Chapter 9

Mathematics of Cryptography

Part III: Primes and Related Congruence Equations

Chapter 9

Objectives

- To introduce prime numbers and their applications in cryptography.
- To discuss some primality test algorithms and their efficiencies.
- To discuss factorization algorithms and their applications in cryptography.
- To describe the Chinese remainder theorem and its application.
- To introduce quadratic congruence.
- To introduce modular exponentiation and logarithm.

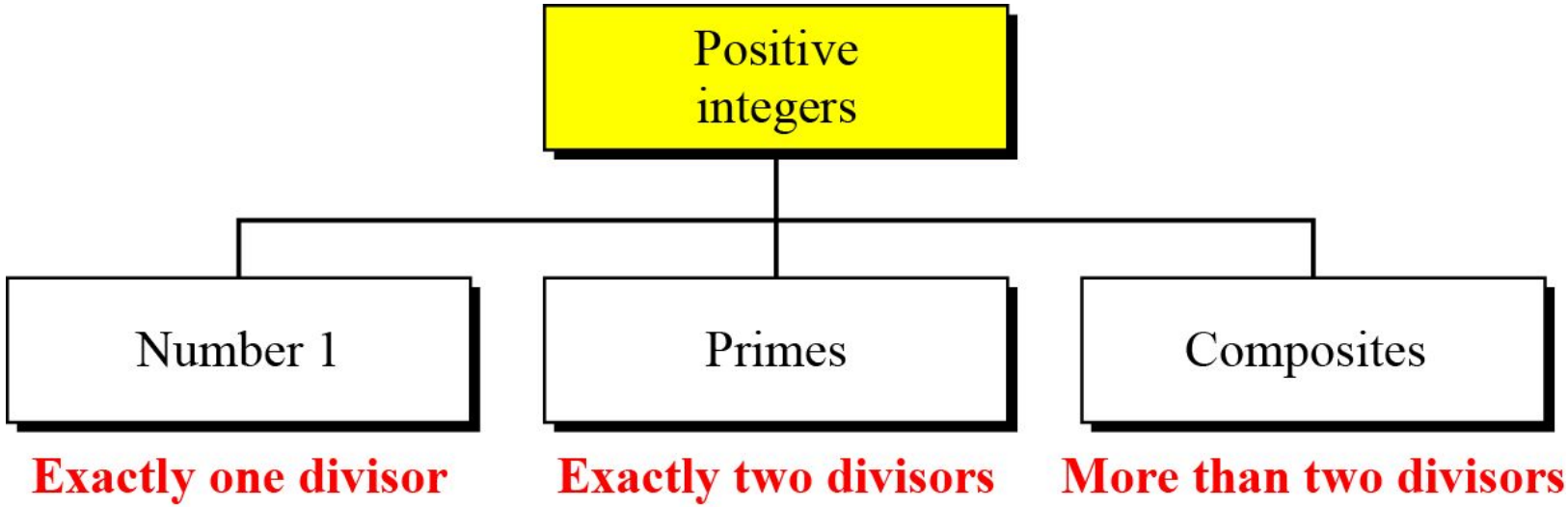
9-1 PRIMES

Asymmetric-key cryptography uses primes extensively. The topic of primes is a large part of any book on number theory. This section discusses only a few concepts and facts to pave the way for Chapter 10.

Topics discussed in this section:

9.1.1 Definition

Figure 9.1 *Three groups of positive integers*



Note

A prime is divisible only by itself and 1.

Example 9.1

What is the smallest prime?

Solution

The smallest prime is 2, which is divisible by 2 (itself) and 1.

Example 9.2

List the primes smaller than 10.

Solution

There are four primes less than 10: 2, 3, 5, and 7. It is interesting to note that the percentage of primes in the range 1 to 10 is 40%. The percentage decreases as the range increases.

9.1.2 Cardinality of Primes

Infinite Number of Primes

Note

There is an infinite number of primes.

Number of Primes

$$[n / (\ln n)] \quad < \quad \pi(n) \quad < \quad [n/(\ln n - 1.08366)]$$

Example 9.3

As a trivial example, assume that the only primes are in the set {2, 3, 5, 7, 11, 13, 17}. Here P = 510510 and P + 1 = 510511. However, 510511 = 19 × 97 × 277; none of these primes were in the original list. Therefore, there are three primes greater than 17.

Example 9.4

Find the number of primes less than 1,000,000.

Solution

The approximation gives the range 72,383 to 78,543. The actual number of primes is 78,498.

9.1.3 Checking for Primeness

Given a number n, how can we determine if n is a prime? The answer is that we need to see if the number is divisible by all primes less than

$$\sqrt{n}$$

We know that this method is inefficient, but it is a good start.

Theorem

If n is composite, then n has a prime divisor less than or equal to $\sqrt{n}$.

Proof.

- Let $n = ab$, $1 < a < n$, $1 < b < n$.
- We can't have both $a > \sqrt{n}$ and $b > \sqrt{n}$ since this would lead to $ab > n$.
- Therefore, n must have a prime divisor less than or equal to $\sqrt{n}$.

Example 9.5

Is 97 a prime?

Solution

The floor of $\sqrt{97} = 9$. The primes less than 9 are 2, 3, 5, and 7. We need to see if 97 is divisible by any of these numbers. It is not, so 97 is a prime.

Example 9.6

Is 301 a prime?

Solution

The floor of $\sqrt{301} = 17$. We need to check 2, 3, 5, 7, 11, 13, and 17. The numbers 2, 3, and 5 do not divide 301, but 7 does. Therefore 301 is not a prime.

Sieve of Eratosthenes

Table 9.1 *Sieve of Eratosthenes*

	2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~
11	~~12~~	13	~~14~~	~~15~~	~~16~~	17	~~18~~	19	~~20~~
~~21~~	~~22~~	23	~~24~~	~~25~~	~~26~~	~~27~~	~~28~~	29	~~30~~
31	~~32~~	~~33~~	~~34~~	~~35~~	~~36~~	37	~~38~~	~~39~~	~~40~~
41	~~42~~	43	~~44~~	~~45~~	~~46~~	47	~~48~~	~~49~~	~~50~~
~~51~~	~~52~~	53	~~54~~	~~55~~	~~56~~	~~57~~	~~58~~	59	~~60~~
61	~~62~~	~~63~~	~~64~~	~~65~~	~~66~~	67	~~68~~	~~69~~	~~70~~
71	~~72~~	73	~~74~~	~~75~~	~~76~~	~~77~~	~~78~~	79	~~80~~
~~81~~	~~82~~	83	~~84~~	~~85~~	~~86~~	~~87~~	~~88~~	89	~~90~~
~~91~~	~~92~~	~~93~~	~~94~~	~~95~~	~~96~~	97	~~98~~	~~99~~	~~100~~

9.1.4 Euler's Phi-Function

Euler's phi-function, φ (n), which is sometimes called the Euler's totient function plays a very important role in cryptography.

1. $\phi(1) = 0$.
2. $\phi(p) = p - 1$ if p is a prime.
3. $\phi(m \times n) = \phi(m) \times \phi(n)$ if m and n are relatively prime.
4. $\phi(p^e) = p^e - p^{e-1}$ if p is a prime.

