

CDM

Basic Number Theory I

K. SUTNER

CARNEGIE MELLON UNIVERSITY



- 1 **Basic Arithmetic**
- 2 **Modular Arithmetic**
- 3 **Chinese Remainder**
- 4 **Towards Algebra**

For $a, b \in \mathbb{Z}$, a **divides** b iff $\exists c \in \mathbb{Z} (a \cdot c = b)$.

This is usually written $a \mid b$.

Note that $\pm 1 \mid a$ and $a \mid 0$.

Proposition

Divisibility is reflexive, transitive and almost antisymmetric.

Lemma (Linear Combinations)

If $d \mid a$ and $d \mid b$ then $d \mid (xa + yb)$ for all $x, y \in \mathbb{Z}$.

Theorem (Division Theorem)

Let b be positive, and a an arbitrary integer. Then there exist integers q and r such that

$$a = q \cdot b + r, \text{ where } 0 \leq r < b.$$

*Moreover, the numbers q and r are uniquely determined (**quotient** and **remainder**).*

In the literature this is often called the “Division Algorithm,” though no algorithm is given.

Notation:

$$r = a \bmod b \qquad \text{remainder}$$

$$q = a \operatorname{div} b \qquad \text{quotient}$$

$p > 1$ is **prime** iff its only positive divisors are 1 and p .

Lemma

For every $n \geq 2$ there is a prime p such that $p \mid n$.

Theorem

There are infinitely many primes.

Lemma

If p is prime and $p \mid ab$ then $p \mid a$ or $p \mid b$.

Lemma 1: Strong induction.

Theorem: Assume p_1, p_2, \dots, p_n is a list of all primes and consider $p_1 \cdot p_2 \cdot \dots \cdot p_n + 1$.

Lemma 2: Use the GCD, see below.

Let $d = \gcd(a, p)$. If $d = p$ then clearly d divides a .

Otherwise $d = 1$, hence $xa + yp = 1$ p divides b .

Theorem

Let $n \geq 2$. Then there exist distinct primes p_1, \dots, p_k such that

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

where $e_i > 0$. The decomposition is unique up to order.

Proof.

Strong induction and the last lemma.



But beware, finding this prime decomposition is very hard. It is exceedingly useful conceptually, but algorithmically there are issues.

The **greatest common divisor** is defined by

$$\gcd(a, b) = \max(d \mid d \text{ divides } a, b)$$

a and b are **coprime** (relatively prime) iff $\gcd(a, b) = 1$.

A look at the GCD function produces an algorithm to compute it.

Lemma

- $\gcd(x, 0) = x$
- $\gcd(x, y) = \gcd(y, x)$
- $\gcd(x, y) = \gcd(y, x \bmod y)$

Typical run: $a = 4234$ and $b = 4693286$.

$$4234 = 0 \cdot 4693286 + 4234$$

$$4693286 = 1108 \cdot 4234 + 2014$$

$$4234 = 2 \cdot 2014 + 206$$

$$2014 = 9 \cdot 206 + 160$$

$$206 = 1 \cdot 160 + 46$$

$$160 = 3 \cdot 46 + 22$$

$$46 = 2 \cdot 22 + 2$$

$$22 = 11 \cdot 2 + 0$$

The table is a (clumsy) proof that $\gcd(4234, 4693286) = 2$.

The last example suggests to take a closer look at **linear combinations**

$$c = x \cdot a + y \cdot b$$

where $x, y \in \mathbb{Z}$.

Obviously c is divisible by $\gcd(a, b)$.

