Today

- Greatest Common Divisor (GCD)
- Euclid’s algorithm
- Proof of Correctness
- Reading: Chapter 4.3
Primes and GCD

- Universe: \( U = \mathbb{N} = \{0, 1, 2, \ldots\} \)
- \( a \) divides \( b \) (written \( a \mid b \)) iff \( \exists k. b = ak \)
- Set of divisors: \( D(a) = \{d : (d \mid a)\} \)
- \( p \geq 2 \) is prime if \( D(p) = \{1, p\} \)
- Common divisors of \( a \) and \( b \): \( D(a) \cap D(b) \)
- \( \text{gcd}(a, b) = \max\{D(a) \cap D(b)\} \)
Primes and GCD

- Universe: \( U = \mathbb{N} = \{0, 1, 2, \ldots\} \)
- \( a \) divides \( b \) (written \( a \mid b \)) iff \( \exists k \cdot b = ak \)
- Set of divisors: \( D(a) = \{d : (d \mid a)\} \)
- \( p \geq 2 \) is prime if \( D(p) = \{1, p\} \)
- Common divisors of \( a \) and \( b \): \( D(a) \cap D(b) \)
- \( \text{gcd}(a, b) = \text{max}\{D(a) \cap D(b)\} \)

**Question:** \( D(30) \cap D(42) = \)

(A) \( \{1, 3, 5\} \); (B) \( \{1, 6, 7\} \); (C) \( \{1, 3\} \); (D) \( \{1, 3, 6\} \);
Prime factorization

- Let \((p_i)_{i \geq 1} = (2, 3, 5, 7, 11, 13, \ldots)\) the sequence of primes
- Every positive integer \(n\) is a product of primes: \(n = \prod_{i \leq k} p_i^{e_i}\)
- Prime factorization is essentially unique
Prime factorization

- Let \((p_i)_{i \geq 1} = (2, 3, 5, 7, 11, 13, \ldots)\) the sequence of primes
- Every positive integer \(n\) is a product of primes: \(n = \prod_{i \leq k} p_i^{e_i}\)
- Prime factorization is essentially unique

If \(n = \prod_{i \leq k} p_i^{e_i}\) then \(D(n)\) is the set

(A) \(\{p_i^{e_i} : i = 1, \ldots, k\}\);
(B) \(\{\prod_{i \leq k} p_i^{c_i} : \forall i, c_i \leq e_i\}\);
(C) \(\{\prod_{i \leq k} p_i^{c_i} : \forall i, c_i < e_i\}\);
(D) \(\{\prod_{i \leq m} p_i^{e_i} : m \leq k\}\);
Prime factorization

- Let \((p_i)_{i \geq 1} = (2, 3, 5, 7, 11, 13, \ldots)\) the sequence of primes.
- Every positive integer \(n\) is a product of primes: \(n = \prod_{i \leq k} p_i^{e_i}\).
- Prime factorization is essentially unique.

If \(n = \prod_{i \leq k} p_i^{e_i}\) then \(D(n)\) is the set

(A) \(\{p_i^{e_i} : i = 1, \ldots, k\}\);

(B) \(\{\prod_{i \leq k} p_i^{c_i} : \forall i. c_i \leq e_i\}\);

(C) \(\{\prod_{i \leq k} p_i^{c_i} : \forall i. c_i < e_i\}\);

(D) \(\{\prod_{i \leq m} p_i^{e_i} : m \leq k\}\);

The common divisors of \(n = \prod_{i \leq k} p_i^{e_i}\) and \(m = \prod_{i \leq k} p_i^{f_i}\) are

\[D(n) \cap D(m) = \left\{ \prod_{i \leq k} p_i^{c_i} : \forall i. c_i \leq \min(e_i, f_i) \right\}\]
The greatest common divisor of \( n = \prod_{i \leq k} p_i^{e_i} \) and \( m = \prod_{i \leq k} p_i^{f_i} \) is

\[
D(n) \cap D(m) = \prod_{i \leq k} p_i^{\min(e_i, f_i)}
\]
The greatest common divisor of $n = \prod_{i \leq k} p_i^{e_i}$ and $m = \prod_{i \leq k} p_i^{f_i}$ is

$$D(n) \cap D(m) = \prod_{i \leq k} p_i^{\min(e_i,f_i)}$$

**Question:** What is the GCD of 30 and 42?

(A) 2; (B) 3; (C) 6; (D) \{1, 2, 3, 6\}
Zeros and Ones

What are the divisors of 1?