9.1.4 Continued

We can combine the above four rules to find the value of φ(n). For example, if n can be factored as

$$n = p_1^{e_1} \times p_2^{e_2} \times \ldots \times p_k^{e_k}$$

then we combine the third and the fourth rule to find

$$\phi(n) = (p_1^{e_1} - p_1^{e_1 - 1}) \times (p_2^{e_2} - p_2^{e_2 - 1}) \times \cdots \times (p_k^{e_k} - p_k^{e_k - 1})$$

Note

The difficulty of finding φ(n) depends on the difficulty of finding the factorization of *n*.

Example 9.7

What is the value of φ(13)?

Solution

Because 13 is a prime, φ(13) = (13 −1) = 12.

Example 9.8

What is the value of φ(10)?

Solution

We can use the third rule: φ(10) = φ(2) × φ(5) = 1 × 4 = 4, because 2 and 5 are primes.

Example 9.9

What is the value of φ(240)?

Solution

We can write $240 = 2^4 \times 3^1 \times 5^1$. Then

$$\varphi(240) = (2^4 - 2^3) \times (3^1 - 3^0) \times (5^1 - 5^0) = 64$$

Example 9.10

Can we say that $\varphi(49) = \varphi(7) \times \varphi(7) = 6 \times 6 = 36$?

Solution

No. The third rule applies when *m* and *n* are relatively prime. Here $49 = 7^2$. We need to use the fourth rule: $\varphi(49) = 7^2 - 7^1 = 42$.

Example 9.11

What is the number of elements in Z_{14}^*?

Solution

The answer is $\varphi(14) = \varphi(7) \times \varphi(2) = 6 \times 1 = 6$. The members are 1, 3, 5, 9, 11, and 13.

Note

Interesting point: If $n > 2$, the value of $\varphi(n)$ is even.

9.1.5 Fermat's Little Theorem

First Version

$$a^{p-1} \equiv 1 \bmod p$$

Second Version

$$a^{p} \equiv a \bmod p$$

Example 9.12

Find the result of 6^{10} mod 11.

Solution

We have 6^{10} mod 11 = 1. This is the first version of Fermat's little theorem where $p = 11$.

Example 9.13

Find the result of 3^{12} mod 11.

Solution

Here the exponent (12) and the modulus (11) are not the same. With substitution this can be solved using Fermat's little theorem.

$$3^{12} \bmod 11 = (3^{11} \times 3) \bmod 11 = (3^{11} \bmod 11)\ (3 \bmod 11) = (3 \times 3) \bmod 11 = 9$$

Multiplicative Inverses

$$a^{-1} \bmod p = a^{p-2} \bmod p$$

Example 9.14

The answers to multiplicative inverses modulo a prime can be found without using the extended Euclidean algorithm:

a. $8^{-1} \bmod 17 = 8^{17-2} \bmod 17 = 8^{15} \bmod 17 = 15 \bmod 17$

b. $5^{-1} \bmod 23 = 5^{23-2} \bmod 23 = 5^{21} \bmod 23 = 14 \bmod 23$

c. $60^{-1} \bmod 101 = 60^{101-2} \bmod 101 = 60^{99} \bmod 101 = 32 \bmod 101$

d. $22^{-1} \bmod 211 = 22^{211-2} \bmod 211 = 22^{209} \bmod 211 = 48 \bmod 211$

9.1.6 Euler's Theorem

First Version

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

Second Version

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

Note

The second version of Euler's theorem is used in the RSA cryptosystem in Chapter 10.

Example 9.15

Find the result of 6^{24} mod 35.

Solution

We have $6^{24} \bmod 35 = 6^{\varphi(35)} \bmod 35 = 1$.

Example 9.16

Find the result of 20^{62} mod 77.

Solution

If we let $k = 1$ on the second version, we have

$$20^{62} \bmod 77 = (20 \bmod 77)\,(20^{\varphi(77)+1} \bmod 77) \bmod 77$$
$$= (20)(20) \bmod 77 = 15.$$

Multiplicative Inverses

Euler's theorem can be used to find multiplicative inverses modulo a composite.

$$a^{-1} \bmod n = a^{\varphi(n)-1} \bmod n$$

Example 9.17

The answers to multiplicative inverses modulo a composite can be found without using the extended Euclidean algorithm if we know the factorization of the composite:

a. $8^{-1} \bmod 77 = 8^{\phi(77)-1} \bmod 77 = 8^{59} \bmod 77 = 29 \bmod 77$

b. $7^{-1} \bmod 15 = 7^{\phi(15)-1} \bmod 15 = 7^{7} \bmod 15 = 13 \bmod 15$

c. $60^{-1} \bmod 187 = 60^{\phi(187)-1} \bmod 187 = 60^{159} \bmod 187 = 53 \bmod 187$

d. $71^{-1} \bmod 100 = 71^{\phi(100)-1} \bmod 100 = 71^{39} \bmod 100 = 31 \bmod 100$

9.1.7 Generating Primes

Mersenne Primes

$$M_p = 2^p - 1$$

$M_2 = 2^2 - 1 = 3$
$M_3 = 2^3 - 1 = 7$
$M_5 = 2^5 - 1 = 31$
$M_7 = 2^7 - 1 = 127$
$M_{11} = 2^{11} - 1 = 2047$ **Not a prime ($2047 = 23 \times 89$)**
$M_{13} = 2^{13} - 1 = 8191$
$M_{17} = 2^{17} - 1 = 131071$

Note

A number in the form $M_p = 2^p - 1$ is called a Mersenne number and may or may not be a prime.

9.1.7 Continued

Fermat Primes

$$F_n = 2^{2^n} + 1$$

$F_0 = 3$ $F_1 = 5$ $F_2 = 17$ $F_3 = 257$ $F_4 = 65537$

$F_5 = 4294967297 = 641 \times 6700417$ *Not a prime*

9-2 PRIMALITY TESTING

Finding an algorithm to correctly and efficiently test a very large integer and output a prime or a composite has always been a challenge in number theory, and consequently in cryptography. However, recent developments look very promising.

Topics discussed in this section:

Divisibility Algorithm

Algorithm 9.1 *Pseudocode for the divisibility test*

```
Divisibility_Test (n)          // n is the number to test for primality
{
  r ← 2
  while (r < √n )
  {
   if (r | n) return "a composite"
   r ← r + 1
  }
  return "a prime"
}
```

Note

The bit-operation complexity of the divisibility test is exponential.

Example 9.18

Assume n has 200 bits. What is the number of bit operations needed to run the divisibility-test algorithm?

Solution

The bit-operation complexity of this algorithm is $2^{n_b/2}$. This means that the algorithm needs 2^{100} bit operations. On a computer capable of doing 2^{30} bit operations per second, the algorithm needs 2^{70} seconds to do the testing (forever).

AKS Algorithm

$$O((\log_2 n_b)^{12})$$

Example 9.19

Assume *n* has 200 bits. What is the number of bit operations needed to run the AKS algorithm?

