ITCS 531: Number Theory 2 - Modular arithmetic

Rob Egrot

Counting with clocks

- What time will it be in 24 hours?
- Easy to answer.
- But, there is some interesting and important mathematics behind this.
- This is a simple example of **modular arithmetic**.

From clocks to encryption

- We don't need an abstract theory to tell the time.
- But combined with prime numbers, this 'clock arithmetic' will give us RSA encryption.
- This is quite important.

Equivalence relations

- We will need a mathematical concept of 'equivalence'.

Definition 1 (Equivalence relation)

A binary relation R on a set X is an *equivalence relation* if it has the following three properties.

1. $R(x, x)$ for all $x \in X$ (reflexive).
2. $R(x, y) \iff R(y, x)$ for all $x, y \in X$ (symmetric).
3. $R(x, y)$ and $R(y, z) \implies R(x, z)$ for all $x, y, z \in X$ (transitive).

Equivalence classes

- If R is an equivalence relation on X, and $x \in X$, then $\{y \in X : R(x, y)\}$ is the **equivalence class** of x.
- We often write $[x]$ for the equivalence class of x.
- We can write e.g. $[x]_R$ when we want to make it explicit.
- Equivalence relations give us a way of grouping objects that are 'essentially the same' together.
- For example, it is a principle of monetary systems that, e.g. one \$10 bill is equivalent any other \$10 bill.
- On the other hand a photo of my family is not equivalent to a photo of your family.
- However, identical copies of the same photograph will normally be equivalent.

Example: coloured balls

Example 2

- Let X be a set of balls.
- Then 'being the same colour' is an equivalence relation on X.
- Every ball is the same colour as itself (reflexive).
- If x is the same colour as y then y is obviously the same colour as x (symmetric).
- If x and y are the same colour, and also y and z are the same colour, then clearly x and z are the same colour (transitive).

Example: friends

Example 3

- 'Being friends' is not an equivalence relation on a group of people.
- We can assume that people are friends with themselves (reflexive).
- Friendship is symmetric by definition.
- However, it's not usually transitive.

Partitions

- The equivalence classes of an equivalence relation on a set divide the set into pieces.
- We can formalize this concept with another definition.

Definition 4 (Partition)

If X is a set then a *partition* of X is a set of pairwise disjoint subsets of X whose union is equal to X.

In other words, a partition of a set divides it into pieces that don't overlap at all.

Partitions and equivalence relations

- Partitions and equivalence relations are different ways of talking about the same thing.
- The next proposition expresses half of this fact:

Proposition 5

If R is an equivalence relation on X then $\{[x] : x \in X\}$ is a partition of X.

- There is also a converse (see homework).

Proof

- Let R be an equivalence relation (definition 1).
- First show $\{[x] : x \in X\}$ satisfies definition 4.
- Must show that the union of all the equivalence classes is X.
  - We have $\bigcup_{x \in X}[x] \subseteq X$ because $[x] \subseteq X$ for all x.
  - Conversely, if $y \in X$ then $y \in [y]$ by reflexivity of R.
  - So $X \subseteq \bigcup_{x \in X}[x]$ and so $\bigcup_{x \in X}[x] = X$ as required.

Proof - continued

- Now show equivalence classes are pairwise disjoint.
  - Suppose $[x] \cap [y] \neq \emptyset$.
  - Then there is $z \in X$ with $R(x,z)$ and $R(y,z)$.
  - But then $R(z,y)$, by symmetry, and so $R(x,y)$ by transitivity.
  - By symmetry again we also have $R(y,x)$.
  - Now, using transitivity and the fact that $R(x,y)$ and $R(y,x)$ we have

$$\begin{aligned} z \in [x] &\iff R(x,z) && \text{(by definition)} \\ &\iff R(y,z) && \text{(by transitivity with } R(x,z) \text{ and } R(y,x)) \\ &\iff z \in [y] && \text{(by definition)} \end{aligned}$$

  - So $[x] = [y]$.
  - I.e. if $[x]$ and $[y]$ are not disjoint they are equal.

Modular equality

- We can now define modular arithmetic seriously.

Definition 6 (Modular equality)

Given $x, y \in \mathbb{Z}$, we say $x \equiv y \mod n$ if there is $k \in \mathbb{Z}$ with $x - y = kn$.
I.e. if the difference between x and y is a multiple of n.
We also write $x \equiv_n y$.

- So, for example, a 24 hour clock uses numbers modulo 24.
- If we add 24 to a number on the clock then we get back the same number.
- I.e., 14:00 is 'essentially the same' as 38:00, which is 'essentially the same' as 52:00 etc.

Modular equality and equivalence

- Modular equality is an equivalence relation.

Proposition 7

Let $n \in \mathbb{N}$. Then $\equiv_n$ is an equivalence relation on $\mathbb{Z}$.

Proof.

We must check each condition from definition 1. Let $x, y \in \mathbb{Z}$.

1. $x - x = 0 = 0n$, so $x \equiv_n x$.
2. If $x - y = kn$ then $y - x = -kn$, and vice versa, so $x \equiv_n y \iff y \equiv_n x$.
3. If $x - y = kn$ and $y - z = ln$, then $x - z = kn + ln = (k + l)n$, so $x \equiv_n y$ and $y \equiv_n z \implies x \equiv_n z$.