More interestingly, we could run through the equations above backwards and write $2 = \gcd(a, b)$ as a linear combination of a and b :

$$\gcd(a, b) = 2 = 205068 \cdot a - 185 \cdot b$$

Lemma (Extended Euclidean Algorithm)

There exist integers x, y such that

$$\gcd(a, b) = x \cdot a + y \cdot b.$$

*Moreover, these so-called **cofactors** can be computed along with the GCD.*

Compute a trace of the Euclidean algorithm. Wlog $a \geq b \geq 0$.

$$r_{i-2} = q_i \cdot r_{i-1} + r_i \quad \text{where } r_0 = a, r_1 = b$$

Hence, $r_n = 0$ for some n , and $r_{n-1} = \gcd(a, b)$. Define

$$x_0 = 1$$

$$y_0 = 0$$

$$x_1 = 0$$

$$y_1 = 1$$

$$x_i = x_{i-2} - q_i \cdot x_{i-1}$$

$$y_i = y_{i-2} - q_i \cdot y_{i-1}$$

A simple induction shows that

$$r_i = a \cdot x_i + b \cdot y_i.$$

q_i	r_i	x_i	y_i
—	1233	1	0
—	1000	0	1
1	233	1	-1
4	68	-4	5
3	29	13	-16
2	10	-30	37
2	9	73	-90
1	1	-103	127
9	0	1000	-1233

We have

$$-103 \cdot 1233 + 127 \cdot 1000 = 1 = \gcd(1233, 1000)$$

We can also think of

$$a \cdot x + b \cdot y = c$$

as an equation, we want solutions for x and y .

Again, we clearly need $d = \gcd(a, b) \mid c$ for any solution to exist.

We can divide by the GCD and use the extended Euclidean algorithm as before.

But note that the solution is not unique: for any solution (x_0, y_0) we get infinitely many other solutions of the form

$$(x_0 + tb/d, y_0 - ta/d)$$

where $t \in \mathbb{Z}$. In fact, these are all the solutions.

One can implement all the necessary arithmetic in $O(k^2)$ steps for k -bit numbers. In fact addition is only $O(k)$, but for mods and remainders we need $O(k^2)$ steps.

But how often does the while-loop execute? Trivially no more than $a \geq b$ times, but that's no good at all.

Note that one must lose one bit at least at every other step. This follows from

$$r_{i-2} = q_i \cdot r_{i-1} + r_i$$

Hence total running time is $O(k^3)$ steps for k -bit inputs.

Incidentally, the worst possible input is two consecutive Fibonacci numbers. In this case, $q_i = 1$ at all times, and the algorithm just runs backwards through the Fibonacci numbers.

Definition

Let p prime. The p -adic valuation of an integer $n \neq 0$ is the largest e such that p^e divides n , in symbols $\nu_p(n)$; we set $\nu_p(0) = \infty$.

$$\nu_p(ab) = \nu_p(a) + \nu_p(b)$$

$$a \mid b \iff \forall p (\nu_p(a) \leq \nu_p(b))$$

$$\gcd(a, b) = \prod_p p^{\min(\nu_p(a), \nu_p(b))}$$

The last formula does not yield an efficient way to compute gcd's.

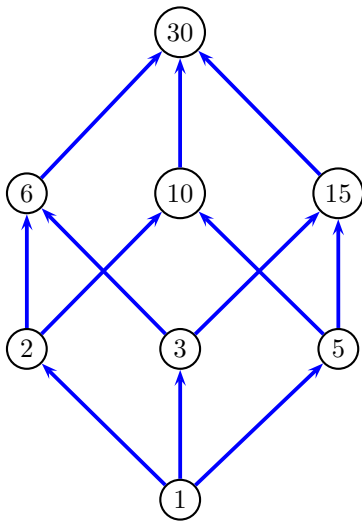
The natural numbers with division $\langle \mathbb{N}, | \rangle$ form a so-called **lattice**: a partial order where any two elements have a join (supremum) and and a meet (infimum).