Solution

This algorithm needs only $(\log_2 200)^{12} = 39{,}547{,}615{,}483$ bit operations. On a computer capable of doing 1 billion bit operations per second, the algorithm needs only 40 seconds.

9.2.2 Probabilistic Algorithms

Fermat Test

If n is a prime, then $a^{n-1} \equiv 1 \bmod n$.

If n is a prime, $a^{n-1} \equiv 1 \bmod n$
If n is a composite, it is possible that $a^{n-1} \equiv 1 \bmod n$

Example 9.20

Does the number 561 pass the Fermat test?

Solution

Use base 2

The number passes the Fermat test, but it is not a prime, because $561 = 33 \times 17$.

Example 9.20

Does the number 561 pass the Fermat test?

Solution

Use base 2

$$2^{561-1} = 1 \bmod 561$$

The number passes the Fermat test, but it is not a prime, because $561 = 33 \times 17$.

Square Root Test

If n is a prime, $\sqrt{1} \bmod n = \pm 1$.
If n is a composite, $\sqrt{1} \bmod n = \pm 1$ and possibly other values.

Example 9.21

What are the square roots of 1 mod n if n is 7 (a prime)?

Solution

The only square roots are 1 and −1. We can see that

$$1^2 = 1 \bmod 7 \qquad (-1)^2 = 1 \bmod 7$$
$$2^2 = 4 \bmod 7 \qquad (-2)^2 = 4 \bmod 7$$
$$3^2 = 2 \bmod 7 \qquad (-3)^2 = 2 \bmod 7$$

Example 9.21

What are the square roots of 1 mod *n* if *n* is 7 (a prime)?

Solution

The only square roots are 1 and −1. We can see that

$$1^2 = 1 \bmod 7 \qquad (-1)^2 = 1 \bmod 7$$
$$2^2 = 4 \bmod 7 \qquad (-2)^2 = 4 \bmod 7$$
$$3^2 = 2 \bmod 7 \qquad (-3)^2 = 2 \bmod 7$$

Note that we don't have to test 4, 5 and 6 because 4 = –3 mod 7, 5 = –2 mod 7 and 6 = –1 mod 7.

Example 9.22

What are the square roots of 1 mod *n* if *n* is 8 (a composite)?

Solution

There are four solutions: 1, 3, 5, and 7 (which is −1). We can see that

$$1^2 = 1 \bmod 8 \qquad (-1)^2 = 1 \bmod 8$$
$$3^2 = 1 \bmod 8 \qquad 5^2 = 1 \bmod 8$$

Example 9.23

What are the square roots of 1 mod *n* if *n* is 17 (a prime)?

Solution

There are only two solutions: 1 and −1

$1^2 = 1 \bmod 17$	$(-1)^2 = 1 \bmod 17$
$2^2 = 4 \bmod 17$	$(-2)^2 = 4 \bmod 17$
$3^2 = 9 \bmod 17$	$(-3)^2 = 9 \bmod 17$
$4^2 = 16 \bmod 17$	$(-4)^2 = 16 \bmod 17$
$5^2 = 8 \bmod 17$	$(-5)^2 = 8 \bmod 17$
$6^2 = 2 \bmod 17$	$(-6)^2 = 2 \bmod 17$
$(7)^2 = 15 \bmod 17$	$(-7)^2 = 15 \bmod 17$
$(8)^2 = 13 \bmod 17$	$(-8)^2 = 13 \bmod 17$

Example 9.24

What are the square roots of 1 mod *n* if *n* is 22 (a composite)?

Solution

Surprisingly, there are only two solutions, +1 and −1, although 22 is a composite.

$$1^2 = 1 \bmod 22$$
$$(-1)^2 = 1 \bmod 22$$

Miller-Rabin Test

$$n - 1 = m \times 2^k$$

Figure 9.2 *Idea behind Fermat primality test*

$$a^{n-1} = a^{m \times 2^k} = [a^m]^{2^k} = [a^m]^{2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}} \quad (k \text{ times})$$

Note

The Miller-Rabin test needs from step 0 to step $k - 1$.

9.2.2 Continued

Algorithm 9.2 *Pseudocode for Miller-Rabin test*

```
Miller_Rabin_Test (n, a)                    // n is the number; a is the base.
{
    Find m and k such that n − 1 = m × 2^k
    T ← a^m mod n
    if (T = ± 1)  return "a prime"
    for (i ← 1 to k − 1)                    // k − 1 is the maximum number of steps.
    {
        T ← T^2 mod n
        if (T = +1) return "a composite"
        if (T = −1) return "a prime"
    }
    return "a composite"
}
```

Example 9.25

Does the number 561 pass the Miller-Rabin test?

Solution

Using base 2, let $561 - 1 = 35 \times 2^4$, which means $m = 35$, $k = 4$, and $a = 2$.

Initialization:	$T = 2^{35} \bmod 561 = 263 \bmod 561$	
$k = 1$:	$T = 263^2 \bmod 561 = 166 \bmod 561$	
$k = 2$:	$T = 166^2 \bmod 561 = 67 \bmod 561$	
$k = 3$:	$T = 67^2 \bmod 561 = +1 \bmod 561$	$\rightarrow$ **a composite**

Example 9.26

We already know that 27 is not a prime. Let us apply the Miller-Rabin test.

Solution

With base 2, let $27 - 1 = 13 \times 2^1$, which means that $m = 13$, $k = 1$, and $a = 2$. In this case, because $k - 1 = 0$, we should do only the initialization step: $T = 2^{13} \bmod 27 = 11 \bmod 27$. However, because the algorithm never enters the loop, it returns a composite.

Example 9.27

We know that 61 is a prime, let us see if it passes the Miller-Rabin test.

Solution

We use base 2.

$$61 - 1 = 15 \times 2^2 \rightarrow m = 15 \quad k = 2 \quad a = 2$$

Initialization: $T = 2^{15} \bmod 61 = 11 \bmod 61$

$k = 1$ $T = 11^2 \bmod 61 = -1 \bmod 61 \rightarrow$ **a prime**

9.2.3 Recommended Primality Test

Today, one of the most popular primality test is a combination of the divisibility test and the Miller-Rabin test.

Example 9.28

The number 4033 is a composite (37 × 109). Does it pass the recommended primality test?

Solution

1. **Perform the divisibility tests first. The numbers 2, 3, 5, 7, 11, 17, and 23 are not divisors of 4033.**
2. **Perform the Miller-Rabin test with a base of 2, 4033 − 1 = 63 × 26, which means *m* is 63 and *k* is 6.**

Initialization: $T \equiv 2^{63} \pmod{4033} \equiv 3521 \pmod{4033}$

$k = 1$ $\quad T \equiv T^2 \equiv 3521^2 \pmod{4033} \equiv -1 \pmod{4033}$ → **Passes**

Example 9.28 **Continued**

3. But we are not satisfied. We continue with another base, 3.

Initialization: $T \equiv 3^{63} \pmod{4033} \equiv 3551 \pmod{4033}$

$k = 1$ $\quad T \equiv T^2 \equiv 3551^2 \pmod{4033} \equiv 2443 \pmod{4033}$

$k = 2$ $\quad T \equiv T^2 \equiv 2443^2 \pmod{4033} \equiv 3442 \pmod{4033}$

$k = 3$ $\quad T \equiv T^2 \equiv 3442^2 \pmod{4033} \equiv 2443 \pmod{4033}$

$k = 4$ $\quad T \equiv T^2 \equiv 2443^2 \pmod{4033} \equiv 3442 \pmod{4033}$

$k = 5$ $\quad T \equiv T^2 \equiv 3442^2 \pmod{4033} \equiv 2443 \pmod{4033}$ **$\rightarrow$Failed (composite)**

9-3 FACTORIZATION

Factorization has been the subject of continuous research in the past; such research is likely to continue in the future. Factorization plays a very important role in the security of several public-key cryptosystems (see Chapter 10).