- $D(1) = \{1\}$
- $D(1) = \{1\}$
- $D(1) = \mathbb{N}$
- $D(1) = \emptyset$

What are the divisors of 0?

- $D(0) = \{0\}$
- $D(0) = \{0\}$
- $D(0) = \mathbb{N}$
- $D(0) = \emptyset$

What is the GCD of 7 and 0?

- $\text{gcd}(7, 0) = 7$
- $\text{gcd}(7, 0) = 0$
- $\text{gcd}(7, 0) = 1$
- $\text{gcd}(7, 0) = \infty$

Special case:

- $\text{gcd}(0, 0) = 0$
What are the divisors of 1?

(A) $D(1) = 1$; (B) $D(1) = \{1\}$; (C) $D(1) = \mathbb{N}$; (D) $D(1) = \emptyset$
What are the divisors of 1?

(A) $D(1) = 1$; (B) $D(1) = \{1\}$; (C) $D(1) = \mathbb{N}$; (D) $D(1) = \emptyset$

What are the divisors of 0?

(A) $D(0) = 0$; (B) $D(0) = \{0\}$; (C) $D(0) = \mathbb{N}$; (D) $D(0) = \emptyset$
What are the divisors of 1?

(A) \( D(1) = 1 \);  (B) \( D(1) = \{1\} \);  (C) \( D(1) = \mathbb{N} \);  (D) \( D(1) = \emptyset \)

What are the divisors of 0?

(A) \( D(0) = 0 \);  (B) \( D(0) = \{0\} \);  (C) \( D(0) = \mathbb{N} \);  (D) \( D(0) = \emptyset \)

What is the GCD of 7 and 0? \( \gcd(7, 0) \) equals . . .

(A) 7;  (A) 0;  (A) 1;  (A) \( \infty \);
What are the divisors of 1?

(A) \( D(1) = 1; \)

(B) \( D(1) = \{1\}; \)

(C) \( D(1) = \mathbb{N}; \)

(D) \( D(1) = \emptyset \)

What are the divisors of 0?

(A) \( D(0) = 0; \)

(B) \( D(0) = \{0\}; \)

(C) \( D(0) = \mathbb{N}; \)

(D) \( D(0) = \emptyset \)

What is the GCD of 7 and 0? \( \gcd(7, 0) \) equals ... 

(A) 7; 

(A) 0; 

(A) 1; 

(A) \( \infty \); 

Special case: \( \gcd(0, 0) = 0 \)
Let’s make it harder

Can you compute it (by hand) in 60 seconds?

- \( \text{gcd}(12, 52) = \text{???} \)

(A) Working on it . . .  (B) Yes, computed!  (C) No way!

. . . but I can write a program to compute it.
Let’s make it harder

Can you compute it (by hand) in 60 seconds?

- \( \text{gcd}(12, 52) = \text{???} \)
- \( \text{gcd}(123, 345) = \text{???} \)

(A) Working on it . . .  (B) Yes, computed!  (C) No way!

. . . but I can write a program to compute it.
Let’s make it harder

Can you compute it (by hand) in 60 seconds?

- $\gcd(12, 52) = ???$
- $\gcd(123, 345) = ???$
- $\gcd(6432, 4366) = ???$

(A) Working on it . . . (B) Yes, computed! (C) No way!

. . . but I can write a program to compute it.
Let’s make it harder

Can you compute it (by hand) in 60 seconds?

- \( \gcd(12, 52) = \text{???} \)
- \( \gcd(123, 345) = \text{???} \)
- \( \gcd(6432, 4366) = \text{???} \)
- \( \gcd(635432, 463366) = \text{???} \)

(A) Working on it … (B) Yes, computed! (C) No way!

