Modular Arithmetic

The mathematical tool we will explore today is called *modular arithmetic.* Here is a brief summary of the points we will cover.

1. The motivating example for modular arithmetic is *clock arithmetic.* What time is it 5 hours after 10 o'clock? The answer $5 + 10 = 15$ is fine for military time, but ordinarily we would throw away the extra 12 hours, and answer 3 o'clock.

2. More generally, we say that integers a and b are *congruent modulo 12* if $a - b$ is divisible by 12. We write this as $a \equiv b \pmod{12}$. In our simple example above, $15 \equiv 3 \pmod{12}$.

   For a more interesting example, suppose we are using a twelve hour clock, and it is 7 o'clock. What will the clock read 100 hours later? To solve this problem we need to compute $7 + 100 = 107$, modulo 12. If we divide 107 by 12, we get 9, with a remainder of 11. But it is only the remainder that is relevant. We write this as $107 \equiv 11 \pmod{12}$.

   Clearly, we could work with any *modulus* m, and talk instead about $a \equiv b \pmod{m}$. For example, $15 \equiv 5 \pmod{10}$, $4 \equiv 49 \pmod{5}$, $365 \equiv 5 \pmod{30}$.

3. Notice that every integer is congruent modulo m to exactly one of the m integers $0, 1, 2, 3, \cdots, m-1$. These are just the remainders left over, when we throw away all the extra m's. Thus $653 \equiv 23 \pmod{30}$: *just throw away the extra 30's!* We are using what mathematicians call the division algorithm: $653 = (30)(21) + 23$, where 21 is the quotient, and we're really only interested in the remainder 23.

4. If $a \equiv b \pmod{m}$, then $a + c \equiv b + c \pmod{m}$. *We can add to both sides of a modular equivalence.* We can also subtract. We'll do lots of examples in class.

5. If $a \equiv b \pmod{m}$, then $ac \equiv bc \pmod{m}$. *We can multiply both sides of a modular equivalence.*

6. If m is a prime number, we can also "divide". For example, if $3x \equiv 4 \pmod{7}$, then $15x \equiv 12 \pmod{7}$. (Multiply both sides by 5.) But $15 \equiv 1 \pmod{7}$. Thus $x \equiv 12 \equiv 5 \pmod{7}$. Way cool!

7. It is not that clear where I got the 5 to use for "division" above. What is true is that 3 has a *multiplicative inverse* 5 modulo 7 precisely because 3 and 7 have no common divisors. We can actually compute the inverse 5 for 3 by division: $7 = 2 \cdot 3 + 1$, and so $1 = 7 + (-2) \cdot 3$. But then $1 \equiv (-2) \cdot 3 \pmod{7}$. Thus $-2 \equiv 5 \pmod{7}$ is our inverse.

8. More typically, we have to divide more than once. To compute the multiplicative inverse of 6 modulo 29, we must divide twice:

   $$\begin{aligned} 29 &= 6 \cdot 4 + 5 \\ 6 &= 5 \cdot 1 + 1 \end{aligned}$$

   We then solve these equations backwards for 1:

   $$\begin{aligned} 1 &= 6 + (-1)5 \\ &= 6 + (-1)(29 + 6(-4)) \\ &= 5 \cdot 6 + (-1)(29) \end{aligned}$$

   But if we take this last equation modulo 29, we see that the multiplicative inverse of 6 modulo 29 is 5.

9. For example, suppose we have the congruence $6x + 11 \equiv 5 \pmod{29}$. Add 18 to both sides: $6x \equiv 23 \pmod{29}$. Then multiply both sides by 5: $x \equiv 30x \equiv 115 \equiv 28 \pmod{29}$. We have solved our congruence!

10. The algorithmic computation of the multiplicative inverse above is clearly something a computer could be computed to do. Fortunately, I have written a little file in Excel that performs this efficiently, so that you can for the most part avoid the painful arithmetic. This file is called Euclid, and is available on the Course drive.