Some Number Theory

Barry Monson, UNB

1 The Ring of Integers

1. The set $\mathbb{Z}$ of integers comes with two *closed* operations. Any two integers a, b have

   - a *sum* $a + b$ which is also an integer; and
   - a *product* ab which is also an integer.

   There are special integers for each operation: 0 is the *additive identity* and 1 is the *multiplicative identity.* Moreover, every integer a has an *additive inverse* (or *negative*) $-a$.

   Since the familiar rules of arithmetic hold for addition (and its close cousin subtraction) and multiplication, we say that $\mathbb{Z}$ is a *ring.*

   However, not every integer has a multiplicative inverse (reciprocal) which is itself an integer; for example, 3 is an integer but $\frac{1}{3}$ is not. Because of this, $\mathbb{Z}$ is not a *field.*

   Another way to phrase this is to say that division of integers is not a closed operation. That deficiency is, however, easily repaired by expanding the integers to the rational numbers $\mathbb{Q}$, which do form a field.

   Not every ring admits such a repair job. An example which we will soon encounter is the residue class ring

   $$\mathbb{Z}_6 = \{0, 1, 2, 3, 4, 5\} ,$$

   equipped with modular addition, subtraction and multiplication. Recall that the convenient symbols $0, 1, \ldots, 5$ are no longer integers; instead they denote residue classes. Thus $3 + 4 = 2$, and $2 \cdot 3 = 0$, even though neither 2 nor 3 equals 0. We say that 2 and 3 are *zero-divisors* in $\mathbb{Z}_6$.

2. For the record, here is a definition for *commutative rings.* Recall that a closed operation on a set R is the usual sort of operation which returns a result in the *same* set. Thus, the dot product on the set $\mathbb{R}^3$ is not closed, since $\mathbf{u} \cdot \mathbf{v}$ is a scalar, not a vector in $\mathbb{R}^3$.

   Definition 1.1. *A* commutative ring *is a set R equipped with two closed operations on R, typically called addition $a + b$ and multiplication ab, satisfying these familar properties:*

   Concerning addition: *For all $a, b, c \in R$,*

   (a) (associative law) $(a + b) + c = a + (b + c)$

   (b) (commutative law) $a + b = b + a$

   (c) (zero) there is a special element 0 such that $0 + a = a$

(d) (negatives) each a has its own special negative, denoted $-a$, such that

$$a + (-a) = 0$$

Concerning multiplication: *For all $a, b, c \in R$,*

(e) (associative law) $(ab)c = a(bc)$

(f) (commutative law) $ab = ba$

(g) (identity) there is a special element 1 such that $1a = a$

Linking the operations: *For all $a, b, c \in R$,*

(h) (distributive law) $a(b + c) = ab + ac$

Remarks. There are many more familiar, and not so familar, algebraic properties; but these must be proved from the above axioms. For example, it can (and must!) be proved that

$$(-1)a = -a, \text{ and } 0a = 0, \text{ for all } a \in R.$$

As a rule, your basic algebraic instincts will carry you forward. Naturally one can learn caution only after experiencing some unpleasant surprises!

Subtraction is a subsidiary operation, defined in terms of what we already have. By definition,

$$b - a := b + (-a)\ .$$

3. There is a huge variety of different kinds of rings, quite unlike the integers $\mathbb{Z}$. Of course, the rationals $\mathbb{Q}$, the reals $\mathbb{R}$ and the complex numbers $\mathbb{C}$ are also rings. But they have the particular virtue of being fields, in which division is possible:

   Definition 1.2. *A* field *is a commutative ring R in which every* non-zero *element a has a multiplicative inverse, denote $1/a$ or a^{-1}, and satisfying*

   $$a(1/a) = 1\ .$$

   Thus division is possible in a field. It, too, is a subsidiary operation. By definition, for $a \neq 0$,

   $$\frac{b}{a} := b(1/a)\ .$$

4. **Aside**. Strictly speaking we have described a commutative ring with unit element 1. Some rings do not have a 1. There are lots of other ways of varying the requirements to produce new and interesting kinds of objects.

5. Even though most integers do not have reciprocals, it is still very interesting to study divisibility, factorization and the like. Indeed, that is a central purpose of number theory.

   Definition 1.3. *Suppose a and b are integers. We say a* divides *b and write*

   $$a|b$$

   if $b = az$, for some integer z. Of course, we also say that a is a factor *of b and that b is a* multiple *of a, etc.*

When discussing factors and divisibility, multiplication by the integer *unit* -1 does no harm:

$$b = (-1)a \cdot (-1)z = (-a)(-z) \; .$$

Because of this we can focus on the positive integers, or natural numbers $\mathbb{N} = \{1, 2, 3, 4, \ldots\}$.