… but I can write a program to compute it.
Euclid’s algorithm

def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return euclid(b, mod(a,b))

Let's prove it!
def euclid(a, b):  # a, b: nonnegative integers
    if (b == 0):
        return a
    else:
        return euclid(b, mod(a, b))

gcd(635432, 463366) = gcd(463366, 635432 − 463366) =
gcd(463366, 172066) = gcd(172066, 463366 − 2 · 172066) =
gcd(172066, 120334) = gcd(120344, 172066 − 120334) =
gcd(120344, 51732) = gcd(51732, 120344 − 2 · 51732) =
gcd(51732, 16880) = gcd(16880, 51732 − 3 · 16880) =
gcd(16880, 1092) = gcd(1092, 16880 − 15 · 1092) =
gcd(1092, 500) = gcd(500, 92) = gcd(92, 40) = gcd(40, 12) =
gcd(12, 4) = gcd(4, 0) = 4.
Euclid’s algorithm

```python
def euclid(a, b):
    # a, b: nonnegative integers
    if (b == 0):
        return a
    else:
        return euclid(b, mod(a, b))
gcd(635432, 463366) = gcd(463366, 635432 − 463366) =
gcd(463366, 172066) = gcd(172066, 463366 − 2 · 172066) =
gcd(172066, 120334) = gcd(120344, 172066 − 120334) =
gcd(120344, 51732) = gcd(51732, 120344 − 2 · 51732) =
gcd(51732, 16880) = gcd(16880, 51732 − 3 · 16880) =
gcd(16880, 1092) = gcd(1092, 16880 − 15 · 1092) =
gcd(1092, 500) = gcd(500, 92) = gcd(92, 40) = gcd(40, 12) =
gcd(12, 4) = gcd(4, 0) = 4.
Let’s prove it!
```
**Question:** If \((a, b)\) and \((c, d)\) have the same divisors, then they have the same GCD.

\[
(D(a, b) = D(c, d)) \implies (\gcd(a, b) = \gcd(c, d))
\]

(A) Yes; (B) No; (C) Maybe
**Question:** If \((a, b)\) and \((c, d)\) have the same divisors, then they have the same GCD.

\[
(D(a, b) = D(c, d)) \implies (\gcd(a, b) = \gcd(c, d))
\]

(A) Yes; (B) No; (C) Maybe

**Question:** If \((a, b)\) and \((c, d)\) have the same GCD, then they then have the same divisors:

\[
(\gcd(a, b) = \gcd(c, d)) \implies (D(a, b) = D(c, d))
\]

(A) Yes; (B) No; (C) Maybe
Claim: $a, b$ and $(a \mod b), b$ have the same common divisors.
Claim: \(a, b\) and \((a \mod b), b\) have the same common divisors.

Proof: We show \(D(a) \cap D(b) = D(a \mod b)) \cap D(b)\) by proving set inclusion in both directions.
Idea: Common divisors

Claim: $a, b$ and $(a \mod b), b$ have the same common divisors.

Proof: We show $D(a) \cap D(b) = D(a \mod b)) \cap D(b)$ by proving set inclusion in both directions.

- Assume $d \in D(a) \cap D(b)$. WTP: $d \in D(a \mod b)$
  - $a = dx$
  - $b = dy$
  - $r = a \mod b$ satisfies $a = qb + r$
  - Therefore, $r = a - qb = dx - qdy = d(x - qy)$ and $d \in D(r)$.
**Claim:** $a, b$ and $(a \mod b), b$ have the same common divisors.

**Proof:** We show $D(a) \cap D(b) = D(a \mod b)) \cap D(b)$ by proving set inclusion in both directions.

- Assume $d \in D(a) \cap D(b)$. WTP: $d \in D(a \mod b)$
  - $a = dx$
  - $b = dy$
  - $r = a \mod b$ satisfies $a = qb + r$
  - Therefore, $r = a - qb = dx - qdy = d(x - qy)$ and $d \in D(r)$.

