Congruences

One of the important notational devices used by Gauss in his *Disquisitiones Arithmeticae* (1801) was the **congruence**: where $a, b, m$ are integers and $m$ is nonzero,

$$a \equiv b \pmod{m} \iff m \mid (a - b)$$

$\iff a, b$ have the same remainder when divided by $m$

Here, $m$ is called the **modulus**. The congruence relation is a prototypical example of an equivalence relation:

**Proposition** Congruence mod $m$ is an equivalence relation (reflexive, symmetric, and transitive). //

At least as important is the fact that congruence mod $m$ is **compatible with arithmetic**.

**Proposition** If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then

1. $a + c \equiv b + d \pmod{m}$;
2. $ac \equiv bd \pmod{m}$;
3. $a^k \equiv b^k \pmod{m}$ for any positive integer $k$. //

**Proposition**

1. (Reduction) If $a \equiv b \pmod{m}$ and $n \mid m$, then $a \equiv b \pmod{n}$.
2. (Cancellation) $ac \equiv bc \pmod{m} \Rightarrow a \equiv b \pmod{\frac{m}{(c,m)}}$. 
Because congruence mod \( m \) is an equivalence relation, the integers are partitioned into equivalence classes under this relation, called more appropriately **congruence classes** mod \( m \). (Every integer belongs to exactly one congruence class mod \( m \) and no two congruence classes have anything in common.) There are exactly \( m \) congruence classes mod \( m \) and they are determined by the \( m \) possible remainders (or in Gauss’ terminology, **residues**) \( r = 0, 1, \ldots, m – 1 \) on division by \( m \). These \( m \) numbers constitute the **standard residue system** (SRS) mod \( m \).

Replacing anyone of the residues by a number to which it is congruent yields another **complete residue system** (CRS) (e.g., \{7, 50, 30, 3, –3, 5, –1\} is a CRS mod 7).

**A least absolute residue system** mod \( m \) is a CRS whose members have the least absolute values possible (e.g., \{–3, –2, –1, 0, 1, 2, 3\} is a least absolute residue system mod 7).
Binary exponentiation

An interesting class of problems of a computational nature ask for the standard residue of a power of a number, e.g.,

What are the last two digits of the number $2^{284}$?

In the absence of powerful software, it may be very difficult to compute this 86-digit number. Luckily, one can answer the question without computing this large power, since all we require is the standard residue mod 100 of the number.

Computing powers mod $m$ is an exercise in artful use of simple operations. In particular, to compute the standard residue $a^n \mod m$, we employ a process called binary exponentiation: express the exponent in binary form as a sum of powers of 2, then perform the computation by repeated squaring and reduction of residues. We illustrate with the example above.

$$2^{284} = 2^{256+16+8+4}$$
$$= 2^{2^8+2^4+2^3+2^2}$$
$$= 2^{2^8 \cdot 2^4 \cdot 2^3 \cdot 2^2}$$
$$= 4^{2^7} \cdot 4^{2^3} \cdot 4^{2^2} \cdot 4^2$$
$$= 16^{2^6} \cdot 16^{2^2} \cdot 16^2 \cdot 16$$
\[ 256^{25} \cdot 256^2 \cdot 256 \cdot 16 \]
\[ \equiv 56^{25} \cdot 56^2 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 36^{24} \cdot 36 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 96^{23} \cdot 36 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 16^{22} \cdot 36 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 56^2 \cdot 36 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 36 \cdot 36 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 96 \cdot 56 \cdot 16 \pmod{100} \]
\[ \equiv 76 \cdot 16 \pmod{100} \]
\[ \equiv 16 \pmod{100} \]

Thus, the last two digits of \( 2^{284} \) are 16.

While this procedure is tedious for hand calculation, it is easily programmed into computational sofware.