GCDs and LCMs

Let \( a, b \in \mathbb{Z} \) (not both 0)

\[
\text{gcd}(a, b) = \text{greatest common divisor of } a \text{ and } b \\
= \text{the largest integer that divides } a \text{ and } b \\
= \max \{d \in \mathbb{Z}^+ : d \mid a \text{ and } d \mid b \} \\
(\text{used to reduce fractions})
\]

\[
\text{Ex} \quad \text{gcd}(90, 100) = 10 \\
\text{Ex} \quad \text{gcd}(24, 48) = 24 \\
\text{Ex} \quad \text{gcd}(13, 14) = 1
\]

\[\checkmark \quad \text{relatively prime } \iff \text{gcd}=1\]

Let \( a, b \in \mathbb{Z}^+ \).

\[
\text{lcm}(a, b) = \text{least common multiple of } a \text{ and } b \\
= \text{the smallest positive integer divisible by } a \text{ and } b \\
= \min \{m \in \mathbb{Z}^+ : a \mid m \text{ and } b \mid m \} \\
(\text{used to find the LCM})
\]

\[
\text{Ex} \quad \text{lcm}(8, 9) = 72 \\
\text{If } a, b \text{ are relatively prime, } \\
\text{lcm}(a, b) = ab
\]

\[
\text{Ex} \quad \text{lcm}(4, 12) = 12 \\
\text{Ex} \quad \text{lcm}(6, 10) = 30
\]
Finding GCDs and LCMs Using Prime Factors

Ex: Find gcd(200, 1500)

200 = $2^3 \cdot 5^2$
1500 = $2^2 \cdot 3 \cdot 5^3$

Put in 0, 1 exponents:

\[
\frac{200}{1500} = \frac{2^3 \cdot 3^0 \cdot 5^2}{2^2 \cdot 3^1 \cdot 5^3}
\]

gcd = $2^2 \cdot 3^0 \cdot 5^2$

= 100

In general, to find gcd(a, b):
Find the prime factors of a and b.
Let $p_1, p_2, \ldots, p_n$ be the primes that appear in the prime factor of a or b.

Let $a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} 
\quad b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}

\gcd(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)}$

Here, \( \gcd(200, 1500) = \min(3, 2), 3, \min(0, 1), 5^2 \)

= $2^2 \cdot 3^0 \cdot 5^2$

= 100
Ex Find \( \text{lcm}(200, 1500) \)

\[
\begin{align*}
200 &= 2^3 \cdot 3^0 \cdot 5^2 \\
1500 &= 2^2 \cdot 3^1 \cdot 5^3 \\
\text{lcm} &= 2^3 \cdot 3^1 \cdot 5^3 \\
&= 3000
\end{align*}
\]

\[\text{For each prime, take the larger exponent}\]

In general, to find \( \text{lcm}(a, b) \):

same as finding \( \text{gcd}(a, b) \), except

\[
\text{lcm}(a, b) = p_1^{\max(a_1, b_1)} p_2^{\max(a_2, b_2)} \ldots p_n^{\max(a_n, b_n)}
\]

If \( a, b \in \mathbb{Z}^+ \), then \( ab = \text{gcd}(a, b) \cdot \text{lcm}(a, b) \)

\[\text{(HW #33)}\]

**Special Case**

If \( a, b \) are relatively prime

\[
\begin{align*}
\text{gcd}(a, b) &= 1 \\
\text{lcm}(a, b) &= ab
\end{align*}
\]
\[ \gcd(12, 18) = 6 \]

\[ \text{lcm}(12, 18) = 36 \]
THE DIVISION "ALGORITHM"

55 is divisible by 5, because

$$55 = 5 \cdot 11$$

57 is not

\[
\begin{array}{cccc}
11 & \text{R} & 2 \\
5 \div 57 & & & \\
5 & \downarrow & \text{quotient} & \text{remainder} \\
-5 & \downarrow & \text{dividend} & \text{divisor} \\
0 & \downarrow & (11) & (\text{must be}) \\
-5 & \downarrow & (11, 4) & \text{when } \div \text{by } 5 \\
2 & \downarrow & -20 & \text{(i.e., } r \in \mathbb{Z}, 0 \leq r < 5) \\
\end{array}
\]

Let at \( Z, d \in \mathbb{Z}^+ \) Then, there are unique \( q, r \in \mathbb{Z} \) (\( 0 \leq r < d \)) such that \( a = dq + r \)

\[
\begin{array}{cccc}
d \mid a & \iff & r = 0 \\
d \mid a \iff r \in \mathbb{Z} \\
\end{array}
\]

Ex What are the quotient and remainder when \(-19\) is divided by \(4\)?

