"I asked you a question, buddy. ... What's the square root of 5,248?"
If $a$ and $b$ are positive integers, then
\[
gcd(a, b) = gcd(b, a \mod b)
\]

If $a$ is a positive integer, $gcd(a, 0) = a$. 
Euclid’s Algorithm

\[ \text{gcd}(a, b) = \text{gcd}(b, a \mod b) \quad \text{gcd}(a, 0) = a \]

```c
int gcd(int a, int b){ /* Assumes: a >= b, b >= 0 */
    if (b == 0) {
        return a;
    } else {
        return gcd(b, a % b);
    }
}
```

Note: \( \text{gcd}(b, a) = \text{gcd}(a, b) \)
Euclid’s Algorithm

Repeatedly use $\text{gcd}(a, b) = \text{gcd}(b, a \mod b)$ to reduce numbers until you get $\text{gcd}(g, 0) = g$.

$\text{gcd}(660, 126) =$
Euclid’s Algorithm

Repeatedly use $\text{gcd}(a, b) = \text{gcd}(b, a \mod b)$ to reduce numbers until you get $\text{gcd}(g, 0) = g$.

$\text{gcd}(660, 126) = \text{gcd}(126, 660 \mod 126) = \text{gcd}(126, 30)$
$\quad = \text{gcd}(30, 126 \mod 30) \quad = \text{gcd}(30, 6)$
$\quad = \text{gcd}(6, 30 \mod 6) \quad = \text{gcd}(6, 0)$
$\quad = 6$
Euclid’s Algorithm

Repeatedly use \( \gcd(a, b) = \gcd(b, a \mod b) \) to reduce numbers until you get \( \gcd(g, 0) = g \).

Equations with recursive calls:

\[
gcd(660, 126) = gcd(126, 660 \mod 126) = gcd(126, 30) = gcd(30, 126 \mod 30) = gcd(30, 6) = gcd(6, 30 \mod 6) = gcd(6, 0) = 6
\]

Tableau form:

\[
\begin{align*}
660 &= 5 \times 126 + 30 \\
126 &= 4 \times 30 + 6 \\
30 &= 5 \times 6 + 0
\end{align*}
\]
Bézout’s theorem

If $a$ and $b$ are positive integers, then there exist integers $s$ and $t$ such that
\[ \gcd(a, b) = sa + tb. \]
Extended Euclidean algorithm

• Can use Euclid’s Algorithm to find \( s, t \) such that

\[
\gcd(a, b) = sa + tb
\]
Extended Euclidean algorithm

- Can use Euclid’s Algorithm to find $s, t$ such that
  \[ \gcd(a, b) = sa + tb \]

**Step 1 (Compute GCD & Keep Tableau Information):**

\[
\begin{array}{cccc}
  a & b & b & a \mod b = r \\
  35 & 27 & 27 & 35 \mod 27 = 8 \\
  \end{array}
\]

\[
\begin{array}{c}
  \gcd(35, 27) = \gcd(27, 35 \mod 27) = \gcd(27, 8) \\
  35 = 1 \times 27 + 8 \\
  \end{array}
\]
Extended Euclidean algorithm

- Can use Euclid’s Algorithm to find $s, t$ such that
  \[ \text{gcd}(a, b) = sa + tb \]

**Step 1 (Compute GCD & Keep Tableau Information):**

\[
\begin{array}{cccccc}
\text{a} & \text{b} & \text{b} & \text{a mod b} & \text{b} & \text{r} \\
35 & 27 & 27 & 35 \mod 27 & 27 & 8 \\
\text{gcd}(35, 27) &= \text{gcd}(27, 35 \mod 27) &= \text{gcd}(27, 8) \\&= \text{gcd}(8, 27 \mod 8) &= \text{gcd}(8, 3) \\&= \text{gcd}(3, 8 \mod 3) &= \text{gcd}(3, 2) \\&= \text{gcd}(2, 3 \mod 2) &= \text{gcd}(2, 1) \\&= \text{gcd}(1, 2 \mod 1) &= \text{gcd}(1, 0) \\
35 &= 1 \times 27 + 8 \\
27 &= 3 \times 8 + 3 \\
8 &= 2 \times 3 + 2 \\
3 &= 1 \times 2 + 1 \\
\end{array}
\]
Extended Euclidean algorithm

- Can use Euclid’s Algorithm to find $s, t$ such that
  \[ \text{gcd}(a, b) = sa + tb \]

