The Euclidean Algorithm

We already observed that

Z_n is a ring for any positive integer n.

We also proved there that

An element b from Z_n has a multiplicative inverse if and only if $\gcd(b, n) = 1$,

and that

The number of positive integers less than n and relatively prime to n is $\phi(n)$.

The set of residues modulo n that are relatively prime to n is denoted Z_n^*.

I t is not hard to see that forms an abelian group under multiplication.

We already have stated that multiplication modulo n is associative and commutative, and that 1 is the multiplicative identity. Any element in will have a multiplicative inverse (which is also in Z_n^*.).

Finally, is closed under multiplication since xy is relatively prime to n whenever x and y are relatively prime to n (prove this!).

At this point, we know that any b from Z_n^*has a multiplicative inverse, b^{-1}, but we do not yet have an efficient algorithm to compute b^{-1}.
Such an algorithm exists; it is called the extended Euclidean algorithm.
First, we describe the Euclidean algorithm, in its basic form, which is used to compute the greatest common divisor of two positive integers, say r_0 and r_1, where $r_0 > r_1$.

The Euclidean algorithm consists of performing the following sequence of divisions:

$$
\begin{aligned}
r_0 &= q_1 r_1 + r_2, && 0 < r_2 < r_1 \\
r_1 &= q_2 r_2 + r_3, && 0 < r_3 < r_2 \\
&\vdots \\
r_{m-2} &= q_{m-1} r_{m-1} + r_m, && 0 < r_m < r_{m-1} \\
r_{m-1} &= q_m r_m.
\end{aligned}
$$

Then it is not hard to show that

$$\gcd(r_0, r_1) = \gcd(r_1, r_2) = \ldots = \gcd(r_{m-1}, r_m) = r_m.$$

Hence, it follows that gcd(r_0 , r_1)=r_m.

Since the Euclidean algorithm computes greatest common divisors,
it can be used to determine b^{-1}, if a positive integer $b < n$ has a multiplicative inverse modulo n. We shall start starting $r_0=n$ and $r_1= b$.
However, it does not compute the value of the multiplicative inverse (if it exists).
Now, suppose we define a sequence of numbers t_0, t_1, . . ., t_m
according to the following recurrence (where the q_j's are defined as above):

$$\begin{aligned} t_0 &= 0 \\ t_1 &= 1 \\ t_j &= t_{j-2} - q_{j-1}t_{j-1} \bmod r_0, \quad \text{if } j \geq 2. \end{aligned}$$

Then we have the following useful result.

THEOREM
For $0 \leq j \leq m$, *we have that* $r_j \equiv t_j r_1 \pmod{r_0}$,
where the q_j*'s and* r_j*'s are defined as in the Euclidean algorithm, and the* t_j*'s are defined in the above recurrence.*

The next corollary is an immediate consequence.

COROLLARY
Suppose $\gcd(r_0, r_1) = 1$. *Then* $t_m = r_1^{-1} \pmod{r_0}$.

Now, the sequence of numbers $t_0, t_1, \ldots t_m$
can be calculated in the Euclidean algorithm at the same time as the q_j 's and the r_j's.

We present the extended Euclidean algorithm to compute the inverse of b modulo n, if it exists. In this version of the algorithm, we do not use an array to keep track of the q_j 's, r_j's and t_j's, since it suffices to remember only the "last" two terms in each of these sequences at any point in the algorithm.
In step 10 of the algorithm, we have written the expression for *temp* in such a way that the reduction modulo n is done with a positive argument. (We mentioned earlier that modular reductions of negative numbers yield negative results in many computer languages; of course, we want to end up with a

positive result here.) We also mention that at step 12, it is always the case that $tb \equiv r \pmod{n}$ (this is the result proved in Theorem.

Here is a small example to illustrate:

Example
Suppose we wish to compute $28^{-1} \bmod 75$. The Extended Euclidean algorithm proceeds as follows:

$75 = 2 \times 28 + 19$	step 6
$73 \times 28 \bmod 75 = 19$	step 12
$28 = 1 \times 19 + 9$	step 16
$3 \times 28 \bmod 75 = 9$	step 12
$19 = 2 \times 9 + 1$	step 16
$67 \times 28 \bmod 75 = 1$	step 12
$9 = 9 \times 1$	step 16

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

The Chinese Remainder Theorem

The Chinese remainder theorem is really a method of solving certain systems of congruences. Suppose $m_1, \ldots, m_r$ are pairwise relatively prime positive integers (that is, gcd $(m_i, m_j) = 1$ if $i \neq j$). Suppose $a_1, \ldots, a_r$ are integers, and consider the following system of congruences:

$$x \equiv a_1 \pmod{m_1}$$
$$x \equiv a_2 \pmod{m_2}$$
$$\vdots$$
$$x \equiv a_r \pmod{m_r}.$$