6. Indeed, we can graph the *divisibility lattice* on $\mathbb{N}$. To do this we lay out $\mathbb{N}^2$, the set of all ordered pairs of natural numbers (a, b), but shade in only those positions in which $a|b$:

   Thus divisibility, which is a relation between certain natural numbers (or more generally integers), can be viewed in the language of set theory as a subset

   $$\mathcal{R} \subset \mathbb{N} \times \mathbb{N}$$

   We now have a precise way of defining just what a relation 'really is'. Indeed, 'functions' are a special case of this. However, in practice we merely take comfort in the fact that all this has been done, and (as humans) think more intuitively.

   It is quite another matter to decide how a computer should deal with relations and functions!

7. Have a look at the numbers to the left of rows which contain exactly two shaded dots. We have found the prime numbers

   $$\mathbb{P} = \{2, 3, 5, 7, 11, 13, 17, 19, 23, \ldots\} \qquad (1)$$

Definition 1.4. *A positive integer p is* prime *if it has exactly two positive divisors,* namely, 1 and p itself.

8. Our diagram above is really a variant of the famous *Sieve of Erathosthenes* (ca. 230 BCE). Here we sieve out the primes from the set $\{2, \dots, 24\}$:

   $\boxed{2}, \boxed{3}, \cancel{4}, \boxed{5}, \cancel{6}, \boxed{7}\ \cancel{8}, \cancel{9}, \cancel{1}0, \boxed{11}, \cancel{1}2, \boxed{13}, \cancel{1}4, \cancel{1}5, \cancel{1}6, \boxed{17}, \cancel{1}8, \boxed{19}, \cancel{2}0, \cancel{2}1, \cancel{2}2, \boxed{23}, \cancel{2}4$

   Clearly this is the sort of thing a computer is made to do; just how to make Gap perform the trick is another matter (see below).

   We can say a little about when the sieving must stop if we want to find the primes in $\{2, 3, 4, \dots, n\}$, for some positive integer n. Recall that a positive integer z is *composite* if it has two proper factors, say

   $$z = ab\ ,$$

   where $a > 1$ and $b > 1$. Thus the composite positive integers are $\{4, 6, 8, 9, 10, \dots\}$. We can easily prove that

   Proposition 1.1. *A composite integer $z > 1$ must have a prime factor $p \leqslant \sqrt{z}$.*

   Meaning. Sieving will stop when we reach $\lfloor \sqrt{n} \rfloor$. Nevertheless, when n becomes quite large, sieving becomes ‘computationally slow’. You can test this using Gap’s version of the Sieve:

```
sieve:=function(n) local numbers,p,primes,m;
numbers:=[2..n];# this is the range of all integers from 2 to n
# the programme sieves out the primes from this range
# [ 2 .. n ]
primes:=[]; # we must have an empty box in which we can put things!
#
 for p in numbers do
Add(primes,p);
for m in numbers do
if m mod p = 0 then
Unbind(numbers[m-1]);
fi;
od;
od;
Print("In the interval [2..",n,"] there are ",Size(primes)," primes.","\n");
Print("They are ","\n");
return primes;
end; # end of function sieve
```

   This function was adapted from the Gap tutorial at

   `http://www.gap-system.org/Manuals/doc/htm/tut/CHAP003.htm#SECT005`

9. **How many primes are there?**

It could be that all integers past a point are sifted out, i.e. that there are no primes left over past a certain, presumably big, positive integer B. It could be; but in fact Euclid (300 BCE) showed 'No, that's not the way it is!'

For if there were no primes after the big integer B, then there would be only finitely many primes, say L of them. We could write these in order as follows:

$$p_1 = 2,\ p_2 = 3,\ p_3 = 5,\ p_4 = 7,\ p_5 = 11,\ \ldots, p_L\ ,$$

where p_L is the last and biggest prime. Since there are only finitely many such primes, we can (as Euclid observed) multiply them out and add 1 to get the integer

$$a = p_1 p_2 \cdots p_L + 1\ .$$

Yes, a is presumably huge; but still it is finite and we can think about it. Now every integer, including a, can be factored into primes with enough patience (see Theorem 1.2 below). But each prime factor of a must appear on our exhaustive list, so at least we have some p_j dividing a. Thus for some integer q we get $a = qp_j$ so that

$$1 + (p_1 \cdots p_{j-1} p_j p_{j+1} \cdots p_L) = a = p_j q\ ,$$

and thus

$$p_j(q - p_1 \cdots p_{j-1} p_{j+1} \cdots p_L) = 1\ .$$

The term in brackets is some integer, so that p_j is a factor of 1. But this means $p_j = \pm 1$, a contradiction.