Step 2 (Solve the equations for $r$):

\[
\begin{align*}
  a &= q \times b + r \\
  35 &= 1 \times 27 + 8 \\
  27 &= 3 \times 8 + 3 \\
  8 &= 2 \times 3 + 2 \\
  3 &= 1 \times 2 + 1 \\

  r &= a - q \times b \\
  8 &= 35 - 1 \times 27
\end{align*}
\]
Extended Euclidean algorithm

• Can use Euclid’s Algorithm to find $s, t$ such that

$$\gcd(a, b) = sa + tb$$

Step 2 (Solve the equations for $r$):

\[
\begin{align*}
  a &= q \cdot b + r & r &= a - q \cdot b \\
  35 &= 1 \cdot 27 + 8 & 8 &= 35 - 1 \cdot 27 \\
  27 &= 3 \cdot 8 + 3 & 3 &= 27 - 3 \cdot 8 \\
  8 &= 2 \cdot 3 + 2 & 2 &= 8 - 2 \cdot 3 \\
  3 &= 1 \cdot 2 + 1 & 1 &= 3 - 1 \cdot 2
\end{align*}
\]
Extended Euclidean algorithm

- Can use Euclid’s Algorithm to find $s, t$ such that
  $gcd(a, b) = sa + tb$

Step 3 (Backward Substitute Equations):

$$8 = 35 - 1 \times 27$$
$$3 = 27 - 3 \times 8$$
$$2 = 8 - 2 \times 3$$
$$1 = 3 - 1 \times 2$$
Extended Euclidean algorithm

- Can use Euclid’s Algorithm to find \( s, t \) such that

\[
gcd(a, b) = sa + tb
\]

**Step 3 (Backward Substitute Equations):**

1. \( 8 = 35 - 1 \times 27 \)
2. \( 3 = 27 - 3 \times 8 \)
3. \( 2 = 8 - 2 \times 3 \)
4. \( 1 = 3 - 1 \times 2 \)

Plug in the def of 2

Re-arrange into 3’s and 8’s
Extended Euclidean algorithm

• Can use Euclid’s Algorithm to find \( s, t \) such that

\[
gcd(a, b) = sa + tb
\]

Step 3 (Backward Substitute Equations):

\[
\begin{aligned}
8 &= 35 - 1 \times 27 \\
3 &= 27 - 3 \times 8 \\
2 &= 8 - 2 \times 3 \\
1 &= 3 - 1 \times 2
\end{aligned}
\]

Plug in the def of 2

Re-arrange into 3’s and 8’s

Plug in the def of 3

Re-arrange into 8’s and 27’s
Extended Euclidean algorithm

- Can use Euclid’s Algorithm to find $s, t$ such that

$$\text{gcd}(a, b) = sa + tb$$

**Step 3 (Backward Substitute Equations):**

8 = 35 - 1 * 27

3 = 27 - 3 * 8

2 = 8 - 2 * 3

1 = 3 - 1 * 2

Plug in the def of 2

Re-arrange into 3’s and 8’s

Plug in the def of 3

Re-arrange into 8’s and 27’s

Re-arrange into 27’s and 35’s
Let $0 \leq a, b < m$. Then, $b$ is the multiplicative inverse of $a$ (modulo $m$) iff $ab \equiv_m 1$.

**Multiplicative inverse mod $m$**

<table>
<thead>
<tr>
<th>$x$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

mod 7

<table>
<thead>
<tr>
<th>$x$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td>2</td>
<td>5</td>
<td>8</td>
<td>1</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>2</td>
<td>6</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>8</td>
<td>4</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>7</td>
<td>4</td>
<td>1</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>9</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

mod 10
Multiplicative inverse mod $m$

Suppose $\gcd(a, m) = 1$

By Bézout’s Theorem, there exist integers $s$ and $t$ such that $sa + tm = 1$.

