Division modulo $N$

$a x \equiv 1 \pmod{N}$

Solve for $x = a^{-1}$

$a \cdot a^{-1} x \equiv a \cdot 1 \pmod{N}$

$x \equiv a^{-1} \pmod{N}$

$a x \equiv b \pmod{N}$

$x \equiv a^{-1} b \pmod{N}$

$a \cdot b$ are given

$x$ is unknown

$a x \equiv 1 \pmod{N}$

$\implies$ when we divide $a x$ by $N$, remainder $= 1$

$N \mid a x - 1$

$1 = x (\overline{a} + (-q) \pmod{N})$
1 = \ x \cdot a + y \cdot N

Find integers \ x \text{ and } y

\text{such that } \gcd(a, N) = 1 \quad \text{Condition 1}

\text{greatest common divisor}

1) check \ \gcd(a, N) = 1 \quad \text{a and N are relatively prime}

2) find \ x \text{ and } y

Euclid's Algorithm \quad (300 \text{ BC !})

\gcd(a, b) \quad a \geq b \geq 0

\gcd(a, b) = \gcd(b, a \mod b)

<table>
<thead>
<tr>
<th>Step</th>
<th>a</th>
<th>b</th>
<th>a \mod b</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>1035</td>
<td>759</td>
<td>276 = 1035 - 3 \cdot 759</td>
</tr>
<tr>
<td>2nd</td>
<td>759</td>
<td>276</td>
<td>207 = 759 - 2 \cdot 276</td>
</tr>
<tr>
<td>3rd</td>
<td>276</td>
<td>207</td>
<td>69 = 276 - 3 \cdot 207</td>
</tr>
<tr>
<td>4th</td>
<td>207</td>
<td>69</td>
<td>0 = 207 - 3 \cdot 69</td>
</tr>
</tbody>
</table>

Last non-zero remainder is \gcd.
\[ \text{gcd}(a, b) : \]
\[
\text{If } b = 0, \quad \text{return } a \\
\text{return } \text{gcd}(b, a \mod b)
\]

1) \text{Convex:} \\
\[
\text{LHS} \\
\text{gcd}(a, b) = \text{gcd}(b, a \mod b) \\
\]
\[
\Rightarrow \text{ assume that } g \text{ is the gcd of } a \text{ & } b \\
g = \text{gcd}(a, b) \\
\text{Show that } g \text{ divides RHS} \\
g \mid a \quad \text{and} \quad g \mid b \\
\text{Show that } g \text{ divides } a \mod b \text{ as well} \\
y = a \mod b \\
\Rightarrow a = b \cdot s + y \\
\Rightarrow y = a - b \cdot s \\
= gt - (gy) \cdot s \\
a \mod b = y = g(t - gs) \\
\Rightarrow g \mid a \mod b \\
g \leq \text{gcd}(b, a \mod b) \\
\]
\[ \Leftarrow: \text{ assume that } g = \text{gcd}(b, a \mod b) \]
Assume that $g = \gcd(b, a \mod b)$ and show that $g \mid a$ and $g \mid b$.

Then $g \leq \gcd(a, b)$.

$\Rightarrow$ $\gcd(a, b) = \gcd(b, a \mod b)$

2) Running time.

$\gcd(a, b) = \gcd(b, a \mod b)$

- $\# \text{ of recursive calls} \cdot \text{Cost per call}$

$\gcd(a, b)$:

$\nu = a \mod b \leftarrow \text{Integer division}$

$\gcd(b, \nu)$

$\# \text{ of recursive calls} \rightarrow O(\text{value of } a) \\
O(\nu) \rightarrow O(n) \rightarrow O(\log n)$

$n = 4096$ bits

largest value

$\nu = 2^{n-1} \approx 2^{4096}$

The value of $a$ and $b$...
Compare the values of $a$ and $b$

$\gcd(a, b) \quad \gamma = a \mod b$

Case I: $a \geq 2b \quad a \geq b \leq a/2$

$(a, b) \rightarrow$ what can we say about $\gamma$

\[
\begin{align*}
\gamma < b & \leq a/2 \\
\implies \gamma & < a/2
\end{align*}
\]

$\gcd(a, \gamma) \quad \gcd(b, \gamma)$

Case II: $a < 2b \quad a \geq b > a/2$

$\gamma = a \mod b = a - b' < a/2$

Both $(a, b)$ will reduce by at least $a/2$ in 2 calls of $\gcd$

Every 2 calls, the number of bits in $a$ and $b$ will reduce by 1 bit

$n$ bits in beginning $\Rightarrow 2^{\log_2 n}$ cells at most

$\gcd: \# \text{ of recursive cells} = O(a) \times O(n) \overbrace{\text{one per call}}^<

= O(n^3)$

Lower bounds: very difficult

For sorting: $\Omega(n \log n)$ for comparisons
Extended Euclid's method

1) \( \gcd(a, b) = g \)
2) \( g = ax + by \)