Theorem 1.1. Euclid's Theorem on the infinity of primes. *There are infinitely many prime numbers.*

10. It is easy to believe, and not so hard to prove by induction, that every integer $n \geqslant 2$ can be factored into primes. In fact, up to a reordering of the terms, there is only one way to do this for a given integer:

Theorem 1.2. The Fundamental Theorem of Arithmetic. *Every integer $n \geqslant 2$ can be uniquely written as a product of primes, with the prime factors written in non-decreasing order.*

We are familiar with this. Of course, we usually group repeated primes using exponential notation:

$$2232 = 2 \cdot 2 \cdot 2 \cdot 3 \cdot 3 \cdot 31 = 2^3 3^2 31\ .$$

Here is a relevant Gap session:

```
gap> FactorsInt(2232);
[ 2, 2, 2, 3, 3, 31 ]
gap> # For more easy reading try:
gap> PrintFactorsInt(2232);Print("\n");
2^3*3^2*31
```

11. It is easy to manufacture a Gap routine which uses Euclid's idea to generate primes. However, the method does not generate all primes and produces what primes it does in a seemingly 'random' order. Moreover, this method is excruciatingly slow. The reason for this is that here we find primes by *factoring* composite integers (like a in the proof of Theorem 1.1 above); but as far as we know, factoring is 'computationally very hard'. Indeed, this supposition is used to justify the supposed security of computer communications systems, including that used between an ATM and your bank!

    Anyway, here is a Gap version of Euclid's idea:

```
gap> # We start with the first prime we know and put it in a list:
gap> pr:=[2];
[ 2 ]
gap> # Now we produce several primes using Euclid's idea.
gap> for j in [1..10] do x:=Factors(Product(pr)+1)[1];Append(pr,[x]);od;
#I  IsPrimeInt: probably prime, but not proven: 38709183810571
#I  IsPrimeInt: probably prime, but not proven: 420743244646304724409
gap> pr;
[ 2, 3, 7, 43, 13, 53, 5, 6221671, 38709183810571, 139, 2801 ]
```

    What we are doing is selecting out the smallest prime factor of $1 + p_1 \cdots p_k$ as the newest prime p_{k+1}.

    In fact, Gap must be using some probabalistic argument to assess whether large integers like 38709183810571 are prime. Presumably we can bypass that uncertainty; but then we might send the computer off for hours of work!

12. Here is a horribly slow (but fail-safe!) routine to test whether a positive integer n is prime. It checks to see whether each number b from 2 to $\lfloor\sqrt{n}\rfloor$ is a factor of n. That is easy; just use long division (see below) and check whether the remainder is 0.

    So repeat this for each $b \in [2..\lfloor\sqrt{n}\rfloor]$. If we ever get remainder 0, then n is composite. Otherwise, n is prime:

```
crudefact:=function(n) local b;
for b in [2..RootInt(n)]  # RootInt(n) is the square root
             # of n (rounded down if necessary). Also,
             # n mod b is the remainder when n is divided by b
do
if (n mod b) = 0 then Print("The integer ",n," is composite.","\n");return;
else fi;
od;
Print("The integer ",n," is prime.","\n");
end; # end of function crudefact:
```

    Even this tends to work well for a 10 or 15 digit number n, since a 'randomly chosen' n of that magnitude will not be prime. However, my computer will start to struggle to make a determination, if by bad luck n does happen to be a prime of that magnitude.

2 Long division and modular arithmetic

1. We will reconsider how it is that we 'divide' an arbitrary integer z by a particular positive integer d. To illustrate matters we take $d = 4$; but it will be clear that our reasoning works for

   $$\text{any particular integer } d \geqslant 1.$$

2. Now lay out the integers in d columns, starting with 0 on the left, and returning to the first column after d steps. Thus we start by writing

   $$0, 1, \ldots (d-1)$$

   in what I'll call the *remainder row*; then we continue the display in a natural way. Here is what we get when $d = 4$:

   $$\begin{array}{rrrr} \vdots & \vdots & \vdots & \vdots \\ -12 & -11 & -10 & -9 \\ -8 & -7 & -6 & -5 \\ -4 & -3 & -2 & -1 \\ 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \\ \vdots & \vdots & \vdots & \vdots \end{array} \tag{2}$$

   Notice that the displayed integers increase by 1 for each step to the right or down, 'wrapping around' when we hit the end of a row. (If you are reading this online, the remainder row appears in red.)

   We will label the columns by the integer r occurring in the remainder row. Thus the left-most column corresponding to $r = 0$ contains all the multiples of d.

   We will label rows by the integer q, taking $q = 0$ for the remainder row, with q increasing by 1 for each row-step down, hence, of course, decreasing by 1 for each row-step up. In other words, the row labelled q begins with the integer qd.

   This means that q is the 'number of d's' found in any integer z in that row. For example, we can take $q = 3$ copies of $d = 4$ out of $z = 13$. How much is left over? Clearly the remainder $r = 1$ is the entry to be found where the remainder row crosses the column containing $z = 13$. In brief,

   $$13 = 3 \cdot 4 + 1 \; .$$

   We have the essence of long division!

   Likewise, $12 = 3 \cdot 4 + 0$ and $15 = 3 \cdot 4 + 3$. Returning to the remainder row, we see that $2 = 0 \cdot 4 + 2$, indicating that $d = 4$ does not go into $z = 2$ and merely leaves the remainder 2.