$\square$

Properties of modular arithmetic

- Suppose the number x is 'essentially the same' as x', and the number y is 'essentially the same as y'.
- We expect e.g. $x+y$ to be 'essentially the same' as $x'+y'$.
- Fortunately this is true:

Proposition 8

Suppose $x \equiv_n x'$, and $y \equiv_n y'$. Then:

1. $x + y \equiv_n x' + y'$, *and*
2. $xy \equiv_n x'y'$.
3. *For all* $k \in \mathbb{N}$, $x^k \equiv_n x'^k$.

Proof.

- For (1) suppose $x - x' = kn$ and suppose $y - y' = ln$.
- Then $(x+y) - (x'+y') = (k+l)n$. I.e. $(x+y) \equiv_n x' + y'$.
- The second part will be an exercise, and (3) follows from (2).

$\square$

More properties of modular arithmetic

Proposition 9

Let $n \in \mathbb{N}$. Then:

(1) $(x + y) + z \equiv_n x + (y + z)$ *for all* $x, y, z \in \mathbb{Z}$ *(Associativity of addition).*

(2) $(xy)z \equiv_n x(yz)$ *for all* $x, y, z \in \mathbb{Z}$ *(Associativity of multiplication).*

(3) $x + y \equiv_n y + x$ *for all* $x, y \in \mathbb{Z}$ *(Commutativity of addition).*

(4) $xy \equiv_n yx$ *for all* $x, y \in \mathbb{Z}$ *(Commutativity of multiplication).*

(5) $x(y + z) \equiv_n (xy) + (xz)$ *for all* $x, y, z \in \mathbb{Z}$ *(Distributivity).*

Proof.

Because $(x + y) + z = x + (y + z)$, we have

$$((x + y) + z) - (x + (y + z)) = 0 = 0 \times n.$$

This proves (1), the rest is similar. $\square$

When to calculate modular values

- Combining propositions 8 and 9 we can also say e.g. that $(x + y \mod n) + z \equiv_n x + (y + z \mod n)$ for all $x, y, z \in \mathbb{Z}$.
- In other words, it doesn't matter at what point we calculate remainders modulo n.
- We can wait till the end or do it as we go along.
- We will still get the same answer.

Calculations in modular arithmetic

- We can exploit properties of modular arithmetic to simplify complex seeming expressions.
- We can perform calculations with large numbers without using a computer.
- We can perform calculations with very large numbers on a computer without running out of memory.

Example 10

$$2^{345} \equiv_{31} (2^5)^{69} \equiv_{31} 32^{69} \equiv_{31} 1^{69} \equiv_{31} 1$$

- Note: It's not true that $x^y \equiv_n x^{y'}$ when $y \equiv_n y'$.
  - E.g. $5 \equiv_4 1$, but $2^5 = 32 \equiv_4 0$, and $2^1 = 2 \equiv_4 2$.

An algorithm for modular calculations

- We often want to evaluate exponentials in modular arithmetic.

- We won't always be able to makes things as easy as they are in example 10.

- But we must do better than the naive approach (i.e. calculating x^y then finding the answer mod n).

- In practical applications, x^y could be too big for our computer to handle.

- Fortunately, we can break exponentials down into small parts, so the numbers never get too large.

$$\text{If } x \equiv_n x' \text{ and } (x')^{y-1} \equiv_n z, \text{ then } x^y \equiv_n zx'.$$

- I.e. to work out $x^y \mod n$, first find $x \mod n$, then find $x(x \mod n) \mod n$ etc.

Speeding things up

- Using this method the numbers never get too big.
- But we need to perform $y-1$ multiplications, which can take a lot of time.
- We can speed up the algorithm with a trick.
- Every number can be written in binary, which represents a sum of powers of 2.
- So we can rewrite x^y so that it is a product of x to the power of various powers of 2. E.g.

$$x^{25} = xx^8x^{16},$$

- This corresponds to the fact that 25 is 11001 in binary.

The worst case run time

- For this method, in the worst case is when the binary representation of y is a string of ones.
- If l is the length of y when written in binary, we have to perform $(l-1)+(l-1)=2l-2$ multiplications.
- This is linear in the length of the binary form of y.
- With a little thought, we can turn this idea into a neat recursive function.
- This function is practical from a computational perspective.

The final algorithm

$$\exp(x, y, n) = \begin{cases} 1 \text{ if } y = 0 \\ (\exp(x, \lfloor \frac{y}{2} \rfloor), n))^2 \mod n \text{ if } y \text{ is even} \\ x(\exp(x, \lfloor \frac{y}{2} \rfloor), n))^2 \mod n \text{ if } y \text{ is odd} \end{cases}$$

- This algorithm is not mysterious.
- The key observation is that, for $y > 0$, we have

$$x^y = \begin{cases} (x^{\frac{y}{2}})^2 \text{ when } y \text{ is even} \\ x.(x^{\frac{y-1}{2}})^2 \text{ when } y \text{ is odd.} \end{cases}$$

- So, for example:

$$x^{25} = x(x^{12})^2 = x((x^6)^2)^2 = x(((x^3)^2)^2)^2 = x(((x(x)^2)^2)^2)^2 = xx^8x^{16}.$$

Example calculation

$$
\begin{aligned}
3^{25} \mod 4 &= 3(3^{12} \mod 4)^2 \mod 4 \\
&= 3((3^6 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((3^3 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((3(3 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((3 \cdot 3^2 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((27 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3((3^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(1^2 \mod 4)^2 \mod 4 \\
&= 3(1^2 \mod 4)^2 \mod 4 \\
&= 3(1^2) \mod 4 \\
&= 3
\end{aligned}
$$