- Assume $d \in D(b) \cap D(a \mod b)$. WTP: $d \in D(a)$
  - $r = a \mod b = dx$
  - $b = dy$
  - Therefore, $a = qb + r = qdy + dx = d(qy + x)$ and $d \in D(a)$
Back to our program

def euclid(a,b):  # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Idea:

- If $b = 0$, then $D(a) \cap D(b) = D(a) \cap \mathbb{N} = D(a)$, and $\gcd(a, b) = a$. 
def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Idea:

- If $b = 0$, then $D(a) \cap D(b) = D(a) \cap \mathbb{N} = D(a)$, and $\gcd(a, b) = a$.
- Otherwise, $D(a) \cap D(b) = D(b) \cap D(a \mod b)$
- So, $\gcd(a, b) = \gcd(b, a \mod b)$, and if $\text{euclid}(b, \text{mod}(a,b))$ is correct, then $\text{euclid}(a,b)$ is also correct.
Proof by induction

def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Claim: For all $a, b$, $\text{euclid}(a, b) = \gcd(a, b)$

Proof: By (strong) induction on what?
def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Claim: For all a, b, euclid(a,b) = gcd(a, b)
Proof: By (strong) induction on what?

(A) a; (B) b; (C) a + b; (D) Don’t know
Proof by induction

```python
def euclid(a, b): # a, b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a, b)))
```

Claim: For all \( a, b \), \( \text{euclid}(a, b) = \text{gcd}(a, b) \)

Proof: By (strong) induction on what?

(A) \( a \); (B) \( b \); (C) \( a + b \); (D) Don’t know

Claim: For all \( a \geq b \), \( \text{euclid}(a, b) = \text{gcd}(a, b) \)
def euclid(a,b):  # a, b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Claim: For all $a, b$, $\text{euclid}(a,b) = \gcd(a, b)$

Proof: By (strong) induction on what?

(A) $a$; (B) $b$; (C) $a + b$; (D) Don’t know

Claim: For all $a \geq b$, $\text{euclid}(a,b) = \gcd(a, b)$

Proof: By (strong) induction on what?

(A) $a$; (B) $b$; (C) $a + b$; (D) Don’t know
def euclid(a,b): # a,b: nonnegative integers
    if (b == 0) then a
    else return (euclid(b, mod(a,b)))

Claim: For all \( a \geq b \), \( \text{euclid}(a,b) = \text{gcd}(a,b) \)

Proof: By strong induction on \( b \).
def euclid(a, b):
    # a, b: nonnegative integers
    if (b == 0) then a
    else return (euclid(b, mod(a, b)))

Claim: For all $a \geq b$, $\text{euclid}(a, b) = \text{gcd}(a, b)$

Proof: By strong induction on $b$.

- Base case: $b = 0$. $\text{euclid}(a, b) = a$ and $\text{gcd}(a, 0) = a$. 
Proof details

def euclid(a, b):
    # a, b: nonnegative integers
    if (b == 0) then a
    else return (euclid(b, mod(a, b)))

Claim: For all \( a \geq b \), \( \text{euclid}(a, b) = \gcd(a, b) \)

Proof: By strong induction on \( b \).

- Base case: \( b = 0 \). \( \text{euclid}(a, b) = a \) and \( \gcd(a, 0) = a \).
- Inductive step: Assume \( b > 0 \) and \( \text{euclid}(a, r) = \gcd(a, r) \)
  for all \( r < b \)
  - \( \text{euclid}(a, b) = \text{euclid}(b, r) \) where \( r = (a \mod b) < b \)
  - By induction hypothesis \( \text{euclid}(b, r) = \gcd(b, r) \)
  - But we proved that \( \gcd(b, r) = \gcd(a, b) \)
  - So, \( \text{euclid}(a, b) = \gcd(a, b) \)
Completing the proof

```python
def euclid(a, b):  # a, b: nonnegative integers
    if (b == 0) then a
    else return (euclid(b, mod(a, b)))
```

**Lemma:** For all $a \geq b$, $\text{euclid}(a, b) = \gcd(a, b)$

Proved!
Completing the proof

```python
def euclid(a, b):
    # a, b: nonnegative integers
    if (b == 0) then a
    else return (euclid(b, mod(a, b)))
```

**Lemma:** For all $a \geq b$, $\text{euclid}(a, b) = \gcd(a, b)$

Proved!