Quotient = \( \lfloor \frac{-19}{4} \rfloor = \lfloor -4.75 \rfloor = \lfloor -4 \rfloor = -4 \)

Remainder = \( a - dq = -19 - (-4 \cdot -5) = -19 - 20 = -39 \)

\(-19 = (4)(-5) + 1 \equiv -20 \equiv 1 \mod 4 \)

\( q = -5, r = 1 \)
Next: Classify integers according to their remainders when you divide by a given divisor.

Ex divisor = "modulus" = 5

Imagine a wheel spoke:

\[ \overline{R_0} \equiv 0 \pmod{5} \]

10

5

0

= 4 \pmod{5} (\overline{R_4} \quad 14 \quad 9 \quad 4 \quad -1 \quad 0 \quad -4 \quad 1 \quad 6 \quad 11) \quad (\overline{R_1} \equiv 1 \pmod{5})

3

2

\[ \overline{R_3} \equiv 3 \pmod{5} \]

\[ \overline{R_2} \equiv 2 \pmod{5} \]

5 congruence classes
MODULAR ARITHMETIC (Gauss)

Let \( a \in \mathbb{Z}, \; m \in \mathbb{Z}^+ \).
Then, \( a \mod m \) is the remainder when \( a \) is divided by \( m \).

**Example:** \( 11 \mod 5 = 1 \)

\[
11 = 5 \cdot 2 + 1
\]

**Example:** \( -1 \mod 5 = 4 \)

\[
\begin{align*}
\text{Ex} \quad 978 \mod 7 &= 7\quad \left( \frac{978}{7} \right) = 139, \\
139 \times 7 &= 973 \quad \text{(largest multiple of } 7 \leq 978) \\
r &= 978 - 973 = 5
\end{align*}
\]

Let \( a, b \in \mathbb{Z}, \; m \in \mathbb{Z}^+ \).
\( a \) is congruent to \( b \) modulo \( m \), or \( a \equiv b \pmod{m} \),
\( \iff a \mod m = b \mod m \) (\( a, b \) have same remainder on division by \( m \))
\( \iff m \mid (a-b) \)
\( \iff \exists k \in \mathbb{Z} \) such that \( a = b + km \) (eqn) (\( a, b \) can differ by some multiple of \( m \))

**Example:**
\[
\begin{align*}
7 &\equiv 2 \pmod{5} \quad \Rightarrow \quad 7 \equiv 22 \pmod{5} \\
22 &\equiv 2 \pmod{5}
\end{align*}
\]

Also: \( 5 \mid (22-7) \)

**Example:** \( 7 \not\equiv 23 \pmod{5} \)

\[
\begin{array}{c}
7 \\
23
\end{array}
\]

\[
\begin{array}{c}
22
\end{array}
\]

\[
\begin{array}{c}
17
\end{array}
\]

\[
\begin{array}{c}
12
\end{array}
\]

\[
\begin{array}{c}
10
\end{array}
\]

\[
\begin{array}{c}
8
\end{array}
\]

\[
\begin{array}{c}
6
\end{array}
\]

\[
\begin{array}{c}
4
\end{array}
\]

\[
\begin{array}{c}
2
\end{array}
\]

\[
\begin{array}{c}
0
\end{array}
\]

\[
\begin{array}{c}
1
\end{array}
\]

\[
\begin{array}{c}
3
\end{array}
\]

\[
\begin{array}{c}
5
\end{array}
\]
Prove \( a \equiv b \pmod{m} \iff \exists k \in \mathbb{Z} \ (a=b+km) \) (1)

To be on the same spoke, \(a\) and \(b\) can differ by a multiple of \(m\).

If \(a \equiv b \pmod{m}\), and \(c \equiv d \pmod{m}\), then

\[a+c \equiv b+d \pmod{m},\]
\[ac \equiv bd \pmod{m}\]

(Proofs helpful for HW) p.122

Ex. \(9 \equiv 2 \pmod{7}\)
\(12 \equiv 5 \pmod{7}\)

\[9+12 \equiv 2+5 \pmod{7}\]
\[21 \equiv 7 \pmod{7}\]

In general,
\[5 \equiv 2 \pmod{7}\]

Only the remainder matters as far as spokes go.
Applications

Hashing Functions

Storing records that are uniquely identified by a key "k" (e.g., SSN).

Division Method:

\[ h(k) = k \mod m \]

"Folding" the list of possible SSNs.

We might get collisions!

Resolutions

1. Rosen: Linear probing

2. Separate chaining

Requires dynamic memory allocation.
Ex: Pseudorandom #s (games, simulations, ...)

We want a sequence of "random" #s between 0, 1.

Most computers use the linear congruential method.

Seed $x_0$,

Recursive def'n:

$$x_{n+1} = (ax_n + c) \pmod{m}$$

Output: $\frac{x_0}{m}, \frac{x_1}{m}, \frac{x_2}{m}, ...$

Ex: Cryptology

Are all proofs?