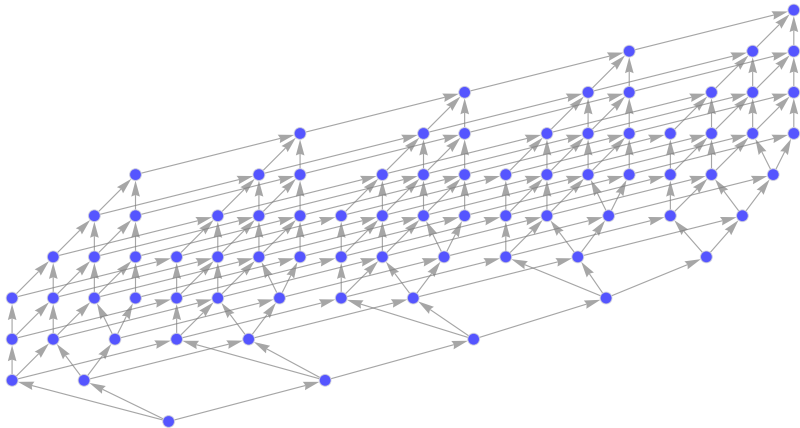
The join is the least common multiple, the meet the greatest common divisor.

If you prefer, you can think of a structure $\langle A, \sqcup, \sqcap \rangle$ where

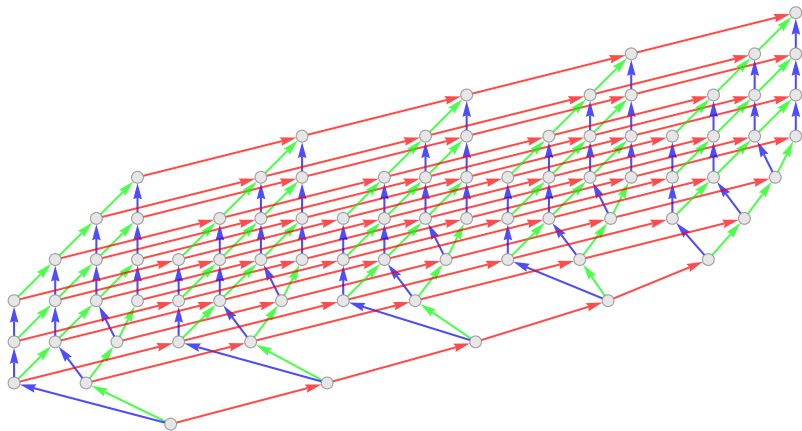
- \sqcup and \sqcap are associative and commutative
- **absorption** holds:

$$x \sqcup (x \sqcap y) = x \qquad x \sqcap (x \sqcup y) = x$$





Challenge: Explain the picture.



Exercise

Verify that $\langle \mathbb{N}, \text{lcm}, \text{gcd} \rangle$ really forms lattice.

Exercise

How are lcm and gcd expressed in the picture of the divisor lattice of 30?

Exercise

How is the structure of prime divisors of $148176 = 2^4 3^3 7^3$ expressed in the picture of the divisor lattice?

1 Basic Arithmetic

2 **Modular Arithmetic**

3 Chinese Remainder

4 Towards Algebra

Suppose we have a polynomial with integer coefficients

$$p(x) = a \cdot x^3 + b \cdot x^2 + c \cdot x + d.$$

Assume that both $p(0)$ and $p(1)$ are odd.

Claim

For all integers x , $p(x) \neq 0$.

To see why, first note that $d = p(0)$ and $a + b + c + d = p(1)$ are odd.

Even/Odd arithmetic:

+	even	odd
even	even	odd
odd	odd	even

·	even	odd
even	even	even
odd	even	odd

Hence n even (odd) implies n^k even (odd) for all $k \geq 1$.

Case 1: So for even x we get

$$p(x) = a \cdot \text{even} + b \cdot \text{even} + c \cdot \text{even} + \text{odd} = \text{odd}$$

so that in particular $p(x) \neq 0$.

Case 2: For odd x we have

$$p(x) = a \cdot \text{odd} + b \cdot \text{odd} + c \cdot \text{odd} + \text{odd}.$$

But $a + b + c$ must be even, so either 0 or 2 of these coefficients must be odd.

In both cases $p(x)$ is odd, and so not equal to 0.



Carl Friedrich Gauss, 1777-1855, my academic grand⁷-father.

Recall from equivalence relations: for $m \geq 0$, x is congruent to y modulo m

$$x \rho y \iff m \text{ divides } x - y$$

is an equivalence relation.

Notation: $x = y \pmod{m}$ or $x \equiv y \pmod{m}$.