Topics discussed in this section:

9.3.1 Fundamental Theorem of Arithmetic

$$n = p_1^{e1} \times p_2^{e2} \times \cdots \times p_k^{ek}$$

Greatest Common Divisor

$$a = p_1^{a_1} \times p_2^{a_2} \times \cdots \times p_k^{a_k} \qquad b = p_1^{b_1} \times p_2^{b_2} \times \cdots \times p_k^{b_k}$$

$$\gcd(a, b) = p_1^{\min(a_1, b_1)} \times p_2^{\min(a_2, b_2)} \times \cdots \times p_k^{\min(a_k, b_k)}$$

Least Common Multiplier

$$a = p_1^{a1} \times p_2^{a2} \times \cdots \times p_k^{ak} \qquad b = p_1^{b1} \times p_2^{b2} \times \cdots \times p_k^{bk}$$

$$\text{lcm}(a, b) = p_1^{\max(a_1, b_1)} \times p_2^{\max(a_2, b_2)} \times \cdots \times p_k^{\max(a_k, b_k)}$$

$$\text{lcm}(a, b) \times \gcd(a, b) = a \times b$$

9.3.2 Factorization Methods

Trial Division Method

Algorithm 9.3 *Pseudocode for trial-division factorization*

```
Trial_Division_Factorization (n)          // n is the number to be factored
{
    a ← 2
    while (a ≤ √n )
    {
        while (n mod a = 0)
        {
            output a                      // Factors are output one by one
            n = n / a
        }
        a ← a + 1
    }
    if (n > 1) output n                   // n has no more factors
}
```

Example 9.29

Use the trial division algorithm to find the factors of 1233.

Solution

We run a program based on the algorithm and get the following result.

$$1233 = 3^2 \times 137$$

Example 9.30

Use the trial division algorithm to find the factors of 1523357784.

Solution

We run a program based on the algorithm and get the following result.

$$1523357784 = 2^3 \times 3^2 \times 13 \times 37 \times 43987$$

9.3.3 Fermat Method

$$n = x^2 - y^2 = a \times b \qquad \text{with a} = (\text{x} + \text{y}) \text{ and b} = (\text{x} - \text{y})$$

Algorithm 9.4 *Pseudocode for Fermat factorization*

```
Feramat_Factorization (n)                    // n is the number to be factored
{
    x ← √n                                   // smallest integer greater than √n
    while (x < n)
    {
    w ← x² − n
    if (w is perfect square)  y ← √w;  a ← x+y;  b ← x−y;   return a and b
    x ← x + 1
    }
}
```

9.3.4 Pollard p – 1 Method

$$p = \gcd\left(2^{B!} - 1, n\right)$$

Algorithm 9.5 *Pseudocode for Pollard p −1 factorization*

```
Pollard_ (p − 1) _Factorization (n, B)          // n is the number to be factored
{
    a ← 2
    e ← 2
    while (e ≤ B)
    {
        a ← a^e mod n
        e ← e + 1
    }
    p ← gcd (a −1, n)
    if 1 < p < n    return p
    return failure
}
```

Example 9.31

Use the Pollard $p - 1$ method to find a factor of 57247159 with the bound B = 8.

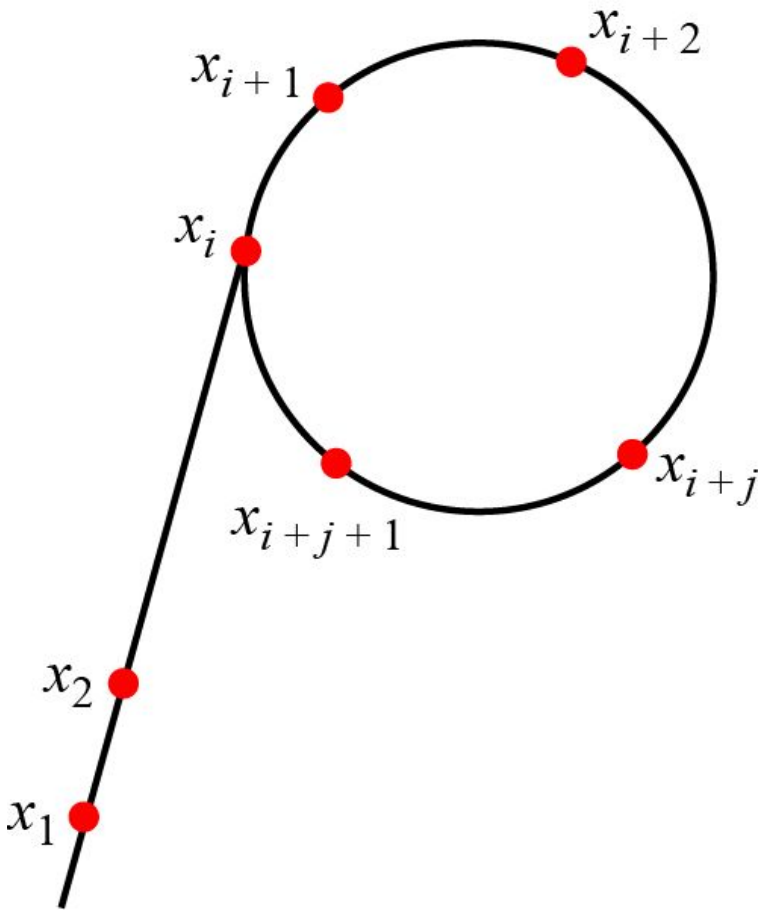
Solution

We run a program based on the algorithm and find that $p = 421$. As a matter of fact $57247159 = 421 \times 135979$. Note that 421 is a prime and $p - 1$ has no factor greater than 8

$$421 - 1 = 2^2 \times 3 \times 5 \times 7$$

9.3.5 Pollard rho Method

Figure 9.3 *Pollard rho successive numbers*



9.3.5 Continued

Algorithm 9.6 *Pseudocode for Pollard rho method*

```
Pollard_ rho _Factorization (n, B)          // n is the number to be factored
{
    x ← 2
    y ← 2
    p ← 1
    while (p = 1)
    {
        x ← f(x) mod n
        y ← f (f (y) mod n) mod n
        p ← gcd (x − y, n)
    }
    return p                                // if p = n, the program has failed
}
```

Example 9.32

Assume that there is a computer that can perform 2^{30} (almost 1 billion) bit operations per second. What is the approximation time required to factor an integer of size
a. 60 decimal digits? b. 100 decimal digits?