Let us continue upward into the negative integers, taking care to continue the pattern. Here we must be careful. For example, we find $z = -10$ in the row beginning $-12 = (-3) \cdot 4$. Thus we should agree that 4 goes into -10 a total of $q = -3$ times, with remainder $r = 2$:

$$-10 = (-3) \cdot 4 + 2 \ .$$

That way all integers behave uniformly and give a non-negative remainder from 0 to $d - 1$. This is a very useful convention.

Since every integer z fits into such an array in exactly one position, we have more or less proved

Theorem 2.1. The division algorithm. *Suppose the integer $d \geqslant 1$. Then for each integer z there exist a unique* quotient *q and* remainder *r such that*

$$z = qd + r, \quad \textit{where} \ \ 0 \leqslant r \leqslant d - 1 \ .$$

3. The grade school algorithm for long division implements this process in a familiar and efficient way. The corresponding Gap commands are illustrated here:

```
gap> z:=-10; d:=4;
-10
4
gap> # for the remainder r we use the command
gap> #    z mod d
gap> r:=z mod d;
2
gap> # Thus the quotient must be
gap> q:=(z-r)/d;
-3
gap> z = q*d+r;
true
```

There are related special Gap commands – QuoInt(z,d) and RemInt(z,d) – which unfortunately do not give the quotient and remainder our way when z is negative. These peculiar functions are there for use in other routines; for simplicity, we avoid using them.

4. It is instructive, but less efficient, to programme Gap to do long division by mimicking our reasoning for the infinite array (2) above. Here is one attempt:

   ```
   # Goal: model elementary division of integers
   # as an illustration of recursive programming.
   # The first function ‘divplus’ handles positive integers only.
   # We divide a positive integer n by a positive integer d.
   divplus:=function(n,d) local q,r;
   if d<0 then Print("bad divisor d = ",d,"\n");
   else
   q:=0;r:=n;
   if r<d then return [q,r];
   else return [1,0]+divplus(n-d,d);# this amounts to stepping up
            #one row in the integer grid. Remainder is unchanged;
            # but quotient is decreased by 1.
   fi;
   fi;
   end;# end function divplus
   #
   # The next function for general integers calls the previous function.
   divint:=function(n,d) local a,b,c;
   if d<=0 then Print("bad divisor d = ",d,"\n");
   else if n<0 then a:= divplus(-n,d);
   if a[2]=0 then return -a;
   else return (-a)+[-1,d];fi;
   #
   else return divplus(n,d);
   fi;
   fi;
   end;#end function divint
   #
   ```

   These implementations will stuggle for big inputs because of the recursion in divplus. The are for understanding not efficiency. Another deficiency is that the routines do not make essential use of decimal notation for integers.

5. Let's study more carefully the integer array in (2). Every integer appears in exactly one of the d columns (in this case $d = 4$). This gives $d = 4$ subsets which partition $\mathbb{Z}$.

   Definition 2.1. *A* partition *of a set A is a collection of non-empty, mutually disjoint subsets B_j whose union is A.* (The subscript j runs over some index set J.)

   In (2), the set A is the integers $\mathbb{Z}$, and there are four subsets in the partition of interest to us. Just assemble each column into its own subset:

$$\begin{aligned} B_0 &= \{\ldots, -8, -4, 0, 4, 8, \ldots\} \\ B_1 &= \{\ldots, -7, -3, 1, 5, 9, \ldots\} \\ B_2 &= \{\ldots, -6, -2, 2, 6, 10, \ldots\} \\ B_3 &= \{\ldots, -5, -1, 3, 7, 11, \ldots\} \end{aligned}$$

   Remark. Thus the index set $J = \{0, 1, 2, 3\}$ is finite, and in fact corresponds exactly to the remainder row. The sets B_r are called *residue classes*; one often writes $\overline{r}$ as a convenient abbreviation for B_r.

   In other applications there could be infinitely many blocks B_j in the partition, so that J would be infinite.

   But for divisor d, it is clear that the d columns give exactly d residue classes B_r, one for each of the possible entries r in the remainder row

$$0, \ldots, d - 1 .$$

6. What do the elements in class B_r have in common? Two answers are immediate from the picture in (2):

   Theorem 2.2. *Two integers z and w lie in the same column*

   - *if and only if z and w have the same remainder r upon division by d (and r then labels their column);*
   - *if and only if their difference $z - w$ is a multiple of d:*

$$z - w = qd .$$

   Definition 2.2. *Suppose d is a positive integer and $z, w \in \mathbb{Z}$. We write*

$$z \equiv w \pmod{d}$$

   if $z - w$ is a multiple of d (equivalently, z and w have the same remainder upon division by d).

   We often read $z \equiv w \pmod{d}$ as 'z is congruent to w modulo d'.