Crucial Point: we can define arithmetic on the equivalence classes to get a structure \mathbb{Z}_m as opposed to \mathbb{Z} :

$$[x] + [y] = [x + y]$$

$$[x] \cdot [y] = [x \cdot y]$$

We obtain **modular numbers**.

A notation like $[x]$ or $[x]_m$ or $[x]_{\mathbb{Z}_m}$ for modular numbers is perfectly correct but fatally clumsy. Likewise for $+_m$ or $+\mathbb{Z}_m$.

To keep notation simple, we will usually ignore the brackets and write x instead of $[x]$. And we write $+$ and \cdot for addition and multiplication of modular numbers.

So, we write $\mathbb{Z}_m = \{0, 1, \dots, m-1\}$ and sometimes think of $\mathbb{Z}_m \subseteq \mathbb{Z}$.

This is the canonical way of choosing representatives, but note that there are other possibilities. For example, for $m = 2k + 1$ we could also use

$$-k, -k+1, \dots, -1, 0, 1, \dots, k-1, k$$

Example (\mathbb{Z}_2)

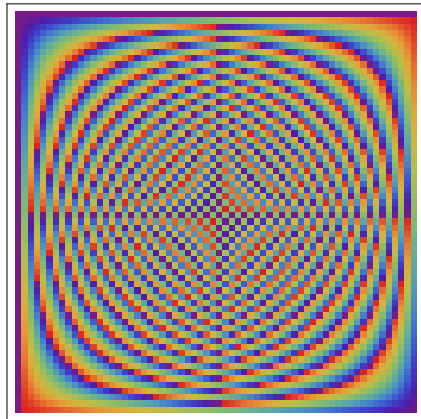
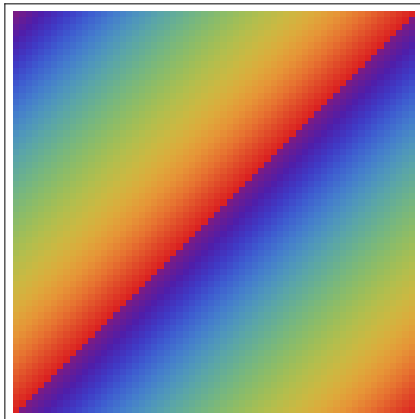
+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

Example (\mathbb{Z}_5)

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1



There is a very important idea hiding here: equivalence relations that are compatible with arithmetic (here: on modular numbers).

Suppose ρ is an equivalence relation on \mathbb{Z} . ρ is a **congruence** iff

$$x = x' \pmod{m} \text{ and } y = y' \pmod{m}$$

implies

$$x + y = x' + y' \pmod{m}$$

$$x \cdot y = x' \cdot y' \pmod{m}$$

This is a huge restriction compared to arbitrary equivalence relations. There are uncountably many equivalence relations on \mathbb{Z} , but we know all the congruences: the Gaussian relations $\text{mod } m$.

Exercise

Prove that any congruence on the integers is already of the form \pmod{m} .

Think of $\text{mod } m$ as a function from \mathbb{Z} to $\mathbb{Z}_m \subseteq \mathbb{Z}$.

Then we have

$$\begin{aligned}(x + y) \text{ mod } m &= ((x \text{ mod } m) + (y \text{ mod } m)) \text{ mod } m \\(x \cdot y) \text{ mod } m &= ((x \text{ mod } m) \cdot (y \text{ mod } m)) \text{ mod } m\end{aligned}$$

Note that the double application of $\text{mod } m$ on the right is clumsy, we'll see a better way in a while (homomorphisms).

A clock (which functions accurately) shows the hour hand positioned at a minute mark, and the the minute hand two marks away. What time is it?

Really have 60 possible positions. Equations:

$$m = h \pm 2 \pmod{60}$$

$$m = 12h \pmod{60}$$

By exploiting the congruence properties it follows that

$$11h = \pm 2 \pmod{60}$$

multiply by 11:

$$h = \pm 22 \pmod{60}$$

It's 4:24 or 7:36.