Solution

a. A number of 60 decimal digits has almost 200 bits. The complexity is then or 2^{50}. With 2^{30} operations per second, the algorithm can be computed in 2^{20} seconds, or almost 12 days.

b. A number of 100 decimal digits has almost 300 bits. The complexity is 2^{75}. With 2^{30} operations per second, the algorithm can be computed in 2^{45} seconds, many years.

Example 9.33

We have written a program to calculate the factors of 434617. The result is 709 (434617 = 709 × 613).

Table 9.2 *Values of x, y, and p in Example 9.33*

x	y	p
2	2	1
5	26	1
26	23713	1
677	142292	1
23713	157099	1
346589	52128	1
142292	41831	1
380320	68775	1
157099	427553	1
369457	2634	1
52128	63593	1
102901	161353	1
41831	64890	1
64520	21979	1
68775	16309	709

9.3.6 More Efficient Methods

Quadratic Sieve

The method uses a sieving procedure to find the value of x^2 mod n.

$$O(e^C), \text{ where } C \approx (\ln n \ln\ln n)^{1/2}$$

Number Field Sieve

The method uses a sieving procedure in an algebraic ring structure to find $x^2 \equiv y^2$ mod n.

$$O(e^C) \text{ where } C \approx 2 (\ln n)^{1/3} (\ln\ln n)^{2/3}$$

Example 9.34

Assume that there is a computer that can perform 230 (almost 1 billion) bit operations per second. What is the approximate time required for this computer to factor an integer of 100 decimal digits using one of the following methods?

a. Quadratic sieve method b. Number field sieve method

Solution

A number with 100 decimal digits has almost 300 bits ($n = 2^{300}$). $\ln(2^{300}) = 207$ and $\ln\ln(2^{300}) = 5$.

a. $(207)^{1/2} \times (5)^{1/2} = 14 \times 2.23 \approx 32$ e^{32} $(e^{32}) / (2^{30}) \approx 20$ hours.

b. $(207)^{1/3} \times (5)^{2/2} = 6 \times 3 \approx 18$. e^{18} $(e^{18}) / (2^{30}) \approx 6$ seconds.

9-4 CHINESE REMAINDER THEOREM

The Chinese remainder theorem (CRT) is used to solve a set of congruent equations with one variable but different moduli, which are relatively prime, as shown below:

$$x \equiv a_1 \pmod{m_1}$$
$$x \equiv a_2 \pmod{m_2}$$
$$\cdots$$
$$x \equiv a_k \pmod{m_k}$$

9-4 Continued

Example 9.35

The following is an example of a set of equations with different moduli:

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

The solution to this set of equations is given in the next section; for the moment, note that the answer to this set of equations is x = 23. This value satisfies all equations: $23 \equiv 2 \pmod 3$, $23 \equiv 3 \pmod 5$, and $23 \equiv 2 \pmod 7$.

9-4 Continued

Solution To Chinese Remainder Theorem

1. Find $M = m_1 \times m_2 \times \ldots \times m_k$. This is the common modulus.
2. Find $M_1 = M/m_1$, $M_2 = M/m_2$, …, $M_k = M/m_k$.
3. Find the multiplicative inverse of M_1, M_2, …, M_k using the corresponding moduli (m_1, m_2, …, m_k). Call the inverses M_1^{-1}, M_2^{-1}, …, M_k^{-1}.
4. The solution to the simultaneous equations is

$$x = (a_1 \times M_1 \times M_1^{-1} + a_2 \times M_2 \times M_2^{-1} + \cdots + a_k \times M_k \times M_k^{-1}) \bmod M$$

9-4 Continued

Example 9.36

Find the solution to the simultaneous equations:

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

Solution

We follow the four steps.

1. $M = 3 \times 5 \times 7 = 105$

2. $M_1 = 105 / 3 = 35$, $M_2 = 105 / 5 = 21$, $M_3 = 105 / 7 = 15$

3. The inverses are $M_1^{-1} = 2$, $M_2^{-1} = 1$, $M_3^{-1} = 1$

4. $x = (2 \times 35 \times 2 + 3 \times 21 \times 1 + 2 \times 15 \times 1) \bmod 105 = 23 \bmod 105$

9-4 Continued

Example 9.37

Find an integer that has a remainder of 3 when divided by 7 and 13, but is divisible by 12.

Solution

This is a CRT problem. We can form three equations and solve them to find the value of x.

$$x = 3 \bmod 7$$
$$x = 3 \bmod 13$$
$$x = 0 \bmod 12$$

If we follow the four steps, we find x = 276. We can check that 276 = 3 mod 7, 276 = 3 mod 13 and 276 is divisible by 12 (the quotient is 23 and the remainder is zero).

9-4 Continued

Example 9.38

Assume we need to calculate $z = x + y$ where $x = 123$ and $y = 334$, but our system accepts only numbers less than 100.

$$\begin{array}{ll} x \equiv 24 \pmod{99} & y \equiv 37 \pmod{99} \\ x \equiv 25 \pmod{98} & y \equiv 40 \pmod{98} \\ x \equiv 26 \pmod{97} & y \equiv 43 \pmod{97} \end{array}$$

Adding each congruence in x with the corresponding congruence in y gives

$$\begin{array}{lcl} x + y \equiv 61 \pmod{99} & \rightarrow & z \equiv 61 \pmod{99} \\ x + y \equiv 65 \pmod{98} & \rightarrow & z \equiv 65 \pmod{98} \\ x + y \equiv 69 \pmod{97} & \rightarrow & z \equiv 69 \pmod{97} \end{array}$$

Now three equations can be solved using the Chinese remainder theorem to find z. One of the acceptable answers is $z = 457$.

9-5 QUADRATIC CONGRUENCE

In cryptography, we also need to discuss quadratic congruence—that is, equations of the form $a_2x^2 + a_1x + a_0 \equiv 0 \pmod{n}$. *We limit our discussion to quadratic equations in which* $a_2 = 1$ *and* $a_1 = 0$, *that is equations of the form*

$$x^2 \equiv a \pmod{n}.$$

Topics discussed in this section:

9.5.1 Quadratic Congruence Modulo a Prime

9.5.2 Quadratic Congruence Modulo a Composite

9.5.1 Quadratic Congruence Modulo a Prime

We first consider the case in which the modulus is a prime.

Example 9.39

The equation $x^2 \equiv 3 \pmod{11}$ has two solutions, $x \equiv 5 \pmod{11}$ and $x \equiv -5 \pmod{11}$. But note that $-5 \equiv 6 \pmod{11}$, so the solutions are actually 5 and 6. Also note that these two solutions are incongruent.

Example 9.40

The equation $x^2 \equiv 2 \pmod{11}$ has no solution. No integer x can be found such that its square is 2 mod 11.