$s$ is the multiplicative inverse of $a$ (modulo $m$):

$$1 = sa + tm \equiv_m sa$$

So... we can compute multiplicative inverses with the extended Euclidean algorithm

These inverses let us solve modular equations...
Example

Solve: \( 7x \equiv_{26} 1 \)
Example

Solve: \( 7x \equiv_{26} 1 \)

\[
\text{gcd}(26, 7) = \text{gcd}(7, 5) = \text{gcd}(5, 2) = \text{gcd}(2, 1) = 1
\]
Example

Solve: $7x \equiv_{26} 1$

$$\text{gcd}(26, 7) = \text{gcd}(7, 5) = \text{gcd}(5, 2) = \text{gcd}(2, 1) = 1$$

$$26 = 3 \times 7 + 5$$
$$7 = 1 \times 5 + 2$$
$$5 = 2 \times 2 + 1$$
Example

Solve: $7x \equiv_{26} 1$

$\text{gcd}(26, 7) = \text{gcd}(7, 5) = \text{gcd}(5, 2) = \text{gcd}(2, 1) = 1$

$26 = 3 \times 7 + 5 \quad 5 = 26 - 3 \times 7$

$7 = 1 \times 5 + 2 \quad 2 = 7 - 1 \times 5$

$5 = 2 \times 2 + 1 \quad 1 = 5 - 2 \times 2$
Example

Solve: \(7x \equiv_{26} 1\)

\[
gcd(26, 7) = gcd(7, 5) = gcd(5, 2) = gcd(2, 1) = 1
\]

\[
26 = 3 \cdot 7 + 5 \quad 5 = 26 - 3 \cdot 7
\]

\[
7 = 1 \cdot 5 + 2 \quad 2 = 7 - 1 \cdot 5
\]

\[
5 = 2 \cdot 2 + 1 \quad 1 = 5 - 2 \cdot 2
\]

\[
1 = 5 - 2 \cdot (7 - 1 \cdot 5)
\]

\[
= (-2) \cdot 7 + 3 \cdot 5
\]

\[
= (-2) \cdot 7 + 3 \cdot (26 - 3 \cdot 7)
\]

\[
= (-11) \cdot 7 + 3 \cdot 26
\]
Example

Solve: \(7x \equiv 26 \ 1\)

\[
\gcd(26, 7) = \gcd(7, 5) = \gcd(5, 2) = \gcd(2, 1) = 1
\]

\[
26 = 3 \times 7 + 5 \quad 5 = 26 - 3 \times 7
\]

\[
7 = 1 \times 5 + 2 \quad 2 = 7 - 1 \times 5
\]

\[
5 = 2 \times 2 + 1 \quad 1 = 5 - 2 \times 2
\]

\[1 = 5 - 2 \times (7 - 1 \times 5)\]

\[= (-2) \times 7 + 3 \times 5\]

\[= (-2) \times 7 + 3 \times (26 - 3 \times 7)\]

\[= (-11) \times 7 + 3 \times 26\]

Multiplicative inverse of 7 modulo 26

Now \((-11) \mod 26 = 15\). So, \(x = 15 + 26k\) for \(k \in \mathbb{Z}\).
Example of a more general equation

Now solve: \( 7y \equiv_{26} 3 \)

We already computed that \( 15 \) is the multiplicative inverse of \( 7 \) modulo \( 26 \). That is, \( 7 \cdot 15 \equiv_{26} 1 \)

If \( y \) is a solution, then multiplying by \( 15 \) we have
\[
15 \cdot 7 \cdot y \equiv_{26} 15 \cdot 3
\]

Substituting \( 15 \cdot 7 \equiv_{26} 1 \) into this on the left gives
\[
y = 1 \cdot y \equiv_{26} 15 \cdot 3 \equiv_{26} 19
\]

This shows that every solution \( y \) is congruent to \( 19 \).
Example of a more general equation

Now solve: \(7y \equiv_{26} 3\)

Multiplying both sides of \(y \equiv_{26} 19\) by 7 gives
\[
7y \equiv_{26} 7 \cdot 19 \equiv_{26} 3
\]
So, any \(y \equiv_{26} 19\) is a solution.

Thus, the set of numbers of the form \(y = 19 + 26k\), for any \(k\), are exactly solutions of this equation.
Math mod a prime is especially nice

gcd(a, m) = 1 if m is prime and 0 < a < m so can always solve these equations mod a prime.

mod 7
Multiplicative Inverses and Algebra

Adding to both sides is an equivalence:

\[ x \equiv_m y \]
\[ x + c \equiv_m y + c \]

The same is not true of multiplication...

unless we have a multiplicative inverse \( cd \equiv_m 1 \)

\[ x \equiv_m y \]
\[ cx \equiv_m cy \]
### Modular Exponentiation mod 7

<table>
<thead>
<tr>
<th>x</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a</th>
<th>a^1</th>
<th>a^2</th>
<th>a^3</th>
<th>a^4</th>
<th>a^5</th>
<th>a^6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>
Exponentiation

• Compute $78365^{81453}$

• Compute $78365^{81453} \mod 104729$

• Output is small
  – need to keep intermediate results small
Small Multiplications

Since $b = qm + (b \mod m)$, we have $b \mod m \equiv_m b$.