The Chinese remainder theorem asserts that this system has a unique solution modulo $M = m_1 \times m_2 \times \ldots \times m_r$. We will prove this result in this section, and describe an efficient algorithm for solving systems of congruences of this type. It is convenient to study the function, which we define as follows:

Example 4.2
Suppose $r = 2$, $m_1 = 5$ and $m_2 = 3$, so $M = 15$. Then the function π has the following values:

$$\begin{array}{lllllllll} \pi(0) & = & (0,0) & \pi(1) & = & (1,1) & \pi(2) & = & (2,2) \\ \pi(3) & = & (3,0) & \pi(4) & = & (4,1) & \pi(5) & = & (0,2) \\ \pi(6) & = & (1,0) & \pi(7) & = & (2,1) & \pi(8) & = & (3,2) \\ \pi(9) & = & (4,0) & \pi(10) & = & (0,1) & \pi(11) & = & (1,2) \\ \pi(12) & = & (2,0) & \pi(13) & = & (3,1) & \pi(14) & = & (4,2). \end{array}$$

Example 4.3
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 function is the following:

$$\pi^{-1}(a_1, a_2, a_3) = 715a_1 + 364a_2 + 924a_3 \bmod 1001.$$

For example, if $x \equiv 5 \pmod 7$, $x \equiv 3 \pmod{11}$ and $x \equiv 10 \pmod{13}$, then this formula tells us that

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

This can be verified by reducing 894 modulo 7, 11 and 13.

For future reference, we record the results of this section as a theorem.

THEOREM *(Chinese Remainder Theorem)*
Suppose $m_1, \ldots, m_r$
are pairwise relatively prime positive integers, and suppose
$a_1, \ldots, a_r$ are integers.
Then, the system of r congruences
$x \equiv a_i \pmod{m_i}$ $(1 \le i \le r)$ has a unique solution modulo
$M = m_1 \times \ldots \times m_r$,
which is given by

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

where $M_i = M/m_i$ and $y_i = M_i^{-1} \bmod m_i$, for $1 \le i \le r$.

Other Useful Facts

We next mention another result from elementary group theory, called Lagrange's Theorem, that will be relevant in our treatment of the **RSA Cryptosystem**. For a (finite) multiplicative group G, define the *order* of an element $g \in G$ to be the smallest positive integer m such that $g_m = 1$. The following result is fairly simple, but we will not prove it here.

THEOREM *(Lagrange)*
Suppose G is a multiplicative group of order n, and $g \in G$. Then the order of g divides n.
For our purposes, the following corollaries are essential.

COROLLARY
If , then $b_{\phi(n)} \equiv 1 \pmod{n}$.
PROOF is a multiplicative group of order $\phi(n)$.

COROLLARY *(Fermat)*
Suppose p is prime and . Then $b_p \equiv b \pmod{p}$.
PROOF If p is prime, then $\phi(p) = p - 1$. So, for (mod p), the result follows from Corollary
For $b \equiv 0 \pmod{p}$, the result is also true since 0
$_p \equiv 0 \pmod{p}$.
At this point, we know that if p is prime, then is a group of order $p - 1$, and any element in has order dividing $p - 1$. However, if p is prime, then the group is in fact *cyclic*: there exists an element having order equal to $p - 1$. We will not prove this very important fact, but we do record it for future reference:

THEOREM
If p is prime, then is a cyclic group.
An element α having order $p - 1$ is called a *primitive* element modulo p. Observe that α is a primitive element if and only if

$$\{\alpha^i : 0 \le i \le p-2\} = \mathbb{Z}_p^*.$$

Now, suppose p is prime and α is a primitive element modulo p. Any element can be written as
$\beta = \alpha_i$
, where $0 \le i \le p - 2$, in a unique way. It is not difficult to prove that the order of
$\beta = \alpha_i$
is
$$\frac{p-1}{\gcd(p-1,i)}.$$

Thus β is itself a primitive element if and only if gcd(p - 1, i) = 1. It follows that the number of primitive elements modulo p is $\phi(p - 1)$.

Example
Suppose p = 13. By computing successive powers of 2, we can verify that 2 is a primitive element modulo 13:

$2^0 \bmod 13 = 1$
$2^1 \bmod 13 = 2$
$2^2 \bmod 13 = 4$
$2^3 \bmod 13 = 8$
$2^4 \bmod 13 = 3$
$2^5 \bmod 13 = 6$
$2^6 \bmod 13 = 12$
$2^7 \bmod 13 = 11$
$2^8 \bmod 13 = 9$
$2^9 \bmod 13 = 5$
$2^{10} \bmod 13 = 10$
$2^{11} \bmod 13 = 7.$

The element 2^i
is primitive if and only if gcd(i, 12) = 1; i.e., if and only if i = 1, 5, 7 or 11.
Hence, the primitive elements modulo 13 are 2, 6, 7 and 11.