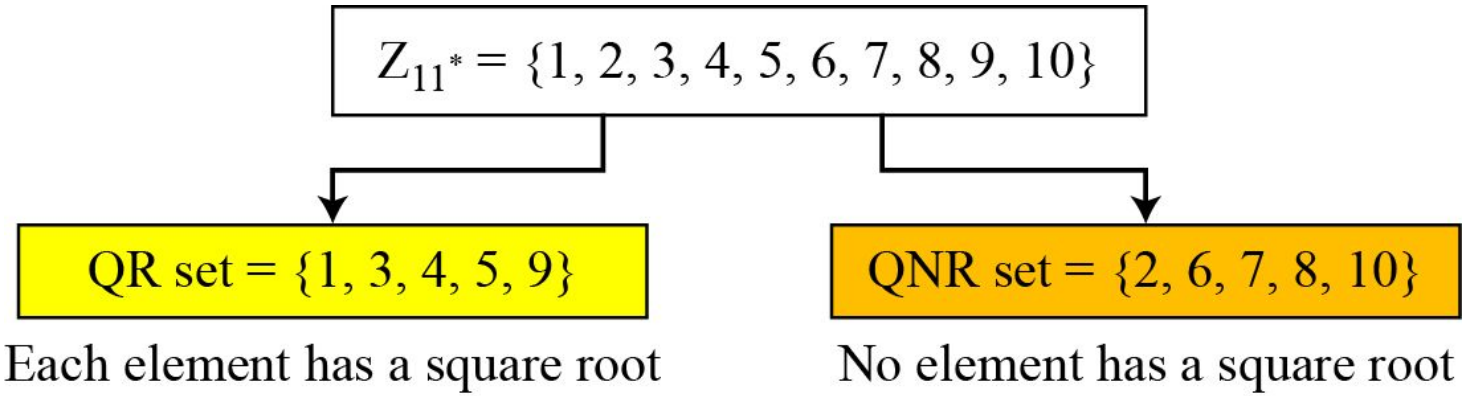
Quadratic Residues and Nonresidue

In the equation $x^2 \equiv a \pmod{p}$, a is called a quadratic residue (QR) if the equation has two solutions; a is called quadratic nonresidue (QNR) if the equation has no solutions.

Example 9.41

There are 10 elements in Z_{11}^*. Exactly five of them are quadratic residues and five of them are nonresidues. In other words, Z_{11}^* is divided into two separate sets, QR and QNR, as shown in Figure 9.4.

Figure 9.4 *Division of Z_{11}^* elements into QRs and QNRs*



Euler's Criterion

a. *If* $a^{(p-1)/2} \equiv 1 \pmod{p}$, *a is a quadratic residue modulo p.*

b. *If* $a^{(p-1)/2} \equiv -1 \pmod{p}$, *a is a quadratic nonresidue modulo p.*

Example 9.42

To find out if 14 or 16 is a QR in Z_{23}^*, we calculate:

$$14^{(23-1)/2} \bmod 23 \rightarrow 22 \bmod 23 \rightarrow -1 \bmod 23 \text{ nonresidue}$$

$$16^{(23-1)/2} \bmod 23 \rightarrow 16^{11} \bmod 23 \rightarrow 1 \bmod 23 \text{ residue}$$

Solving Quadratic Equation Modulo a Prime

Special Case: $p = 4k + 3$

$$x \equiv a^{(p+1)/4} \pmod{p} \qquad \text{and} \qquad x \equiv -a^{(p+1)/4} \pmod{p}$$

Example 9.43

Solve the following quadratic equations:

a. $x^2 \equiv 3 \pmod{23}$

b. $x^2 \equiv 2 \pmod{11}$

c. $x^2 \equiv 7 \pmod{19}$

Solutions

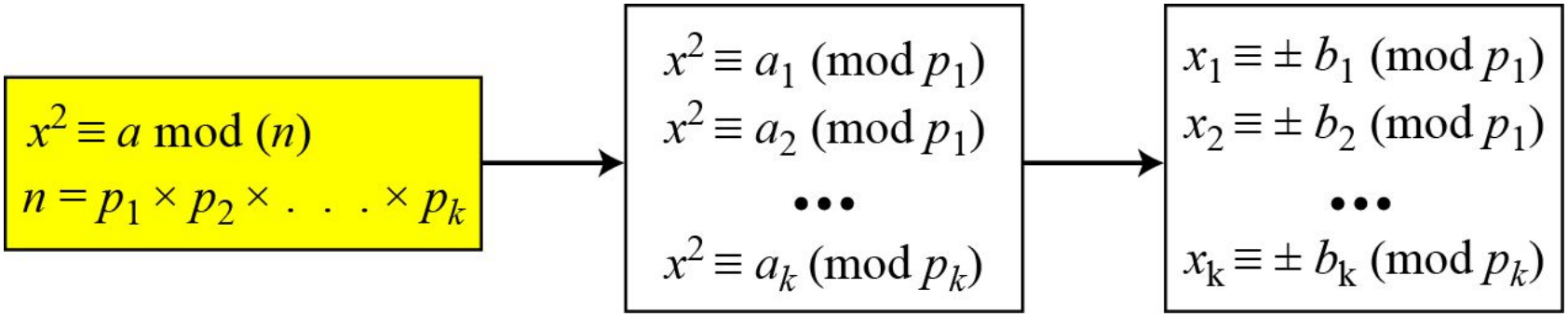
a. $x \equiv \pm 16 \pmod{23}$ $\sqrt{3} \equiv \pm 16 \pmod{23}$.

b. There is no solution for $\sqrt{2}$ in Z11.

c. $x \equiv \pm 11 \pmod{19}$. $\sqrt{7} \equiv \pm 11 \pmod{19}$.

9.5.2 Quadratic Congruence Modulo a Composite

Figure 9.5 *Decomposition of congruence modulo a composite*



Example 9.44

Assume that $x^2 \equiv 36 \pmod{77}$. We know that $77 = 7 \times 11$. We can write

$$x^2 \equiv 36 \pmod 7 \equiv 1 \pmod 7 \qquad \text{and} \qquad x^2 \equiv 36 \pmod{11} \equiv 3 \pmod{11}$$

The answers are $x \equiv +1 \pmod 7$, $x \equiv -1 \pmod 7$, $x \equiv +5 \pmod{11}$, and $x \equiv -5 \pmod{11}$. Now we can make four sets of equations out of these:

Set 1: $x \equiv +1 \pmod 7$ $\quad x \equiv +5 \pmod{11}$
Set 2: $x \equiv +1 \pmod 7$ $\quad x \equiv -5 \pmod{11}$
Set 3: $x \equiv -1 \pmod 7$ $\quad x \equiv +5 \pmod{11}$
Set 4: $x \equiv -1 \pmod 7$ $\quad x \equiv -5 \pmod{11}$

The answers are $x = \pm 6$ and ± 27.

Complexity

How hard is it to solve a quadratic congruence modulo a composite? The main task is the factorization of the modulus. In other words, the complexity of solving a quadratic congruence modulo a composite is the same as factorizing a composite integer. As we have seen, if *n* is very large, factorization is infeasible.

Note

Solving a quadratic congruence modulo a composite is as hard as factorization of the modulus.