7. It is remarkable and very useful that the symbol $\equiv$ behaves rather like the ordinary equality symbol '='. More precisely, it has the three natural looking properties described in the next

   Theorem 2.3. *For each fixed $d \geqslant 1$, the relation '$\equiv \pmod{d}$' is an* equivalence relation *on $\mathbb{Z}$. That is, for all $z, w, v \in \mathbb{Z}$, we have*

   (a) the reflexive property*: $z \equiv z \pmod{d}$.*

   (b) the symmetric property*: If $z \equiv w \pmod{d}$, then $w \equiv z \pmod{d}$.*

   (c) the transitive property*: If $z \equiv w \pmod{d}$ and $w \equiv v \pmod{d}$, then $z \equiv v \pmod{d}$.*

   Proof. Try it yourself! In part (c), for example, we assume $z - w = qd$ and $w - v = yd$ for certain unknown integers q and y. But then add these equations to get $z - v = (q + y)d$. Done! (Really all we are saying is that if z and w lie in the same column in (2) and also w and v lie in the same column, then z and v lie in the same column. □

   Remark. Intuitively, this means you can play nearly as fast and loose with '$\equiv \pmod{d}$' as with '='.

   At a more abstract level, we begin to see that any partitioning of a set A lets us think of the blocks in terms of a new variant of 'equality'.

8. There is much more to the algebra of congruences. The interactions between congruences and the ring operations on $\mathbb{Z}$ are most useful tools in number theory and algebra. The following results are easily proved:

   Theorem 2.4. *For all $z, w, a, b \in \mathbb{Z}$ (integers!), and a fixed modulus $d \geqslant 1$, we have:*

   (a) If $z \equiv w \pmod{d}$ and $a \equiv b \pmod{d}$, then $a + z \equiv b + w \pmod{d}$. Likewise, $a - z \equiv b - w \pmod{d}$.

   (b) If $z \equiv w \pmod{d}$ and $a \equiv b \pmod{d}$, then $az \equiv bw \pmod{d}$. In particular,

   (c) If $z \equiv w \pmod{d}$, then $az \equiv aw \pmod{d}$.

   We should not be surprised, however, that we cannot 'divide' one congruence by another, at least without a more careful look at what that would entail. For a different take on much the same issues, consult Theorems 2.5 and 2.6 below.

9. **The idea of modular arithmetic a.k.a. clock arithmetic**.

   (a) If it is Wednesday today, then 300 days from now it will be Tuesday, since

   $$300 = 42(7) + 6\ .$$

   Likewise, 300 days ago it must have been Thursday, since

   $$-300 = (-43)(7) + 1\ .$$

   For many mathematical purposes, what counts for an integer z is its remainder (or residue) after division by a fixed positive modulus d. (For week-work, $d = 7$.) We can define a new arithmetic on the standard residues by replacing the ordinary sum and product of integers by the remainder modulo d. Basically this is clock arithmetic.

(b)

Definition 2.3. *The residue class ring $\mathbb{Z}_d$.*
For a fixed integer modulus $d \geqslant 1$, we denote the set of standard remainders by

$$\mathbb{Z}_d := \{0, \ldots, d-1\} \ .$$

On this set we define two closed operations, temporarily denoted $\oplus$ and $\odot$. For $a, b \in \mathbb{Z}_d$ let

- *$a \oplus b := r$, if as ordinary integers $a + b = qd + r$, with remainder r satisfying $0 \leqslant r \leqslant d-1$ as usual.*
- *$a \odot b := r$, if as ordinary integers $a \cdot b = qd + r$, with remainder r satisfying $0 \leqslant r \leqslant d-1$ as usual.*
- *$-a := r$, if as ordinary integers $-a = qd + r$, with remainder r satisfying $0 \leqslant r \leqslant d-1$ as usual.*

These operations are closed, since by definition each always produces a result back in the set $\mathbb{Z}_d := \{0, \ldots, d-1\}$.

(c) In brief, then, we operate on $\mathbb{Z}_d$ by replacing, wherever possible, any integer z by its remainder upon division by d. This remainder is often called the *residue* of z (mod d).

It is not too hard to prove that under these new operations we still have the normal rules of arithmetic, as detailed in Definition 1.1. In short, we have

Theorem 2.5. *$\mathbb{Z}_d$ with operations $\oplus$ and $\odot$ is a commutative ring.*

Proof. We must check each property in Definition 1.1, which is easy. As a test case, let's prove that multiplication is associative.

For any and all $a, b, c \in \mathbb{Z}_d$ our goal is to show $(a \odot b) \odot c = a \odot (b \odot c)$. We need to give integer names to the relevant factors.