**Theorem:** For all $a, b$, $\text{euclid}(a, b) = \gcd(a, b)$

Proof:

- If $a \geq b$, then $\text{euclid}(a, b) = \gcd(a, b)$ by the lemma
- If $a < b$, then $b \geq 1$ and $a \mod b = a$.
- So, $\text{euclid}(a, b) = \text{euclid}(b, a) = \gcd(b, a) = \gcd(a, b)$
Application: breaking weak RSA keys

“Mining Your Ps and Qs: Widespread weak keys in network devices”, Heninger et al., 2012.

- RSA cryptosystem: Public Key $K = P \cdot Q$, product of two large secret random primes
- Many devices use bad pseudorandom generators, so $P, Q$ are not very random
- Different devices end up using the same $P$ or $Q$
- Collect all RSA public keys from the internet, and compute their GCD
- If $K_1 = PQ, K_2 = PQ'$, then $\gcd(K_1, K_2) = P$. 
def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Theorem: For any positive integers $a, b$, there exists integers $x, y$ such that $\gcd(a, b) = xa + yb$. 
def euclid(a, b):  # a, b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a, b)))

Theorem: For any positive integers $a, b$, there exists integers $x, y$ such that $\text{gcd}(a, b) = xa + yb$.

- Can you prove the theorem?

(A) Yes; (B) No; (C) Maybe.
def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Theorem: For any positive integers $a, b$, there exists integers $x, y$ such that $\text{gcd}(a, b) = xa + yb$.

- Can you prove the theorem?

(A) Yes; (B) No; (C) Maybe.

- What proof method would you use?
Bezout’s Identity

def euclid(a,b): # a,b: nonnegative integers
    if (b == 0):
        return a
    else:
        return (euclid(b, mod(a,b)))

Theorem: For any positive integers \(a, b\), there exists integers \(x, y\) such that \(\text{gcd}(a, b) = xa + yb\).

- Can you prove the theorem?

(A) Yes; (B) No; (C) Maybe.

- What proof method would you use?

- Can you modify the algorithm to return \((x, y)\)?
def bezout(a,b):  # a,b: nonnegative integers
  if (b == 0):
    return (1,0)
  else:
    (q,r) = divmod(a,b)  # a = qb+r
    (x,y) = bezout(b,r)  # xb+yr = gcd(b,r) = gcd(a,b)
    return (????,???)

Question: What shall we return?

(A) (x + qy, x)

(B) (y, x − qy)

(C) (x − qy, x)

(D) (y, x + qy)
How can we check if the program is correct?

(A) Prove by induction that `bezout` returns valid \((x, y)\)

(B) Run the program on some test inputs

(C) First run some tests, then prove correctness
Correctness?

How can we check if the program is correct?

(A) Prove by induction that \texttt{bezout} returns valid \((x, y)\)

(B) Run the program on some test inputs

(C) First run some tests, then prove correctness

- How can you test the program?
- How can you generate tests without already having a correct program?
def euclid(a, b):  # a, b: nonnegative integers
    if (b == 0):
        return a
    else:
        (q, r) = divmod(a, b)  # a = qb+r
        return (euclid(b, r))

def bezout(a, b):  # a, b: nonnegative integers
    if (b == 0):
        return (1, 0)
    else:
        (q, r) = divmod(a, b)  # a = qb+r
        (x, y) = bezout(b, r)  # xb+yr = gcd(b, r) = gcd(a, b)
        return (y, x-q*y)

def test(a, b):
    (x, y) = bezout(a, b)
    return (x*a+y*b == euclid(a, b))
Proving correctness

```python
def euclid(a,b):  # a,b: nonnegative integers
    if (b == 0): return a
    else:
        (q,r) = divmod(a,b)  # a = qb+r
        return (euclid(b,r))

def bezout(a,b):  # a,b: nonnegative integers
    if (b == 0): return (1,0)
    else:
        (q,r) = divmod(a,b)  # a = qb+r
        (x,y) = bezout(b,r)  # xb+yr = gcd(b,r) = gcd(a,b)
        return (y,x-q*y)

Claim: For every positive integers \(a, b\), \bezout(a,b)\ computes integers \(x, y\) such that \(xa+yb = \euclid(a,b)\)

Proof: By induction on \(b\) ... left as exercise
```
Let $m > 1$ be a positive integer
We have seen how to compute $a + b \pmod{m}$, $a - b \pmod{m}$, $a \cdot b \pmod{m}$
Can you also compute $a/b \pmod{m}$?
More simply, can you compute $a^{-1} \pmod{m}$?