And since $c = tm + (c \mod m)$, we have $c \mod m \equiv_m c$.

Multiplying these gives $(b \mod m)(c \mod m) \equiv_m bc$.

By the Lemma from a few lectures ago, this tells us $bc \mod m = (b \mod m)(c \mod m) \mod m$.

Okay to mod $b$ and $c$ by $m$ before multiplying if we are planning to mod the result by $m$. 
Repeated Squaring – small and fast

Since \( b \mod m \equiv_m b \) and \( c \mod m \equiv_m c \)
we have \( bc \mod m = (b \mod m)(c \mod m) \mod m \)

So \( a^2 \mod m = (a \mod m)^2 \mod m \)
and \( a^4 \mod m = (a^2 \mod m)^2 \mod m \)
and \( a^8 \mod m = (a^4 \mod m)^2 \mod m \)
and \( a^{16} \mod m = (a^8 \mod m)^2 \mod m \)
and \( a^{32} \mod m = (a^{16} \mod m)^2 \mod m \)

Can compute \( a^k \mod m \) for \( k = 2^i \) in only \( i \) steps
What if \( k \) is not a power of 2?
Fast Exponentiation Algorithm

81453 in binary is 10011111000101101

81453 = 2^{16} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^5 + 2^3 + 2^2 + 2^0

a^{81453} = a^{2^{16}} \cdot a^{2^{13}} \cdot a^{2^{12}} \cdot a^{2^{11}} \cdot a^{2^{10}} \cdot a^9 \cdot a^5 \cdot a^3 \cdot a^2 \cdot a^0

a^{81453} \text{ mod m} =
(...((((a^{2^{16}} \text{ mod m} \cdot 
    a^{2^{13}} \text{ mod m} ) \text{ mod m} \cdot 
    a^{2^{12}} \text{ mod m}) \text{ mod m} \cdot 
    a^{2^{11}} \text{ mod m}) \text{ mod m} \cdot 
    a^{2^{10}} \text{ mod m}) \text{ mod m} \cdot 
    a^9 \text{ mod m}) \text{ mod m} \cdot 
    a^5 \text{ mod m}) \text{ mod m} \cdot 
    a^3 \text{ mod m}) \text{ mod m} \cdot 
    a^2 \text{ mod m}) \text{ mod m} \cdot 
    a^0 \text{ mod m}) \text{ mod m}

The fast exponentiation algorithm computes $a^k \text{ mod m}$ using $\leq 2\log k$ multiplications mod m

Uses only $16 + 9 = 25$ multiplications
Fast Exponentiation: $a^k \mod m$ for all $k$

Another way....

$$a^{2j} \mod m = (a^j \mod m)^2 \mod m$$

$$a^{2j+1} \mod m = ((a \mod m) \cdot (a^{2j} \mod m)) \mod m$$
Fast Exponentiation

public static int FastModExp(int a, int k, int modulus) {

    if (k == 0) {
        return 1;
    } else if ((k % 2) == 0) {
        long temp = FastModExp(a, k/2, modulus);
        return (temp * temp) % modulus;
    } else {
        long temp = FastModExp(a, k-1, modulus);
        return (a * temp) % modulus;
    }
}

\[ a^{2j} \mod m = (a^j \mod m)^2 \mod m \]
\[ a^{2j+1} \mod m = ((a \mod m) \cdot (a^{2j} \mod m)) \mod m \]
Using Fast Modular Exponentiation

• Your e-commerce web transactions use SSL (Secure Socket Layer) based on RSA encryption

• RSA
  – Vendor chooses random 512-bit or 1024-bit primes $p, q$ and 512/1024-bit exponent $e$. Computes $m = p \cdot q$
  – Vendor broadcasts $(m, e)$
  – To send $a$ to vendor, you compute $C = a^e \mod m$ using fast modular exponentiation and send $C$ to the vendor.
  – Using secret $p, q$ the vendor computes $d$ that is the multiplicative inverse of $e \mod (p - 1)(q - 1)$.
  – Vendor computes $C^d \mod m$ using fast modular exponentiation.
  – Fact: $a = C^d \mod m$ for $0 < a < m$ unless $p \mid a$ or $q \mid a$