The receiver recovers the original message by computing

$$128^{23} \equiv 2 \pmod{187}. \qquad \blacktriangle$$

In Example 25.2 some of the computations would be long and tedious without the use of a computer. For large primes p and q, it is essential to have an efficient algorithm to compute $m^s \pmod{n}$ and $y^r \pmod{n}$. This can be accomplished by using base 2. We illustrate with the following example.

25.3 Example In Example 25.2 we needed to compute $128^{23} \pmod{187}$. We can compute this value by expressing 23 in base 2, $23 = 16 + 4 + 2 + 1$, and then computing the following:

$$\begin{aligned}
128^1 &= 128 \\
128^2 &= 1638 \equiv 115 \pmod{187} \\
128^4 &= (128^2)^2 \equiv 115^2 \equiv 135 \pmod{187} \\
128^8 &= (128^4)^2 \equiv 135^2 \equiv 86 \pmod{187} \\
128^{16} &= (128^8)^2 \equiv 86^2 \equiv 103 \pmod{187},
\end{aligned}$$

Thus

$$\begin{aligned}
128^{23} &\equiv 128^{16+4+2+1} \\
&\equiv (128^{16}128^4)(128^2 128^1) \\
&\equiv (103 \cdot 135)(115 \cdot 128) \\
&\equiv 67 \cdot 134 \\
&\equiv 2 \pmod{187}.
\end{aligned} \qquad \blacktriangle$$

As illustrated in the above example, this method gives a more efficient computation of $a^k \pmod{n}$.

The Euclidean algorithm is a simple and efficient way to compute the inverse of a unit in $\mathbb{Z}_{(p-1)(q-1)}$. It involves the repeated use of the division algorithm. However, we will not discuss the Euclidean algorithm here.

The reader may have noticed a potential flaw in the RSA encryption scheme. It is possible that m is a multiple of either p or q. In that case, $m^{(p-1)(q-1)} \not\equiv 1 \pmod{n}$, which means that m^{rs} may not be equivalent to m modulo n. In this case RSA encryption fails. However, when using large prime numbers the probability that the message is a multiple of p or q is extremely low. If one is concerned about this issue, the algorithm could be modified slightly to be sure that the message is smaller than both p and q.

How are the large prime numbers p and q in RSA encryption found? Basically, the process is to guess a value and check that it is prime. Unfortunately, there is no known fast method to test for primality, but it is possible to do a fast probabilistic test. One simple probabilistic test uses Fermat's Theorem (Theorem 24.1). The idea is to generate a random positive integer less than p and check if $a^{p-1} \equiv 1 \pmod{p}$. If p is prime, then $a^{p-1} \equiv 1 \pmod{p}$, so if $a^{p-1} \not\equiv 1 \pmod{p}$, then p is not a prime number and the number p is rejected. On the other hand, if $a^{p-1} \equiv 1 \pmod{p}$, then p passes the test and p could be a prime. If p passes the test, we repeat the process for a different random value of a. The probability that a composite number p is picked given that p passes the test several times is low enough to safely assume that p is prime.

EXERCISES 25

In Exercises 1 through 8, the notation is consistent with the notation used in the text for RSA encryption. It may be helpful to use a calculator or computer.

1. Let $p = 3$ and $q = 5$. Find n, and all possible pairs (r, s).

2. Let $p = 3$ and $q = 7$. Find n and all possible pairs (r, s).

3. Let $p = 3$ and $q = 11$. Find n and all possible pairs (r, s).

4. Let $p = 5$ and $q = 7$. Find n and all possible pairs (r, s).

5. Let $p = 13$, $q = 17$, and $r = 5$. Find the value of s.

6. For RSA encryption it is assumed that the message m is at least 2. Why should m not be 1?

7. The public key is $n = 143$ and $s = 37$.

- **a.** Compute the value of y if the message is $m = 25$.
- **b.** Find r. (Computer Algebra Systems have built-in functions to compute in $\mathbb{Z}_m$.)
- **c.** Use your answers to parts a) and b) to decrypt y.

8. The public key is $n = 1457$ and $s = 239$.

- **a.** Compute the value of y if the message is $m = 999$.
- **b.** Find r. (Computer Algebra Systems have built-in functions to compute in $\mathbb{Z}_m$.)
- **c.** Use your answers to parts a) and b) to decrypt y.

9. For $p = 257$, $q = 359$, and $r = 1493$ identify the private and public keys.

This page is intentionally left blank

PART VI Constructing Rings and Fields

SECTION 26 The Field of Quotients of an Integral Domain

Let L be a field and D a subring of L that contains the unity. The ring D is an integral domain since it has no zero divisors. Also F, the set of all quotients of the form $\frac{a}{b}$ with a and $b \neq 0$ both in D, forms a subfield of L. The field F is called a *field of quotients of the integral domain D*.

26.1 Example Let $L = \mathbb{R}$. If $D = \mathbb{Z}$, then

$$F = \left\{ \frac{a}{b} \,\middle|\, a, b \in \mathbb{Z}, b \neq 0 \right\} = \mathbb{Q}$$

which is a field.

If $D = \{x + y\sqrt{2} \mid x, y \in \mathbb{Z}\}$, then

$$F = \left\{ \frac{a}{b} \,\middle|\, a, b \in D, b \neq 0 \right\} = \left\{ \frac{x + y\sqrt{2}}{z + w\sqrt{2}} \,\middle|\, x, y, z, w \in \mathbb{Z}, z + w\sqrt{2} \neq 0 \right\}.$$

By rationalizing the denominator we see that

$$F = \left\{ r + s\sqrt{2} \mid r, s \in \mathbb{Q} \right\}$$

which is a field by Exercise 12 in Section 22. ▲

In this section, we start with an integral domain D and construct a field F. We then show that D is isomorphic with a subring D' of F and that F consists of all quotients $\frac{a}{b}$ with $a, b \in D'$, $b \neq 0$. Thus we can think of any integral domain as being a subring of a field and every element of the field is the quotient of elements from the integral domain.

The Construction

Let D be an integral domain that we desire to enlarge to a field of quotients F. A coarse outline of the steps we take is as follows:

1. Define what the elements of F are to be.
2. Define the binary operations of addition and multiplication on F.