The real question is: how does one solve equations modulo m ?

Since there are only finitely many modular numbers one could, in principle, use brute force. Alas, for large moduli this is not a realistic option. (Though we will admit that sometimes brute force works modulo m).

Unfortunately life becomes fairly difficult even for quadratic equations, but we can handle linear ones $ax = c \pmod{m}$.

Proposition

*Let $ab = ac \pmod{m}$ and $m' = m / \gcd(a, m)$.
Then $b = c \pmod{m'}$.*

In particular when a and m are coprime we can simply drop the a .

Exercise

Use p -adic valuations to prove the proposition.

First an important special case.

Lemma

The equation

$$a \cdot x = 1 \pmod{m}$$

has a solution if, and only if, a and m are coprime. The solution is unique modulo m , if it exists.

Proof.

A solution means that $ax - 1 = qm$, so a and m must be coprime.

In the opposite direction use the extended Euclidean algorithm to compute cofactors $ax + my = 1$.

□

The situation in the lemma is very important.

The solution x such that $ax = 1 \pmod{m}$ is called the **multiplicative inverse** of $a \pmod{m}$.

Notation: $a^{-1} \pmod{m}$.

Example

$m = 11$.

x	1	2	3	4	5	6	7	8	9	10
x^{-1}	1	6	4	3	9	2	8	7	5	10

Note that $10 = 10^{-1}$ (no surprise, really: $10 = -1$).

So $1/2 = 6 \pmod{11}$.

The collection of all modular numbers that have a multiplicative inverse is usually written \mathbb{Z}_m^* and called the **multiplicative subgroup** (see next week).

$$\mathbb{Z}_m^* = \{ a \in \mathbb{Z}_m \mid \gcd(a, m) = 1 \}$$

Definition (Euler's Totient Function)

The cardinality of \mathbb{Z}_m^* is written $\varphi(m)$.

Here are the first few values of φ

1, 1, 2, 2, 4, 2, 6, 4, 6, 4, 10, 4, 12, 6, 8, 8, 16, 6, 18, 8, 12, 10, 22, 8, 20, 12, 18, 12, 28,
8, 30, 16, 20, 16, 24, 12, 36, 18, 24, 16, 40, 12, 42, 20, 24, 22, 46, 16, 42, 20, 32, 24, 52

Looks complicated.

Note that we can compute $\varphi(n)$ if we know the prime factorization of n :

- For p prime $\varphi(p) = p - 1$ and $\varphi(p^k) = (p - 1)p^{k-1}$.
- For m and n coprime, $\varphi(mn) = \varphi(m)\varphi(n)$.

We will see an elegant proof of the second claim later.

Hence for prime decomposition

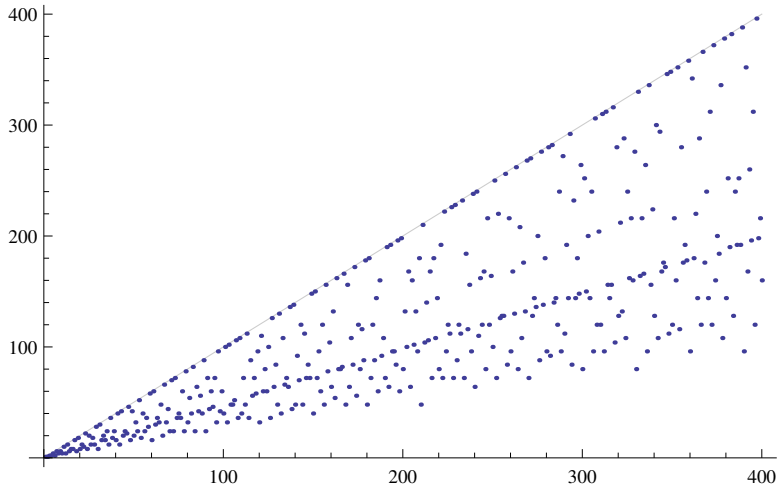
$$n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$$

we have

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

Note that it is not clear how to compute $\varphi(n)$ without the prime decomposition.

