Section 10.5.1 Congruences

In short, a *congruence relation* is an equivalence relation on the carrier of an algebra such that the operations of the algebra are preserved by the relation.

Example. For the algebra $\langle \mathbf{Z}: +, \cdot \rangle$, let $x \sim y$ mean $x \bmod 4 = y \bmod 4$. Notice that $\sim$ is an equivalence relation on $\mathbf{Z}$. The four equivalence classes are

$$[0] = \{4k \mid k \in \mathbf{Z}\},\ [1] = \{4k + 1 \mid k \in \mathbf{Z}\},\ [2] = \{4k + 2 \mid k \in \mathbf{Z}\},\ [3] = \{4k + 3 \mid k \in \mathbf{Z}\}.$$

Notice also that addition and multiplication are preserved by $\sim$. In other words, if $a \sim b$ and $c \sim d$, then we also have

$$a + c \sim b + d \quad \text{and} \quad a \cdot c \sim b \cdot d$$

Can you verify these facts? So $\sim$ is a congruence relation on the algebra $\langle \mathbf{Z}: +, \cdot \rangle$.

Notation: The relation $x \bmod n = y \bmod n$ is also written as $x \equiv y \pmod{n}$ and we say, " x is congruent to y mod n."

Application I

Let $[0], [1], \ldots, [n - 1]$ be the equivalence classes for the relation $x \equiv y \pmod{n}$ on $\mathbf{Z}$. Then they form the elements of an algebra with operations + and · defined by by

$$[a] + [b] = [a + b] \text{ and } [a] \cdot [b] = [a \cdot b]$$

For example, with $x \equiv y \pmod 4$, we have the following calculations:

$$[2] + [3] = [2 + 3] = [5] = [1] \quad \text{and} \quad [2] \cdot [3] = [2 \cdot 3] = [6] = [2].$$

These facts are used to give a short proof of:

Fermat's little theorem: If p is prime and p does not divide a, then $a^{p-1} \equiv 1 \pmod{p}$.

Application II
Fermat's little theorem is used to prove the RSA theorem in cryptology (upcoming).

Application III
Solving a congruence. If gcd(a, n) = 1, then $ax \equiv b \pmod{n}$ has a solution.

Algorithm:
(1) Find integers s and t such that $1 = as + nt$. (e.g., use Euclidean algorithm in reverse).
(2) Then $x = bs$ solves the congruence. The complete set of solutions is $\{bs + nk \mid k \in \mathbf{Z}\}$.

Example. Solve $10x \equiv 5 \pmod{27}$
Solution. Use the Euclidean algorithm to find the gcd(10, 27):

$27 = 10 \cdot 2 + 7$

$10 = 7 \cdot 1 + 3$

$7 = 3 \cdot 2 + 1$ So the gcd(10, 27) = 1.

Now reverse the process to find s and t such that $1 = 10s + 27t$.

$$\begin{aligned} 1 &= 7 - 3 \cdot 2 \\ &= 7 - (10 - 7 \cdot 1) \cdot 2 \\ &= 7 \cdot 3 - 10 \cdot 2 \\ &= (27 - 10 \cdot 2) \cdot 3 - 10 \cdot 2 \\ &= 10 \cdot (-8) + 27 \cdot 3. \end{aligned}$$

So $s = -8$ and $t = 3$. Thus $x = bs = -40$ is a solution, and the set of all solutions is $\{-40 + 27k \mid k \in \mathbf{Z}\}$.

Application IV

Chinese Remainder Theorem. Given n congruences

$$x \equiv a_1 \pmod{m_1}, \ldots, x \equiv a_n \pmod{m_n},$$

where $\gcd(m_i, m_j) = 1$ for each $i \neq j$. The following algorithm finds a unique solution x in the range $0 \leq x < m = m_1 \ldots m_n$.

(1) For each i find b_i such that $(m/m_i)b_i \equiv 1 \pmod{m_i}$.

(2) Set $x = (m/m_1)b_1a_1 + \ldots + (m/m_n)b_na_n$.

(3) If x is not in the proper range, then add or subtract a multiple of m.

Example. Solve the following three congruences for the unique x specified by the CRT.

$x \equiv 6 \pmod{5}$
$x \equiv 4 \pmod{7}$
$x \equiv 2 \pmod{11}$

Solution. Since the moduli are prime, the gcd requirement is satisfied.

(1) Find b_1, b_2, b_3 such that

$$7 \cdot 11 b_1 \equiv 1 \pmod{5}$$
$$5 \cdot 11 b_2 \equiv 1 \pmod{7}$$
$$5 \cdot 7 b_3 \equiv 1 \pmod{11}$$

Three solutions are $b_1 = 3$, $b_2 = 6$, and $b_3 = 6$.

(2) Set $x = 7 \cdot 11 \cdot 3 \cdot 6 + 5 \cdot 11 \cdot 6 \cdot 4 + 5 \cdot 7 \cdot 6 \cdot 2 = 3126$.

(3) Since 3126 is outside the range $0 \leq x < 5 \cdot 7 \cdot 11 = 385$, set $x = 3126 - 8 \cdot 385 = 46$.

Section 10.5.2 Cryptology: The RSA Algorithm

The algorithm allows the public to send encrypted messages by using a publicly available key, but to decrypt a message the receiver needs to know a privately held key.

The RSA Algorithm

Let p and q be primes and let $n = pq$. Let d satisfy the equation $\gcd(d, (p-1)(q-1)) = 1$. Let e be a solution to the congruence $de \equiv 1 \pmod{(p-1)(q-1)}$. If a is a message in the range $0 \le a < n$ where n and e are known to the sender, then the sender can encrypt a by calculating

$$c = a^e \bmod n.$$

The sender sends c. If the receiver knows d, then upon receipt of c the receiver can decrypt c by calculating

$$a = c^d \bmod n.$$

Usefulness

The RSA algorithm is useful because for very large primes, it is hard to factor n to find p and q. So it is very hard to find d. But there are some efficient algorithms to encrypt and decrypt a and c.

Example. Given primes $p = 7$ and $q = 13$, find two keys d and e.

Solution. Then $n = pq = 91$ and $(p-1)(q-1) = 6 \cdot 12 = 72$. Choose $d = 41$, since it satisfies the equation $\gcd(d, 72) = 1$. Now find a suitable value for e by solving the congruence $41e \equiv 1 \pmod{72}$. Using the Euclidean algorithm in reverse, it follows that $1 = 41 \cdot (-7) + 72 \cdot 4$. So we could pick e to be -7. But positive numbers are easier to work with, so we'll add a multiple of 72 to get $e = -7 + 72 = 65$.

Example. For the previous example, encrypt the message $a = 2$.

Solution: We need to calculate

$c = a^e \bmod n = 2^{65} \bmod 91$

$= (2^{12})^5 \cdot 2^5 \bmod 91$

$= (1)^5 \cdot 2^5 \bmod 91$ (Since $2^{12} \bmod 91 = 1$)

$= 32 \bmod 91$

$= 32.$

Example. For the previous examples, decrypt the encrypted message $c = 32$.

Solution: We need to calculate

$a = c^d \bmod n = (32)^{41} \bmod 91$

$= 2^{205} \bmod 91$

$= (2^{12})^{17} \cdot 2 \bmod 91$

$= (1)^{17} \cdot 2 \bmod 91$ (Since $2^{12} \bmod 91 = 1$)

$= 2 \bmod 91$

$= 2.$