9-6 EXPONENTIATION AND LOGARITHM

Exponentiation: $y = a^x$ $\rightarrow$ **Logarithm:** $x = \log_a y$

Topics discussed in this section:

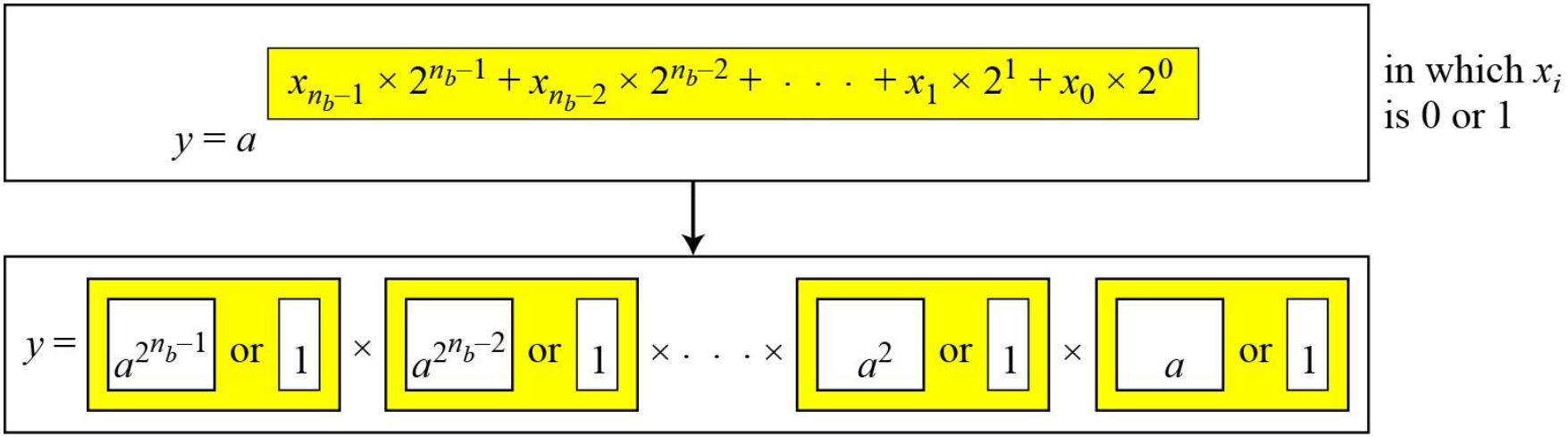
9.6.1 Exponentiation

9.6.2 Logarithm

9.6.1 Exponentiation

Fast Exponentiation

Figure 9.6 *The idea behind the square-and-multiply method*



Example:

$$y = a^9 = a^{1001_2} = a^8 \times 1 \times 1 \times a$$

9.6.1 Continued

Algorithm 9.7 *Pseudocode for square-and-multiply algorithm*

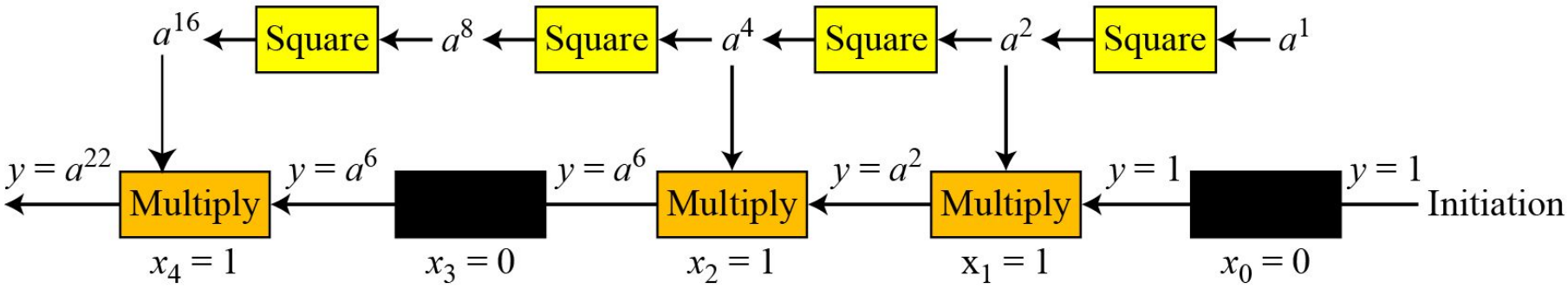
```
Square_and_Multiply (a, x, n)
{
    y ← 1
    for (i ← 0 to n_b − 1)                  // n_b is the number of bits in x
    {
        if (x_i = 1)   y ← a × y mod n      // multiply only if the bit is 1

        a ← a^2 mod n                       // squaring is not needed in the last iteration
    }
    return y
}
```

Example 9.45

Figure 9.7 shows the process for calculating $y = a^x$ using the Algorithm 9.7 (for simplicity, the modulus is not shown). In this case, $x = 22 = (10110)2$ in binary. The exponent has five bits.

Figure 9.7 *Demonstration of calculation of a^{22} using square-and-multiply method*



9.6.1 Continued

Table 9.3 *Calculation of 17^{22} mod 21*

i	x_i	*Multiplication (Initialization: $y = 1$)*		*Squaring (Initialization: $a = 17$)*
0	0		$\rightarrow$	$a = 17^2 \bmod 21 = 16$
1	1	$y = 1 \times 16 \bmod 21 = 16$	$\rightarrow$	$a = 16^2 \bmod 21 = 4$
2	1	$y = 16 \times 4 \bmod 21 = 1$	$\rightarrow$	$a = 4^2 \bmod 21 = 16$
3	0		$\rightarrow$	$a = 16^2 \bmod 21 = 4$
4	1	$y = 1 \times 4 \bmod 21 = 4$	$\rightarrow$	

How about 21^{24} mod 8?

9.6.2 Logarithm

In cryptography, we also need to discuss modular logarithm.

Exhaustive Search

Algorithm 9.8 *Exhaustive search for modular logarithm*

```
Modular_Logarithm (a, y, n)
{
    for (x = 1 to n −1)                          // k is the number of bits in x
    {
        if (y ≡ a^x mod n) return x
    }
    return failure
}
```

Table 8.3 Powers of Integers, Modulo 19

a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^{10}	a^{11}	a^{12}	a^{13}	a^{14}	a^{15}	a^{16}	a^{17}	a^{18}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	4	8	16	13	7	14	9	18	17	15	11	3	6	12	5	10	1
3	9	8	5	15	7	2	6	18	16	10	11	14	4	12	17	13	1
4	16	7	9	17	11	6	5	1	4	16	7	9	17	11	6	5	1
5	6	11	17	9	7	16	4	1	5	6	11	17	9	7	16	4	1
6	17	7	4	5	11	9	16	1	6	17	7	4	5	11	9	16	1
7	11	1	7	11	1	7	11	1	7	11	1	7	11	1	7	11	1
8	7	18	11	12	1	8	7	18	11	12	1	8	7	18	11	12	1
9	5	7	6	16	11	4	17	1	9	5	7	6	16	11	4	17	1
10	5	12	6	3	11	15	17	18	9	14	7	13	16	8	4	2	1
11	7	1	11	7	1	11	7	1	11	7	1	11	7	1	11	7	1
12	11	18	7	8	1	12	11	18	7	8	1	12	11	18	7	8	1
13	17	12	4	14	11	10	16	18	6	2	7	15	5	8	9	3	1
14	6	8	17	10	7	3	4	18	5	13	11	2	9	12	16	15	1
15	16	12	9	2	11	13	5	18	4	3	7	10	17	8	6	14	1
16	9	11	5	4	7	17	6	1	16	9	11	5	4	7	17	6	1
17	4	11	16	6	7	5	9	1	17	4	11	16	6	7	5	9	1
18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1

Order of the Group.

