

Definition 1: For any $n \geq 2$ define \mathbb{Z}_n to be the set of integers $\{0, 1, 2, 3, \dots, n-1\}$.

Solving congruences

Consider the function

$$f(x) = 7x + 4 \pmod{12}.$$

The implied domain of this function is the set of integers, and the codomain is the list of remainders $\{0, 1, 2, \dots, 11\}$ which are possible when dividing an integer by 12. That is, the codomain is \mathbb{Z}_{12} . When we look to *solve* the (congruence) equation

$$7x + 4 \equiv 9 \pmod{12}, \tag{1}$$

we seek to describe those inputs x to f which produce the particular output 9.

From these facts

1. If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $a + c \equiv b + d \pmod{m}$.
2. If $a \equiv b \pmod{m}$ and $c \in \mathbb{Z}$, then $ac \equiv bc \pmod{m}$.

we know that we can perform some of the basic steps of algebra. With regards to the example equation (1), we solve by

- adding 8 to both sides, which gets rid of the +4 since $(8 + 4) \pmod{12} = 0$

$$\left. \begin{array}{l} \text{new LHS : } 7x + 4 + 8 \equiv 7x + 12 \equiv 7x \pmod{12} \\ \text{new RHS : } 9 + 8 \equiv 17 \equiv 5 \pmod{12} \end{array} \right\} \Rightarrow 7x \equiv 5 \pmod{12}.$$

- multiplying both sides by 7, since $(7 \cdot 7) \pmod{12} = 1$.

$$\left. \begin{array}{l} \text{new LHS : } 7 \cdot 7x \equiv 49x \equiv x \pmod{12} \\ \text{new RHS : } 7 \cdot 5 \equiv 35 \equiv 11 \pmod{12} \end{array} \right\} \Rightarrow x \equiv 11 \pmod{12}.$$

These two steps have led to the solution: the integers x which satisfy Equation (1) are those which are equivalent to 11 $\pmod{12}$.

The general linear congruence equation, with modulus $m \geq 2$, looks like

$$ax + b \equiv n \pmod{m}. \tag{2}$$

It can be solved in much the same way as above—adding $(-b)$, the **additive inverse** of b , to both sides, then multiplying by the **multiplicative inverse** of a —provided that that $\gcd(a, m) = 1$ (a sufficient condition for a to *have* a multiplicative inverse \pmod{m}).

For small integers a, m it is generally possible to figure out

- what the $\gcd(a, m)$ is, and
- when $\gcd(a, m) = 1$, which number $\bar{a} \in \mathbb{Z}_m$ is the multiplicative inverse—i.e., satisfies $a\bar{a} \equiv 1 \pmod{m}$.

When these cannot be determined so easily, we resort to the Euclidean and Extended Euclidean Algorithms.¹

Example 1:

Show that the integers 311 and 6215 are relatively prime, and then find the multiplicative inverse of 311 $\pmod{6215}$.

Answer: We perform steps of the Euclidean algorithm (left side) and, rewrite (right side) the equations to express the newest remainder r_j in terms of two prior ones r_{j-1} and r_{j-2} (helpful steps for the extended Euclidean algorithm):

$$\begin{array}{ll} 6215 = (19)(311) + 306 & \Rightarrow \quad 306 = (1)(6215) - (19)(311) \\ 311 = (1)(306) + 5 & \quad 5 = (1)(311) - (1)(306) \\ 306 = (61)(5) + 1 & \quad 1 = (1)(306) - (61)(5) \\ 5 = (5)(1) + 0 & \end{array}$$

The last nonzero remainder is $\gcd(6215, 311)$, and since it is 1, the two numbers are relatively prime.

To find the multiplicative inverse, we use the equations, starting at the bottom, on the right-hand side, continually substituting the next-higher equation:

$$\begin{aligned} 1 &= (1)(306) - (61)(5) && \text{(bottom equation on the right)} \\ &= (1)(306) - (61)[(1)(311) - (1)(306)] &&= (62)(306) - (61)(311) \\ &= (62)[(1)(6215) - (19)(311)] - (61)(311) &&= (62)(6215) - (1239)(311) \\ &= (62)(6215) + (-1239)(311) \end{aligned}$$

If we consider the operations above as happening $\pmod{6215}$, we have

$$(62)(6215) + (-1239)(311) \equiv (4976)(311) \equiv 1 \pmod{6215}.$$

Thus, 4976 is the multiplicative inverse $\pmod{6215}$ of 311. ■

Example 2: Affine ciphers

¹An app that implements both the Euclidean and Extended Euclidean Algorithms is linked to the class webpage. The direct url is <https://www.extendedeuclideanalgorithm.com/calculator.php>.

Affine ciphers are based on functions of the form

$$f(x) := ax + b \pmod{26}.$$

When $\gcd(a, 26) = 1$, such a function $f: \mathbb{Z}_{26} \rightarrow \mathbb{Z}_{26}$ is *bijective* and is, thus, *invertible*.