In fact, the two problems are computationally closely related. Some cryptographic schemes depend on the totient function being hard to compute.



Lemma

In the general case $a \cdot x = c \pmod{m}$ we have a solution if, and only if, $\gcd(a, m)$ divides c .

Moreover, the number of solutions is $\gcd(a, m)$.

Exercise

Prove the general case.

Consider the additive function

$$\begin{aligned}\alpha : \mathbb{Z}_m &\rightarrow \mathbb{Z}_m \\ x &\mapsto x + s \bmod m\end{aligned}$$

Clearly α is injective, so the orbits are all periodic (plain cycles).

Moreover, since $\alpha(x) + y = \alpha(x + y) \pmod{m}$ all the cycles are just rotations of each other and it suffices to understand $\text{orb}(0, \alpha)$.

So we need the least $k > 0$ such that $ks = 0 \pmod{m}$.

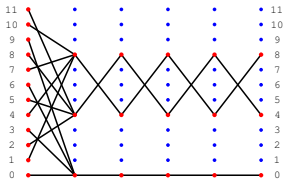
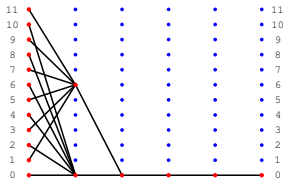
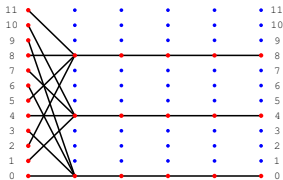
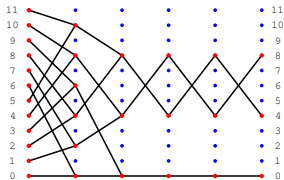
Let $d = \gcd(s, m)$. Then clearly $k = m/d$.

Proposition

α has $\gcd(s, m)$ distinct orbits, each of length $m / \gcd(s, m)$.

We also saw that the corresponding problem for multiplication is quite a bit harder.

$$\mu(x) = s \cdot x \bmod m$$



Naturally one should try to understand the case when μ is injective first, since all orbits are just cycles.

Clearly μ is injective iff s and m are coprime, i.e., $s \in \mathbb{Z}_m^*$.

0 is a fixed point, so one should consider the orbit of 1 next.

So this time we need the least $k > 0$ such that $s^k = 1 \pmod{m}$.

This number k has a natural algebraic interpretation in \mathbb{Z}_m^* that we will discuss shortly.

Exercise

Determine the structure of μ orbits in the “anti-coprime case”: every prime factor of m also divides s .

Exercise

Determine the structure of μ orbits when $m = pq$, p, q prime and $s = p$.

Exercise

Determine the structure of μ orbits in the general mixed case.

When p is prime the structure of \mathbb{Z}_p^\star is particularly simple:

$$\mathbb{Z}_p^\star = \{1, 2, \dots, p-1\}$$

As a consequence, we can solve all equations $ax = b \pmod{p}$ as long as $a \not\equiv 0 \pmod{p}$.

Here are some classical results concerning prime moduli.

Theorem (Wilson's Theorem)

p is prime if, and only if, $(p-1)! \equiv -1 \pmod{p}$.

Proof.

First assume p is prime, wlog $p > 2$. We can pair off $a \in \mathbb{Z}_p^\star$ and $a^{-1} \in \mathbb{Z}_p^\star$.

a and a^{-1} are always distinct except in the case $a = \pm 1$: the quadratic equation $x^2 = 1 \pmod{p}$ has at most two solutions since $x^2 - 1 = (x+1)(x-1)$.

For the opposite direction assume p fails to be prime, say, $ab = p$ for $1 < a < b < p$. But then $(p-1)!$ and p are not coprime whereas -1 and p are coprime, contradiction.

□

Theorem (Fermat's Little Theorem)

If p is prime and coprime to a , then $a^{p-1} = 1 \pmod{p}$.

Proof.

