do $M \leftarrow M \times m_i$
for $i \leftarrow 1$ **to** r
  do $\begin{cases} M_i = M/m_i \\ y_i = \text{MULTINV}(m_i, M_i) \end{cases}$
$x \leftarrow 0$
for $i \leftarrow 1$ **to** r
  do $\{x \leftarrow x + a_i \times M_i \times y_i \bmod M$
return (x)

We illustrate the Algorithm 1.6 with an example. Suppose $r = 3$, $m_1 = 7$, $m_2 = 11$ and $m_3 = 13$. Then $M = 1001$. We compute $M_1 = 143$, $M_2 = 91$ and $M_3 = 77$, and then $y_1 = 5$, $y_2 = 4$ and $y_3 = 12$. Then the solution to the given system is

$$x = (715a_1 + 364a_2 + 924a_3) \bmod 1001.$$

For example, if $a_1 = 5$, $a_2 = 3$ and $a_3 = 10$, then

$$\begin{aligned} x &= (715 \times 5 + 364 \times 3 + 924 \times 10) \bmod 1001 \\ &= 13907 \bmod 1001 \\ &= 894. \end{aligned}$$

The Square-and-multiply Algorithm

Our next algorithm, known as the SQUARE-AND-MULTIPLY ALGORITHM, computes $z = x^c$ in any group whose operation is written multiplicatively. The algorithm assumes that the exponent, c, is represented in binary notation, say

$$c = \sum_{i=0}^{\ell-1} c_i 2^i,$$

where $c_i = 0$ or 1, $0 \leq i \leq \ell - 1$. The algorithm to compute $z = x^c$ is presented as Algorithm 1.7.

Algorithm 1.7: SQUARE-AND-MULTIPLY(x, c)

external GROUPMULT
$z \leftarrow id$
for $i \leftarrow \ell - 1$ **downto** 0
do $\begin{cases} z \leftarrow \text{GROUPMULT}(z, z) \\ \textbf{if } c_i = 1 \\ \quad \textbf{then } z \leftarrow \text{GROUPMULT}(z, x) \end{cases}$
return (z)

it can be seen that Algorithm 1.7 "processes" the exponent c from left to right. As an example, we compute $9726^{3533} \bmod 11413$, using Algorithm 1.7. The exponent $c = 3533$ has binary representation $c = 110111001101_2$. The operation GROUPMULT is multiplication in $\mathbb{Z}_{11413}$, and the algorithm proceeds as follows:

i	c_i	z
11	1	$1^2 \times 9726 = 9726$
10	1	$9726^2 \times 9726 = 2659$
9	0	$2659^2 = 5634$
8	1	$5634^2 \times 9726 = 9167$
7	1	$9167^2 \times 9726 = 4958$
6	1	$4958^2 \times 9726 = 7783$
5	0	$7783^2 = 6298$
4	0	$6298^2 = 4629$
3	1	$4629^2 \times 9726 = 10185$
2	1	$10185^2 \times 9726 = 105$
1	0	$105^2 = 11025$
0	1	$11025^2 \times 9726 = 5761$

1.5.2 Example: Rabin Decryption

The three algorithms described above are incorporated into many cryptographic encryption and decryption operations. We illustrate by considering the problem of decryption in the *Rabin cryptosystem*. Suppose that $p \equiv q \equiv 3 \pmod 4$, where $n = pq$ is the public modulus. The input will be a ciphertext, $y \in {\mathbb{Z}_n}^*$. The output should be the four possible plaintexts, i.e., the elements $x \in {\mathbb{Z}_n}^*$ such that $x^2 \equiv y \pmod n$. It is apparent that $x^2 \equiv y \pmod n$ if and only if $x^2 \equiv y \pmod p$ and $x^2 \equiv y \pmod q$. Because p and q are both congruent to 3 modulo 4, we can compute the two square roots of y modulo p, and the two square roots of y modulo q, using the formula derived in Section 1.5. Then we can use the Chinese remainder theorem to compute the four square roots of y modulo n.

Algorithm 1.8: RABINDECRYPT(n, p, q, y)

external CRT
if $(y^{(p-1)/2} \not\equiv 1 \pmod{p})$ **or** $(y^{(q-1)/2} \not\equiv 1 \pmod{q})$
  then return ("y is not a valid ciphertext")
  else $\begin{cases} x_p \leftarrow y^{(p+1)/4} \bmod p \\ x_q \leftarrow y^{(q+1)/4} \bmod q \\ x_1 \leftarrow \mathrm{CRT}(p, q, x_p, x_q) \\ x_2 \leftarrow \mathrm{CRT}(p, q, x_p, q - x_q) \\ x_3 \leftarrow \mathrm{CRT}(p, q, p - x_p, x_q) \\ x_4 \leftarrow \mathrm{CRT}(p, q, p - x_p, q - x_q) \\ \textbf{return } (\text{"the four plaintexts are } x_1, x_2, x_3 \text{ and } x_4\text{"}) \end{cases}$

As an initial step in Algorithm 1.8, we verity that y is a quadratic residue modulo p and modulo q. If so, then y is a valid ciphertext and the values x_1, x_2, x_3 and x_4 computed by the algorithm are, in fact, the four possible plaintexts. Notice that this algorithm performs several modular exponentiations, which would incorporate the SQUARE-AND-MULTIPLY algorithm. Also, the algorithm CRT itself computes modular inverses.

We return to Example 1.2 to illustrate the steps in Rabin decryption. In this example, we have $n = 7 \times 11 = 77$, and we are given the ciphertext $y = 23$ to decrypt. $23^3 \equiv 1 \pmod{7}$ and $23^5 \equiv 1 \pmod{11}$, so this confirms that y is a valid ciphertext. The square roots of 23 modulo 7 and 11 are

$$23^{(7+1)/4} \equiv 2^2 \equiv 4 \pmod{7}$$

and

$$23^{(11+1)/4} \equiv 1^3 \equiv 1 \pmod{11},$$

respectively. Hence, we have $x_p = 4$ and $x_q = 1$. Finally, using the Chinese remainder theorem, we compute the four square roots of 23 modulo 77 to be $\pm 10, \pm 32 \bmod 77$, as stated eariler.