One equates the 26 letters of the English alphabet with the numbers 0–25: $a \leftrightarrow 0, b \leftrightarrow 1, \dots, z \leftrightarrow 25$. This makes a natural map between simple strings of letters and finite sequences of integers, such as

$$\text{the word } pencil \leftrightarrow 15, 4, 13, 2, 8, 11,$$

which, in its numerical equivalent, could hardly be said to be “encrypted.” However, if we let $y = f(x)$, then an encrypted version of $x_1, x_2, x_3, x_4, x_5, x_6$ would be $y_1, y_2, y_3, y_4, y_5, y_6$, with

$$\begin{aligned} x_1 = 15 : & \quad y_1 = f(15) = 15a + b \pmod{26}, \\ x_2 = 4 : & \quad y_2 = f(4) = 4a + b \pmod{26}, \\ x_3 = 13 : & \quad y_3 = f(13) = 13a + b \pmod{26}, \\ x_4 = 2 : & \quad y_4 = f(2) = 2a + b \pmod{26}, \\ x_5 = 8 : & \quad y_5 = f(8) = 8a + b \pmod{26}, \\ x_6 = 11 : & \quad y_6 = f(11) = 11a + b \pmod{26}. \end{aligned}$$

In the case where $a = 19$ and $b = 4$, these encrypted values would be 3, 2, 17, 16, 0, 5, though we would generally transmit this as its alphabetic equivalent *dcrqaf*.

The person on the receiving end needs the inverse function to f in order to perform decryption of the message. We can obtain it in the same manner described above—by solving the congruence equation $ax + b \equiv y \pmod{26}$. If \bar{a} is the multiplicative inverse of $a \pmod{26}$, then

$$ax + b \equiv y \pmod{26} \quad \Rightarrow \quad x \equiv \bar{a}(y - b) \pmod{26}.$$

One needs $\gcd(a, 26) = 1$, of course, so that \bar{a} exists, in which case $g(y) = \bar{a}(y - b) \pmod{26}$ is the inverse function to $f(x) = ax + b \pmod{26}$.

You can explore affine ciphers without the tedium of all the letter-to-number conversions at <http://www.calvin.edu/~scofield/courses/m100/materials/scriptForms/affineTranslator.shtml> a link which appears on the class webpage. While affine ciphers are a neat application of congruences, they are quite easily broken.

■

Systems of linear congruences

A **system of congruences** is nothing more than multiple individual congruences, with the requirement that any solution x must be an integer which simultaneously satisfies them all. First, an

example.

Example 3:

Find integer solutions x which simultaneously solve the linear congruences

$$3x + 2 \equiv 4 \pmod{7} \quad \text{and} \quad 2x + 3 \equiv 2 \pmod{9}.$$

Answer: Using the techniques described in the previous section, we can solve the first of these congruences,² obtaining

$$x \equiv 3 \pmod{7}.$$

The integers which satisfy this requirement come from the list

$$\dots, -39, -32, -25, -18, -11, -4, 3, 10, 17, 24, 31, 38, 45, \dots,$$

and take the form $x = 7s + 3$, $s \in \mathbb{Z}$. That $x \equiv 3 \pmod{7}$, or $7 \mid (x - 3)$, stands as one requirement, the one imposed by the first congruence equation, on our solutions x . So, we insert the form $x = 7s + 3$, $s \in \mathbb{Z}$, into the second congruence equation:

$$2x + 3 \equiv 2(7s + 3) + 3 \equiv 2 \pmod{9} \quad \text{and solve to obtain} \quad s \equiv 4 \pmod{9},$$

which means $9 \mid (s - 4)$, or $s = 9t + 4$, $t \in \mathbb{Z}$. Putting these statements, together,

$$x = 7s + 3 = 7(9t + 4) + 3 = 63t + 31, \quad t \in \mathbb{Z},$$

which is the same as saying the solutions to this system of two linear congruences are precisely those x which satisfy

$$x \equiv 31 \pmod{63}.$$

One can check that these x , ones in the list

$$\dots, -158, -95, -32, 31, 94, \dots$$

are all in the list of solutions to $3x + 2 \equiv 4 \pmod{7}$, but also satisfy the other congruence relation $2x + 3 \equiv 2 \pmod{9}$. ■

We solved this system of linear congruence equations by a method Rosen calls **back substitution**. Specifically, we

- solved the first congruence, putting it in the form $x \equiv b \pmod{m}$.
- employed the fact that the integers which satisfy $x \equiv b \pmod{m}$ all take the form $x = ms + b$, where s is an integer.

²The key is that there be multiplicative inverse to 3 in $\pmod{7}$ arithmetic, which is so, since $\gcd(3, 7) = 1$.