<table>
<thead>
<tr>
<th>( a )</th>
<th>( b )</th>
<th>( r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1035</td>
<td>759</td>
<td>276</td>
</tr>
<tr>
<td>759</td>
<td>276</td>
<td>207</td>
</tr>
<tr>
<td>276</td>
<td>207</td>
<td>69</td>
</tr>
<tr>
<td>207</td>
<td>69</td>
<td>0</td>
</tr>
</tbody>
</table>

\( a \mod b \quad g \)

\( 1035 - 1 \cdot 759 = 276 \)
\( 759 - 2 \cdot 276 = 207 \)
\( 276 - 1 \cdot 207 = 69 \)
\( 207 - 3 \cdot 69 = 0 \)

Forward pass

Back substitution phase

\( g = 276 - 1 \cdot 207 \)
\( = 276 - 1 \cdot (759 - 2 \cdot 276) \)
\( = 276 - 1 \cdot 759 + 2 \cdot 276 \)
\( = -1 \cdot 759 + 3 \cdot 276 \)
\( = -1 \cdot 759 + 3 \cdot (1035 - 1 \cdot 759) \)

\( g = \frac{276}{(1035 - 1 \cdot 759)} \)

\( 0 \in O(n^3) \) time

\( -1 \mod 25 \)
\[ 11x + 3 = \equiv 25 \mod 25 \]

\[ x = 11^{-1} \mod 25 \]

\( -9 \) is the inverse of 11

\[-9 + 25 = 16 \]

16 is the inverse of 11 mod 25

\[ 11 \cdot 16 \equiv 1 \mod 25 \]

\[ 176 \equiv 1 \mod 25 \]

\[ 25 \mid 176 - 1 \]

\[ \begin{array}{ccc} a & b & r \\ \hline 25 & 11 & 3 \\ 11 & 3 & 2 \\ 3 & 2 & 1 \end{array} \]

\[ 1 = 3 - 2 \\
= 3 - (11 - 3 \cdot 3) \\
= -1 \cdot 11 + 4 \cdot 3 \\
= -1 \cdot 11 + 4 \cdot 1 \\
= 1 = 4 \cdot 25 + (-9) \cdot 11 \]

\[ \begin{array}{c} x \\ y \end{array} \]

modular exponentiation

\[ x^y \mod N \]

\[ \left( \begin{array}{c} x \\ y \end{array} \right)^n \mod N \]

\[ 2^n \]

\[ \frac{a}{25} \]
1) Compute \( \left( \frac{327}{7} \mod 853 \right)^y \mod N \)

2) \( (7, 7 \mod 853) \)

\[
\begin{align*}
    x' &= (x \cdot x) \mod N \\
    x &= (x' \cdot x) \mod N
\end{align*}
\]

327 multiplications & mod operations

\( y \) multiplications

\( \sim 2^n \) multiplications

\( 2^n \cdot O(n^2) \) operations

Exponential time

3) Doubling the power each time

\[
\begin{align*}
    7^{327} \mod 853 \\
    7^1 &\equiv 7 \mod 853 \\
    7^2 &= 7 \cdot 7 \equiv 49 \mod 853
\end{align*}
\]
\[
7^2 = 7 \times 7 = 49 \equiv 49 \mod 853 \\
7^3 = 49 \times 49 = 2401 = 695 \mod 853 \\
7^4 = 695^2 = 492025 = 227 \mod 853 \\
7^5 = 349 \\
7^6 = 675 \\
7^7 = 123 \\
7^8 = 628 \\
7^9 = 298 \\
7^{10} = 123 \\
7^{11} = 628 \\
7^{12} = 298 \\
7^{327} = 7^{252} \cdot 7^4 \cdot 7 \cdot 7 \cdot 7 \cdot 7 \equiv 7 \\
= 298 \cdot 123 \cdot 695 \cdot 49 \cdot 7 \equiv 298 \mod 853 \\
= 828 \cdot 695 \cdot 49 \cdot 7 \\
= 538 \cdot 49 \cdot 7 \\
= 772 \cdot 7 \\
= 286 \\
7^{327} = 286 \mod 853
\]
\[
\log(y) \cdot o(n^2)
\]

\[
\log(2^n) \cdot o(n^2)
\]

\[
\underbrace{n \cdot o(n^2)} = o(n^3)
\]

---

**Modular exponentiation \((x, y, N)\)** \(x, y, N\) all take \(n\) bits

**Phase 1:**

Compute powers of \(x\) via doubling

\[
\displaystyle o(n) = \left\lfloor \log y \right\rfloor \text{ multiplication } o(n) \text{ integer division to compute } x \mod N
\]

\[
o(n) \times o(n^2)
\]

\[
= o(n^3) \text{ time}
\]

**Phase 2:**

Multiply relevant powers, 2 at a time, \(\mod N\) each time

At most \(n\) "powers" have to be multiplied

\(n\) \(\mod\) operations

\[
o(n) \times o(n^2) = o(n^3)
\]

Total time = \(o(n^3)\)