First of all, $a \odot b = r$ means simply that $ab = qd + r$, where $0 \leqslant r \leqslant d-1$. Thus $(a \odot b) \odot c = r \odot c = s$, means $rc = q_1 d + s$, again with $0 \leqslant s \leqslant d-1$. Thus the ordinary integer

$$(ab)c = (qc + q_1)d + s = q_2 d + s \ ,$$

where $q_2 = qc + q_1$. But the division algorithm Theorem 2.1 produces a unique quotient q_2 and remainder s regardless of the involved process which got us to this result. Since $a(bc) = (ab)c$ as ordinary integers, the same sort of calculation has to give $a \odot (b \odot c) = s$, as well. In other words, we must always have

$$(a \odot b) \odot c = a \odot (b \odot c) \ .$$

The same sort of verification will work for all ring properties. □

(d) **Calculation in $\mathbb{Z}_d$ in practice**. We usually abuse notation and revert from the tiresome $a \oplus b, a \odot b$ to the usual $a + b, ab$.

For example, in $\mathbb{Z}_6$ we will simply write

$$1 + 3 = 4, \ 2 + 5 = 1, \ 2 \cdot 3 = 0, \ 4 \cdot 4 = 4 \ , -1 = 5 \ .$$

In this context it is useful to interpret '=' as '$\equiv \pmod 6$'. And strictly speaking the symbols $6, 7, -13$ make no sense, although we cannot go wrong by agreeing that

$$6 = 0, \quad 7 = 1, \quad -13 = -1 = 5 \ .$$

It is important to note a built-in deficiency in $\mathbb{Z}_6$. The crucial blemish is that 2, for instance, has no multiplicative inverse (what we would want to label 1/2) in $\mathbb{Z}_6$. For suppose some $a \in \mathbb{Z}_6$ did the job. This means $2 \cdot a = 1$. But then

$$3 = 3 \cdot 1 = 3 \cdot (2 \cdot a) = (3 \cdot 2) \cdot a = 0 \cdot a = 0 \ ,$$

a contradiction! (3 and 0 really are different remainders upon division by 6.) Consulting Definition 1.2, we conclude that $\mathbb{Z}_6$ is not a field.

The core problem here is that the modulus $d = 6 = 3 \cdot 2$ is composite. Thus, the ring $\mathbb{Z}_d$ cannot be a field if d is a composite integer. But what happens if d is prime?

(e)

Theorem 2.6. *$\mathbb{Z}_d$ is a (finite) field if d is prime.*

Proof. Fix any $a \neq 0$ in $\mathbb{Z}_d$. (We're supposing d is a prime.) In other words, $1 \leqslant a \leqslant d-1$. Now put the multiples of a in the list

$$M = [0 = 0a, 1a, 2a, 3a, \ldots, (d-2)a, (d-1)a] \ ,$$

all taken $\pmod d$. Could two of these be equal? Well suppose $ia = ja$ for $0 \leqslant i \leqslant j \leqslant d-1$. This means $(j-i)a = 0$ in $\mathbb{Z}_d$, which in turn means by definition that $(j-i)a$ has remainder 0 on division by d. In other words, $(j-i)a$ is a multiple of the prime number d. But we know (and will later prove in Theorem 3.2(b)) that a prime like d has the special property that it must be a factor of either $j-i$ or of a, if it divides their product. But this is impossible since a lies between 1 and $d-1$, as does $j-i$, unless $i = j$.

We conclude that the elements listed in M are distinct. But there are d such elements, so we must simply have a rearrangement of $[0, 1, \ldots, d-1]$, with 0 matching up in the first slot. Thus 1 must appear somewhere else in M! Hence there exists an integer $i \neq 0$ such that $i \cdot a = 1 \pmod d$. This i will serve duty as $1/a$.

We now see why every $a \neq 0$ has a multiplicative inverse $1/a \in \mathbb{Z}_d$. We conclude that the ring $\mathbb{Z}_d$ is really a field. □

3 The greatest common divisor

1. Any two integers a and b have some common divisors, at least $+1$ and -1:

$$a = a \cdot 1 \text{ and } b = b \cdot 1, \text{ so } 1|a \text{ and } 1|b \ .$$

On the other hand, for $\underline{\text{any}}$ integer n, no matter how large, we have $n|0$ and $n|0$, so that 0 and 0 do not have a *greatest* common divisor. (Or perhaps we might call it ∞.)

Finally, we observe that if say $b \neq 0$ and $k|b$, then $k \leqslant |b|$, so that there is a largest possible divisor for any non-zero integer, namely the positive integer $|b|$.

These thoughts motivate

Definition 3.1. *Suppose $a, b \in \mathbb{Z}$ are not both* 0*. Then the* greatest common divisor *of a, b is the largest integer g which divides both a and b. We write*

$$g = \gcd(a, b) \ .$$

Notice that $gcd(a, b)$ is then a positive integer, namely at least 1. The minimal case where we hit 1 is interesting and familiar:

Definition 3.2. *Two integers $a, b \in \mathbb{Z}$ are said to be* relatively prime *if* $\gcd(a, b) = 1$.

For example, we say that a rational number $\frac{a}{b}$ is in *lowest terms* if $\gcd(a, b) = 1$. We often say, a little abusively, that a and b have 'no common factor', by which we really mean that $+1$ and -1 are the only common divisors of a and b.