- (back-)substituted the expression $ms + b$ in for x in the next congruence equation. One can repeatedly solve one congruence to get an expression that is inserted into the next one.

Is it possible to solve any system consisting of an arbitrary list of linear congruence relations in this manner? Indeed, not, when the system doesn't even have solutions, like in these scenarios:

- (i) Suppose one of your congruences were

$$2x \equiv 1 \pmod{4}.$$

Even with no knowledge of the other congruences in the system, this one fouls things up by itself, since it has no (integer) solutions.

- (ii) Suppose two of your congruences were

$$x \equiv 5 \pmod{7} \quad \text{and} \quad x \equiv 11 \pmod{14}.$$

The two lists of numbers these generate,

$$\dots, -16, -9, -2, 5, 12, 19, \dots$$

and

$$\dots, -17, -3, 11, 25, 39, \dots$$

have no overlaps—no common entries in both lists.

In these two scenarios, there are no solutions, so back-substitution will fail.

Is it possible to categorize situations in which solutions exist? That is one role the next theorem plays.

Theorem 1 (Chinese Remainder Theorem): Suppose the moduli m_1, m_2, \dots, m_n in the system of congruences

$$\begin{aligned} x &\equiv a_1 \pmod{m_1} \\ x &\equiv a_2 \pmod{m_2} \\ &\vdots \\ x &\equiv a_n \pmod{m_n} \end{aligned} \tag{3}$$

are pairwise relatively prime. Then the system has solutions which can be characterized by $x \equiv A \pmod{m}$, with $m := m_1 m_2 \cdots m_n$, for some $A \in \{0, 1, 2, \dots, m-1\}$.

Had we known this theorem prior to doing Example 3, we would have known the system was solvable because

- both congruences could be put into the form $x \equiv b_i \pmod{m_i}$. This required the existence of multiplicative inverses to 3 in $\pmod{7}$ arithmetic and to 2 in $\pmod{9}$ arithmetic. And
- the moduli, 7 and 9, were relatively prime.

We still had to do some work to figure out that $A = 31$ (for that example). If, however, two of our congruences have moduli $m_1 = 7$ and $m_2 = 14$ (as in Scenario (ii) above), then we have no assurance of a solution, since $\gcd(7, 14) \neq 1$.

An alternative to back-substitution. Often the proof of a mathematical fact is constructive, providing an algorithm. In particular, the proof of the Chinese Remainder Theorem offers an alternative to back-substitution for solving a system of linear congruences already in the form (3). Under the assumptions of the theorem, if we use the moduli m_1, \dots, m_n to form the related list of n numbers

$$M_i = \frac{m_1 m_2 \cdots m_n}{m_i} = \frac{m}{m_i}, \quad k = 1, \dots, n,$$

(i.e, M_1 is the product of the moduli leaving m_1 out, M_2 is the product of moduli leaving m_2 out, etc.), then for each $1 \leq i \leq n$, the numbers m_i and M_i are relatively prime. This means that M_i has a multiplicative inverse, call it y_i , in $\pmod{m_i}$ arithmetic. Having found these multiplicative inverses so that $M_i y_i \equiv 1 \pmod{m_i}$, $1 \leq i \leq n$, we may take

$$A = \sum_{i=1}^n a_i M_i y_i \pmod{m_1 m_2 \cdots m_n} = a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots a_n M_n y_n \pmod{m}.$$

The solutions, then, of the system (3) are those $x \equiv A \pmod{m}$.

See Example 5 in the Rosen text, pp. 278–279 to see this method used to solve the system of congruences posed in Example 4, p. 277.

Representations of (large) integers and the Chinese Remainder Theorem

The integers 0–255 are all representable as 8-bit binary strings:

n	8-bit binary representation	n	8-bit binary representation
0	00000000	127	01111110
1	00000001	\vdots	\vdots
2	00000010	252	11111100
3	00000011	253	11111101
4	00000100	254	11111110
\vdots	\vdots	255	11111111

The Chinese Remainder Theorem indicates that the system of congruences (3) offers an *alternative way* of representing uniquely all integers $k \in \mathbb{Z}_m$ (still taking $m = m_1 m_2 \cdots m_n$)—namely, we represent k as the n -tuple $(k \pmod{m_1}, k \pmod{m_2}, \dots, k \pmod{m_n})$. With the understanding that

the tuple $(32, 18, 91)$ is the representation of some number k with $0 \leq k < (97)(98)(99) = 941,094$ using the three moduli $m_1 = 97$, $m_2 = 98$, $m_3 = 99$ (pairwise relatively prime), we could recover k by solving the system

$$x \equiv 32 \pmod{97}, \quad x \equiv 18 \pmod{98}, \quad x \equiv 91 \pmod{99}.$$

See Example 7, p. 279 for the representations of numbers in \mathbb{Z}_{12} (i.e., 0–11) when the two relatively prime moduli $m_1 = 3$ and $m_2 = 4$ are used.