Example 9.46

What is the order of group G = $\langle Z_{21}^*, \times \rangle$? $|G| = \varphi(21) = \varphi(3) \times \varphi(7) = 2 \times 6 = 12$. There are 12 elements in this group: 1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, and 20. All are relatively prime with 21.

Order of an Element

Example 9.47

Find the order of all elements in $G = \langle Z_{10}^*, \times \rangle$.

Solution

This group has only $\varphi(10) = 4$ elements: 1, 3, 7, 9. We can find the order of each element by trial and error.

a. $1^1 \equiv 1 \bmod (10) \rightarrow \text{ord}(1) = 1$.
b. $3^4 \equiv 1 \bmod (10) \rightarrow \text{ord}(3) = 4$.
c. $7^4 \equiv 1 \bmod (10) \rightarrow \text{ord}(7) = 4$.
d. $9^2 \equiv 1 \bmod (10) \rightarrow \text{ord}(9) = 2$.

Euler's Theorem

Example 9.48

Table 9.4 *Finding the orders of elements in Example 9.48*

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$
$a = 1$	**x: 1**	x: 1	x: 1	x: 1	x: 1	x: 1	x: 1
$a = 3$	x: 3	**x: 1**	x: 3	x: 1	x: 3	x: 1	x: 3
$a = 5$	x: 5	**x: 1**	x: 5	x: 1	x: 5	x: 1	x: 5
$a = 7$	x: 7	**x: 1**	x: 7	x: 1	x: 7	x: 1	x: 7

9.6.2 Continued

Primitive Roots *In the group* $G = \langle Z_n^*, \times \rangle$, *when the order of an element is the same as* $\varphi(n)$, *that element is called the primitive root of the group.*

Example 9.49

Table 9.4 shows that there are no primitive roots in $G = \langle Z_8^*, \times \rangle$ **because no element has the order equal to** $\varphi(8) = 4$. **The order of elements are all smaller than 4.**

Example 9.50

Table 9.5 shows the result of $a^i \equiv x \pmod 7$ for the group $G = \langle Z_7^*, \times \rangle$. In this group, $\varphi(7) = 6$.

Table 9.5 *Example 9.50*

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
$a = 1$	x: 1	x: 1	x: 1	x: 1	x: 1	x: 1
$a = 2$	x: 2	x: 4	x: 1	x: 2	x: 4	x: 1
Primitive root → $a = 3$	x: 3	x: 2	x: 6	x: 4	x: 5	x: 1
$a = 4$	x: 4	x: 2	x: 1	x: 4	x: 2	x: 1
Primitive root → $a = 5$	x: 5	x: 4	x: 6	x: 2	x: 3	x: 1
$a = 6$	x: 6	x: 1	x: 6	x: 1	x: 6	x: 1

Note

The group $G = \langle Z_n^*, \times \rangle$ has primitive roots only if n is 2, 4, p^t, or $2p^t$.

Example 9.51

For which value of n, does the group $G = \langle Z_n^*, \times \rangle$ have primitive roots: 17, 20, 38, and 50?

Solution

a. $G = \langle Z_{17}^*, \times \rangle$ has primitive roots, 17 is a prime.

b. $G = \langle Z_{20}^*, \times \rangle$ has no primitive roots.

c. $G = \langle Z_{38}^*, \times \rangle$ has primitive roots, $38 = 2 \times 19$ prime.

d. $G = \langle Z_{50}^*, \times \rangle$ has primitive roots, $50 = 2 \times 5^2$ and 5 is a prime.

Note

If the group $G = \langle Z_n^*, \times \rangle$ has any primitive root, the number of primitive roots is $\varphi(\varphi(n))$.

Cyclic Group *If g is a primitive root in the group, we can generate the set Z_n^* as $Z_n^* = \{g^1, g^2, g^3, \ldots, g^{\varphi(n)}\}$*

Example 9.52

The group $G = \langle Z_{10}^*, \times \rangle$ has two primitive roots because $\varphi(10) = 4$ and $\varphi(\varphi(10)) = 2$. It can be found that the primitive roots are 3 and 7. The following shows how we can create the whole set Z_{10}^* using each primitive root.

$g = 3$	$\rightarrow$	$g^1 \bmod 10 = 3$	$g^2 \bmod 10 = 9$	$g^3 \bmod 10 = 7$	$g^4 \bmod 10 = 1$
$g = 7$	$\rightarrow$	$g^1 \bmod 10 = 7$	$g^2 \bmod 10 = 9$	$g^3 \bmod 10 = 3$	$g^4 \bmod 10 = 1$

The group $G = \langle Z_n^*, \times \rangle$ is a cyclic group if it has primitive roots.
The group $G = \langle Z_p^*, \times \rangle$ is always cyclic.

The idea of Discrete Logarithm

Properties of $G = \langle Z_p^*, \times \rangle$:

1. **Its elements include all integers from 1 to $p - 1$.**

2. **It always has primitive roots.**

3. **It is cyclic. The elements can be created using g^x where x is an integer from 1 to $\varphi(n) = p - 1$.**

4. **The primitive roots can be thought as the base of logarithm.**

Solution to Modular Logarithm Using Discrete Logs

Tabulation of Discrete Logarithms

Table 9.6 *Discrete logarithm for* $\mathbf{G} = \langle \mathbf{Z}_7^*, \times \rangle$

y	1	2	3	4	5	6
$x = \mathrm{L}_3\, y$	6	2	1	4	5	3
$x = \mathrm{L}_5\, y$	6	4	5	2	1	3

Example 9.53

Find x in each of the following cases:

a. $4 \equiv 3^x \pmod 7$.

b. $6 \equiv 5^x \pmod 7$.

Solution

We can easily use the tabulation of the discrete logarithm in Table 9.6.

a. $4 \equiv 3^x \bmod 7 \rightarrow x = L_3 4 \bmod 7 = 4 \bmod 7$

b. $6 \equiv 5^x \bmod 7 \rightarrow x = L_5 6 \bmod 7 = 3 \bmod 7$

Using Properties of Discrete Logarithms

Table 9.7 *Comparison of traditional and discrete logarithms*

Traditional Logarithm	*Discrete Logarithms*
$\log_a 1 = 0$	$L_g 1 \equiv 0 \pmod{\phi(n)}$
$\log_a (x \times y) = \log_a x + \log_a y$	$L_g(x \times y) \equiv (L_g x + L_g y) \pmod{\phi(n)}$
$\log_a x^k = k \times \log_a x$	$L_g x^k \equiv k \times L_g x \pmod{\phi(n)}$

Using Algorithms Based on Discrete

Note

The discrete logarithm problem has the same complexity as the factorization problem.