Consider the map $f : \mathbb{Z}_p^\star \rightarrow \mathbb{Z}_p^\star$, $f(x) = ax$.

f is a bijection, so

$$a^{p-1} \prod_{x \in \mathbb{Z}_p^\star} x = \prod_{x \in \mathbb{Z}_p^\star} ax = \prod_{x \in \mathbb{Z}_p^\star} f(x) = \prod_{x \in \mathbb{Z}_p^\star} x \pmod{p}$$

Since $\varphi(p) = p - 1$, done.

□

We will see a stronger version of this in our discussion of groups.

1 Basic Arithmetic

2 Modular Arithmetic

3 **Chinese Remainder**

4 Towards Algebra

How about multiple equations with several moduli:

$$a_i x = b_i \pmod{m_i} \quad \text{where } i = 1, \dots, n$$

We can simplify this system a little: for a solution to exist we need that $\gcd(a_i, m_i)$ divides b_i .

So we get an equivalent equation $a'_i x = b'_i \pmod{m'_i}$ where a'_i and m'_i are coprime.

But that is equivalent to $x = c_i \pmod{m'_i}$ for some appropriate c_i .

So let's only consider

$$x = a_i \pmod{m_i} \quad i = 1, \dots, n$$

Tricky in general, but for coprime moduli easy. We only consider $n = 2$.

Let $m = m_1 m_2$ and define the function

$$\begin{aligned} f : \mathbb{Z}_m &\longrightarrow \mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \\ f(x) &= (x \bmod m_1, x \bmod m_2) \end{aligned}$$

Claim

f is injective and hence bijective.

Proof. To see this, suppose $f(x) = f(x')$, where $0 \leq x \leq x' < m$.

Then

$$x' - x = q_1 m_1 = q_2 m_2.$$

But m_1 and m_2 are coprime, so $m \mid x' - x$ and therefore $x = x'$.

Since domain and codomain of f both have cardinality m , f must be a bijection by General Abstract Nonsense.

□

Hence we can solve $x = a \pmod{m_1}$ and $x = b \pmod{m_2}$: let

$$x = f^{-1}(a, b)$$

Great. But how do we find the x computationally?

Let $m_1 = 3$ and $m_2 = 5$, so $m = 15$.

Here is the canonical map $f : \mathbb{Z}_{15} \rightarrow \mathbb{Z}_3 \times \mathbb{Z}_5$, $f(x) = (x \bmod 3, x \bmod 5)$.

0	(0, 0)	8	(2, 3)
1	(1, 1)	9	(0, 4)
2	(2, 2)	10	(1, 0)
3	(0, 3)	11	(2, 1)
4	(1, 4)	12	(0, 2)
5	(2, 0)	13	(1, 3)
6	(0, 1)	14	(2, 4)
7	(1, 2)		

By table lookup, the solution to $x = 2 \bmod 3$, $x = 1 \bmod 5$ is $x = f^{-1}(2, 1) = 11$.

A better method is to use the EEA. Compute the cofactors:

$$\alpha m_1 + \beta m_2 = 1$$

Then

$$f(\alpha m_1) = (0, 1)$$

$$f(\beta m_2) = (1, 0)$$

whence

$$f(b\alpha m_1 + a\beta m_2) = (a, b)$$

So the solution is $x = b \cdot \alpha m_1 + a \cdot \beta m_2$.

As we have seen, the solution to

$$x = 2 \bmod 3 \qquad x = 1 \bmod 5$$

is $x = 11$.

Here is the computationally superior solution: determine cofactors

$$(-3) \cdot 3 + 2 \cdot 5 = 1$$

which produce a solution

$$x = 1 \cdot (-3) \cdot 3 + 2 \cdot 2 \cdot 5 = 11$$

Our result also holds for more than 2 equations (and is very old).

Theorem (CRT)

Let $m_i, i = 1, \dots, n$ be pairwise coprime. Then the equations

$$x = a_i \pmod{m_i} \qquad i = 1, \dots, n$$

have a unique solution in \mathbb{Z}_m , $m = m_1 m_2 \dots m_n$.

This follows from repeated application of the solution for $n = 2$ since m_1 and $m_2 \dots m_n$ are also coprime.