2. The *Euclidean Algorithm* lets us compute $g = \gcd(a, b)$ quickly and furthermore write g as a $\mathbb{Z}$-linear combination of a and b. This means we can find integers u, v such that

$$g = ua + vb \ .$$

In fact, we then have

$$\mathbb{Z}g = \mathbb{Z}a + \mathbb{Z}b \ .$$

This is actually a statement concerning *ideals* in the ring $\mathbb{Z}$.

For example,

$$6 = \gcd(150, 114)$$

and

$$6 = (-3)150 + (4)114 \ .$$

3. To ease our proof, we need to understand some facts about divisibility:

   Proposition 3.1. *For integers x, y, z we have*

   (a) $\gcd(x, y) = \gcd(y, x)$.

   (b) $\gcd(x, 0) = |x|$, *if* $x \neq 0$.

   (c) $\gcd(x, y) = \gcd(x, y - zx)$, *for any integer* z.

   Proof. Part (a) is clear, since when talking of *common* divisors, the order in which we consider x or y is irrelevant. We have already commented on part (b).

   In part (c) we need only observe that the two numbers x, y have the same set of common divisors as the two numbers $x, y - zx$. For if $x = q_1 d$ and $y = q_2 d$, so that d is a common divisor of x and y, then $y - zx = (q_2 - zq_1)d$. Since $q_2 - zq_1$ is some integer, d is also a divisor of $y - zx$. The reasoning is reversible, since $y = (y - zx) - (-z)x$. □

4. Here then is the procedure. To make the recursive notation easier, we start off by conveniently relabelling a and b as x_1 and y_1. We furthermore initialize integer vectors $\mathbf{w}_{-1}$ and $\mathbf{w}_0$ whose two components will ultimately be transformed into the coefficients u, v in

   $$\gcd(a, b) = ua + vb \ .$$

   You can omit the **w**s if you don't need the $\mathbb{Z}$-linear representation.

   Theorem 3.1. Euclidean Algorithm.
   Suppose a and b are positive integers.

   - Initialize: let $x_1 := a,\ y_1 := b,\ \mathbf{w}_{-1} = [1, 0],\ \mathbf{w}_0 = [0, 1]$.
   - Start: by dividing x_1 by y_1:

     $$x_1 = q_1 y_1 + r_1, \text{ where } 0 \leqslant r_1 < y_1 \ .$$

     Let $\mathbf{w}_1 = \mathbf{w}_{-1} - q_1 \mathbf{w}_0$.
   - Repeat as long as $r_j > 0$:

     Update: let $x_{j+1} := y_j$, $y_{j+1} := r_j$ and divide x_{j+1} by y_{j+1} to get

     $$x_{j+1} = q_{j+1} y_{j+1} + r_{j+1}, \text{ where } 0 \leqslant r_{j+1} < r_j = y_{j+1} \ .$$

     Also let $\mathbf{w}_{j+1} = \mathbf{w}_{j-1} - q_{j+1} \mathbf{w}_j$.
   - Stop: when the remainder hits 0, say at step $k + 1$, meaning $r_{k+1} = 0$.
   - Return $\gcd(a, b) = r_k$ and $[u, v] = \mathbf{w}_k$.

   Proof. The sequence of non-negative remainders is decreasing:

   $$0 \leqslant \ldots r_3 < r_2 < r_1 < y_1 = b \ .$$

   Thus at some step, say $k + 1$, we must get remainder 0 and stop.

   Now since $r_{k+1} = 0$ we have $x_{k+1} = q_{k+1} y_{k+1} + 0 = q_{k+1} y_{k+1}$, so that $r_k = y_{k+1}$ is a divisor of x_{k+1}. Hence $r_k = \gcd(x_{k+1}, y_{k+1})$.

But for each j, $x_{j+1} = y_j$ and $y_{j+1} = r_j = x_j - q_j y_j$. Now Theorem 3.1(c),(a) imply that

$$\gcd(x_{j+1}, y_{j+1}) = \gcd(y_j, x_j - q_j y_j) = \gcd(x_j, y_j) \; .$$

Tracing all the way back to the start we find that

$$r_k = \ldots = \gcd(x_1, y_1) = \gcd(a, b) \; .$$

Finally we must deal with the $\mathbf{w}_j$s. We start off with $\mathbf{w}_1 = [1, 0] - q_1[0, 1] = [1, -q_1]$, and note that $r_1 = x_1 - q_1 y_1 = 1a - q_1 b = \mathbf{w}_1 \cdot [a, b]$ (think dot product). This starts an induction in which we assume $r_j = \mathbf{w}_j \cdot [a, b]$. Then

$$\begin{aligned}
\mathbf{w}_{j+1} \cdot [a, b] &= (\mathbf{w}_{j-1} - q_{j+1}\mathbf{w}_j) \cdot [a, b] \text{ (by defn.)} \\
&= \mathbf{w}_{j-1} \cdot [a, b] - q_{j+1} \, \mathbf{w}_j \cdot [a, b] \text{ (props. of } \cdot) \\
&= r_{j-1} - q_{j+1} r_j \text{ (by inductive hypoth.)} \\
&= y_j - q_{j+1} r_j \text{ (by defn.)} \\
&= x_{j+1} - q_{j+1} y_{j+1} \text{ (by defn.)} \\
&= r_{j+1}.
\end{aligned}$$

This completes the induction. (There is a small inductive lapse here that I will let you fill in yourself.) In particular, we get $\gcd(a, b) = r_k = \mathbf{w}_k \cdot [a, b]$. □

5.