**Question:** Can you compute $a^{-1} \pmod{1}$ for any of the following numbers

(A) $a=2$; (B) $a=5$; (C) $a=7$; (D) $a=12$;
Modular inversion using Bezout’s identity

- Want to compute \(a^{-1} \pmod m\)
- Compute \((x, y)\) such that \(xa + ym = \gcd(a, m) = d\)
- If \(d = 1\), then \(xa = d - ym = 1 \pmod m\), so \(x = a^{-1} \pmod m\)
- If \(d > 1\), then \(a\) has no inverse modulo \(m\)

Proof:

- Assume for contradiction that \(d = \gcd(a, m) > 1\) and \(\exists c. a \cdot c = 1 \pmod m\)
- Then, \(\exists k. ac = 1 + km\)
- Since \(d\) divides both \(a\) and \(m\), it must also divide \(1 = ac - km\).
- This is only possible if \(d = 1\), contradiction!
Finite Fields

If $p$ is a prime, then $(\mathbb{Z}_p, +, \times)$ is a field

Axioms:
If $p$ is a prime, then $(\mathbb{Z}_p, +, \times)$ is a field

Axioms:

- **Associativity of $+$**: $\forall a, b, c. (a + b) + c = a + (b + c)$
- **Commutativity of $+$**: $\forall a, b. (a + b) = (b + a)$
- **Neutral element**: $\exists 0. \forall a. a + 0 = 0 + a = a$
- **Additive Inverses**: $\forall a. \exists b. a + b = 0. (b = -a)$
- **Associativity of $\times$**: ... 
- **Commutativity of $\times$**: ... 
- **Neutral element**: $\exists 1. \forall a. a \times 1 = a$
Finite Fields

If \( p \) is a prime, then \((\mathbb{Z}_p, +, \times)\) is a field

Axioms:

- **Associativity of \(+\):** \( \forall a, b, c. (a + b) + c = a + (b + c) \)
- **Commutativity of \(+\):** \( \forall a, b. (a + b) = (b + a) \)
- **Neutral element:** \( \exists 0. \forall a. a + 0 = 0 + a = a \)
- **Additive Inverses:** \( \forall a. \exists b. a + b = 0. (b = -a) \)
- **Associativity of \(\times\):** \( \ldots \)
- **Commutativity of \(\times\):** \( \ldots \)
- **Neutral element:** \( \exists 1. \forall a. a \times 1 = a \)
- **Multiplicative Inverses:** \( \forall a. (a \neq 0) \rightarrow \exists b. a \times b = b \times a = 1. (b = a^{-1}) \)
- **Distributivity:** \( \forall a, b, c. a \times (b + c) = a \times b + a \times c. \)
Applications of Finite Fields in CS

- Error Correcting Codes
- Fast Integer Arithmetics
- Cryptography
- Hashing
- ...

Daniele Micciancio  CSE20: Discrete Mathematics
Let $p, q$ be two different primes

$f : \mathbb{Z}_{pq} \rightarrow \mathbb{Z}_p \times \mathbb{Z}_q$ where $f(x) = (x \mod p, x \mod q)$

**Theorem:** $f$ is a bijection
Let $p, q$ be two different primes

$f : \mathbb{Z}_{pq} \to \mathbb{Z}_p \times \mathbb{Z}_q$ where $f(x) = (x \mod p, x \mod q)$

**Theorem:** $f$ is a bijection

Example: $p = 7, q = 5$

$f(22) = ??$

(A) (1, 1); (B) (2, 2); (C) (1, 2); (D) (2, 1);
Chinese Remainder Theorem

- Let \( p, q \) be two different primes
- \( f : \mathbb{Z}_{pq} \to \mathbb{Z}_p \times \mathbb{Z}_q \) where \( f(x) = (x \mod p, x \mod q) \)
- **Theorem:** \( f \) is a bijection

Example: \( p = 7, q = 5 \)

\( f(22) = ??? \)

(A) (1, 1); (B) (2, 2); (C) (1, 2); (D) (2, 1);

\( f^{-1}(5, 3) = ??? \)

(A) 12; (B) 19; (C) 26; (D) 33;
CRT proof

How can you define $f^{-1}$?