How do we compute the solution for $n > 2$? We could use the method for $n = 2$ recursively, but that is a bit tedious. Here is a better way.

Define

$$c_i = m/m_i$$

so that $c_i = 0 \pmod{m_j}$, $i \neq j$, but c_i and m_i are coprime. Use EEA to find inverses

$$\alpha_i c_i = 1 \pmod{m_i}$$

Then

$$x = a_1 \alpha_1 c_1 + a_2 \alpha_2 c_2 + \dots a_n \alpha_n c_n \pmod{m}$$

A bored bank clerk has a big pile of one-dollar bills in front of him. He rearranges the bills first in groups of 2, then 3, and 4, 5, 6, 7, 8, 9, 10 and 11. In all cases except the last, one bill is left over. In the last case, no bill is left over.

How big is the original pile?

Note that we cannot use the CRT directly: the moduli are not coprime.

But $x = 1 \bmod 8$ implies $x = 1 \bmod 4$ and $x = 1 \bmod 2$.

Moreover, $x = 1 \bmod 9$ implies $x = 1 \bmod 3$. And both together imply $x = 1 \bmod 6$.

Similarly we can drop the condition modulo 10.

So, the whole system boils down to

$$\begin{aligned}x &= 1 \pmod{m} \text{ where } m = 5, 7, 8, 9 \\x &= 0 \pmod{11}\end{aligned}$$

We have $m = 27720$ and

$$(4, 3, 1, 5, 1) = (5544, 3960, 3465, 3080, 2520)^{-1} \pmod{5, 7, 8, 9, 11}$$

and thus

$$\begin{aligned}x &= 4 \cdot 5544 + 3 \cdot 3960 + 1 \cdot 3465 + 5 \cdot 3080 \\&= 25201 \pmod{27720}\end{aligned}$$

In general, a solution may exist even if some of the moduli are not coprime. This is expressed in the following generalization.

Theorem (Generalized CRT)

The equations

$$x = a_i \pmod{m_i} \qquad i = 1, \dots, n$$

have a solution if, and only if, for all $i \neq j$:

$$a_i = a_j \pmod{\gcd(m_i, m_j)}$$

The solution is unique modulo $m = \text{lcm}(m_1, m_2, \dots, m_n)$.

1 **Basic Arithmetic**

2 **Modular Arithmetic**

3 **Chinese Remainder**

4 **Towards Algebra**

In the CRT example from above we had

$$11 + 8 = 4 \bmod 15 \qquad (2, 1) + (2, 3) = (1, 4) \bmod (3, 5)$$

$$11 \cdot 8 = 13 \bmod 15 \qquad (2, 1) \cdot (2, 3) = (1, 3) \bmod (3, 5)$$

It looks like

$$f(x + y) = f(x) + f(y)$$

$$f(x \cdot y) = f(x) \cdot f(y)$$

Here we are joyfully abusing notation, we do not distinguish between the $+$ operation in \mathbb{Z}_{15} and its counterpart in $\mathbb{Z}_3 \times \mathbb{Z}_5$.

We could write something like

$$f(x +_{15} y) = f(x) +_{3,5} f(y)$$

or even

$$f(x +_{\mathbb{Z}_{15}} y) = f(x) +_{\mathbb{Z}_3 \times \mathbb{Z}_5} f(y)$$

but that's awful to look at.

It's just operator overloading, no problem for a CS major.

Here is the classic historical example of such a map:

$$\log : \mathbb{R}^+ \rightarrow \mathbb{R}$$

which translates multiplication into addition (next week: a group isomorphism from $\langle \mathbb{R}^+, \cdot, 1 \rangle$ to $\langle \mathbb{R}, +, 0 \rangle$).

We can compute products of (positive) reals by

$$x \cdot y = e^{\log x + \log y}$$

Makes a huge difference: $O(k)$ plus table lookup rather than $O(k^2)$ where k is the number of decimal digits.

Of course, we have to compute a logarithm table first—but only once. It took John Napier some 20 years to construct such a table in the early 1600s.