Corollary 3.1. *The* 2×2 *matrix*

$$A = \begin{bmatrix} \mathbf{w}_k \\ \mathbf{w}_{k+1} \end{bmatrix}$$

(with rows $\mathbf{w}_k, \mathbf{w}_{k+1}$*) is* unimodular*: it has integer entries and determinant* ± 1*. Moreover,*

$$A \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \gcd(a, b) \\ 0 \end{bmatrix} \; .$$

Proof. Note that $\mathbf{w}_{k+1} \cdot [a, b] = r_{k+1} = 0$. □

6. **An important application to relatively prime integers**.

Recall that two integers a and b are *relatively prime* (or *coprime*) if $\gcd(a, b) = 1$.

Theorem 3.2. *(a) Suppose a and b are relatively prime integers and that $a|(bc)$. Then a must divide c (i.e. $a|c$).*

(b) Suppose the prime number p divides the product bc. Then p must divide either b or c (or both).

(c) Suppose d is some common divisor of a and b. Then d divides $\gcd(a, b)$.

Proof. In part (a) we assume $bc = za$ for some integer z. We are also assuming a and b are relatively prime, so from Theorem 3.1 we get (easily computed!) integers u, v such that $1 = ua + bv$. Now here is a simple but very useful trick - multiply by c to get

$$c = c \cdot 1 = uac + bcv = a(uc + zv) \ .$$

Thus a really does divide c.

In part (b), $g = \gcd(p, b)$ could only equal p or 1, being a positive divisor of the prime p. If $\gcd(p, b) = p$, then p is also a divisor of b and we have the result of the theorem. On the other hand, if $\gcd(p, b) = 1$, then by part (a) we get that p is a divisor of c.

In part (c), we have $g = \gcd(a, b) = ua + bv$; we also assume $a = q_1d$, $b = q_2d$ for some integers q_1, q_2. But then we have $g = d(uq_1 + vq_2)$, so that d really is a divisor of g. □

7. **Another important application in number theory**.

There is an extensive set of techniques for solving congruences. We will only dabble in this. Here is a central

Theorem 3.3. The Chinese Remainder Theorem.

(a) *Suppose integers $a_1, a_2 \geqslant 1$ are relatively prime. Then the system of congruences*

$$\left\{ \begin{array}{lll} x & \equiv & k_1 \pmod{a_1} \\ x & \equiv & k_2 \pmod{a_2} \end{array} \right. \qquad (3)$$

has a solution for any integers k_1, k_2. Moreover, this solution is unique modulo the product a_1a_2. In detail, all the infinitely many solutions are given by $x + ta_1a_2$, where x is one particular solution and $t \in \mathbb{Z}$.

(b) *More generally, suppose $a_1, \ldots, a_n$ are pairwise relatively prime positive integers. Then for any $k_1, \ldots, k_n \in \mathbb{Z}$, the system*

$$\left\{ \begin{array}{lll} x & \equiv & k_1 \pmod{a_1} \\ & \vdots & \\ x & \equiv & k_n \pmod{a_n} \end{array} \right. \qquad (4)$$

has a solution which is unique modulo the product $a_1 \cdots a_n$.

Proof. We prove just part (a). By Theorem 3.1 there are integers u_1, u_2 such that

$$1 = u_1 a_1 + u_2 a_2 \ .$$

Let $x = k_1(u_2 a_2) + k_2(u_1 a_1)$. Then

$$\begin{aligned} x &= k_1(1 - u_1 a_1) + k_2 u_1 a_1 \\ &\equiv k_1(1 - 0) + k_1 0 \pmod{a_1} \\ &\equiv k_1 \pmod{a_1} \ . \end{aligned}$$

Likewise $x \equiv k_2 \pmod{a_2}$. Now if $y \in \mathbb{Z}$ is any other solution, then $y \equiv k_1 \equiv x \pmod{a_1}$. Thus $y - x$ is divisible by a_1 and similarly by a_2. Say then that

$$y - x = r a_1 = s a_2 \ .$$

Thus $a_2 | (r a_1)$, so $a_2 | r$ by Theorem 3.2(a). Therefore $y - x = t(a_1 a_2)$ for some $t \in \mathbb{Z}$ and $y \equiv x \pmod{a_1 a_2}$. □