- $\gcd(p, q) = 1$

**Question:** We prove that $f$ is

(A) Injective (one-to-one);

(B) Surjective (onto);

(C) Bijective (one-to-one correspondence);
How can you define $f^{-1}$?

- $\gcd(p, q) = 1$
- Inverse of $p \pmod{q}$: $\exists x. xp = 1 \pmod{q}$

**Question:** We prove that $f$ is

(A) Injective (one-to-one);

(B) Surjective (onto);

(C) Bijective (one-to-one correspondence);
How can you define $f^{-1}$?

- $\gcd(p, q) = 1$
- Inverse of $p \pmod{q}$: $\exists x. xp = 1 \pmod{q}$
- Inverse of $q \pmod{p}$: $\exists y. yq = 1 \pmod{p}$

**Question**: We prove that $f$ is

(A) Injective (one-to-one);

(B) Surjective (onto);

(C) Bijective (one-to-one correspondence);
How can you define $f^{-1}$?

- $\gcd(p, q) = 1$
- Inverse of $p \pmod{q}$: $\exists x. xp = 1 \pmod{q}$
- Inverse of $q \pmod{p}$: $\exists y. yq = 1 \pmod{p}$
- $g(a, b) = xpb + yqa$

**Question:** We prove that $f$ is

(A) Injective (one-to-one);
(B) Surjective (onto);
(C) Bijective (one-to-one correspondence);
How can you define $f^{-1}$?

- $\gcd(p, q) = 1$
- Inverse of $p \pmod{q}$: $\exists x. xp = 1 \pmod{q}$
- Inverse of $q \pmod{p}$: $\exists y. yq = 1 \pmod{p}$
- $g(a, b) = xp b + yq a$
- $g(a, b) \pmod{p} \equiv yq a \equiv 1 \cdot a \equiv a \pmod{p}$

**Question:** We prove that $f$ is

(A) Injective (one-to-one);

(B) Surjective (onto);

(C) Bijective (one-to-one correspondence);
How can you define $f^{-1}$?

- $\gcd(p, q) = 1$
- Inverse of $p$ (mod $q$): $\exists x. xp = 1 \pmod{q}$
- Inverse of $q$ (mod $p$): $\exists y. yq = 1 \pmod{p}$
- $g(a, b) = xpb + yqa$
- $g(a, b) \pmod{p} \equiv yqa \equiv 1 \cdot a \equiv a \pmod{p}$
- $g(a, b) \pmod{q} \equiv xpb \equiv 1 \cdot b \equiv b \pmod{q}$

**Question:** We prove that $f$ is

**A** Injective (one-to-one);

**B** Surjective (onto);

**C** Bijective (one-to-one correspondence);
How can you define $f^{-1}$?

- $\gcd(p, q) = 1$
- Inverse of $p \pmod q$: $\exists x. xp = 1 \pmod q$
- Inverse of $q \pmod p$: $\exists y. yq = 1 \pmod p$
- $g(a, b) = xp + yqa$
- $g(a, b) \pmod p \equiv yqa \equiv 1 \cdot a \equiv a \pmod p$
- $g(a, b) \pmod q \equiv xp \equiv 1 \cdot b \equiv b \pmod q$
- $(f \circ g)(a, b) = (a, b)$

**Question:** We prove that $f$ is

(A) Injective (one-to-one);

(B) Surjective (onto);

(C) Bijective (one-to-one correspondence);
Let $A, B$ be finite sets of the same size $|A| = |B|$

If $f : A \rightarrow B$ is injective, then it is a bijection.

If $f : A \rightarrow B$ is surjective, then it is a bijection.
CAPE course evaluations are open
You should have received information to submit your evaluations on May 28
Evals are open till before final week
Remember to submit your course evals!