CSE 311: Foundations of Computing

Topic 3: Number Theory

"I *asked* you a question, buddy. ... What's the square root of 5,248?"

Number Theory

- **Direct relevance to computing**
  - **everything in a computer is a number**
    - colors on the screen are encoded as numbers

- **Many significant applications**
  - **Cryptography & Security**
  - **Data Structures**
  - **Distributed Systems**

I'm ALIVE!

```java
public class Test {
    final static int SEC_IN_YEAR = 365*24*60*60;
    public static void main(String args[]) {
        System.out.println(
            "I will be alive for at least " +
            SEC_IN_YEAR * 101 + " seconds."
        );
    }
}
```

I'm ALIVE!

```java
public class Test {
    final static int SEC_IN_YEAR = 365*24*60*60;
    public static void main(String args[]) {
        System.out.println(
            "I will be alive for at least " +
            SEC_IN_YEAR * 101 + " seconds."
        );
    }
}
```

```
    ----jGRASP exec: java Test
 I will be alive for at least -186619904 seconds.

    ----jGRASP: operation complete.
```

Modular Arithmetic

Modular Arithmetic

- Arithmetic over a finite domain

- Almost all computation is over a finite domain

Recall: Elementary School Division

For a, b with $b > 0$, we can divide b into a.

If $b \nmid a$, then we end up with a *remainder r* with $0 < r < b$. Now,

instead of $\quad \dfrac{a}{b} = q \quad$ we have $\quad \dfrac{a}{b} = q + \dfrac{r}{b}$

Multiplying both sides by b gives us $\quad a = qb + r$

Recall: Elementary School Division

For a, b with $b > 0$, we can divide b into a.

If $b \mid a$, then we have $a = qb$ for some q.
If $b \nmid a$, then we have $a = qb + r$ for some q, r with $0 < r < b$.

In general, we have $a = qb + r$ for some q, r with $0 \leq r < b$, where $r = 0$ iff $b \mid a$.

Division Theorem

Division Theorem

For a, b with $b > 0$

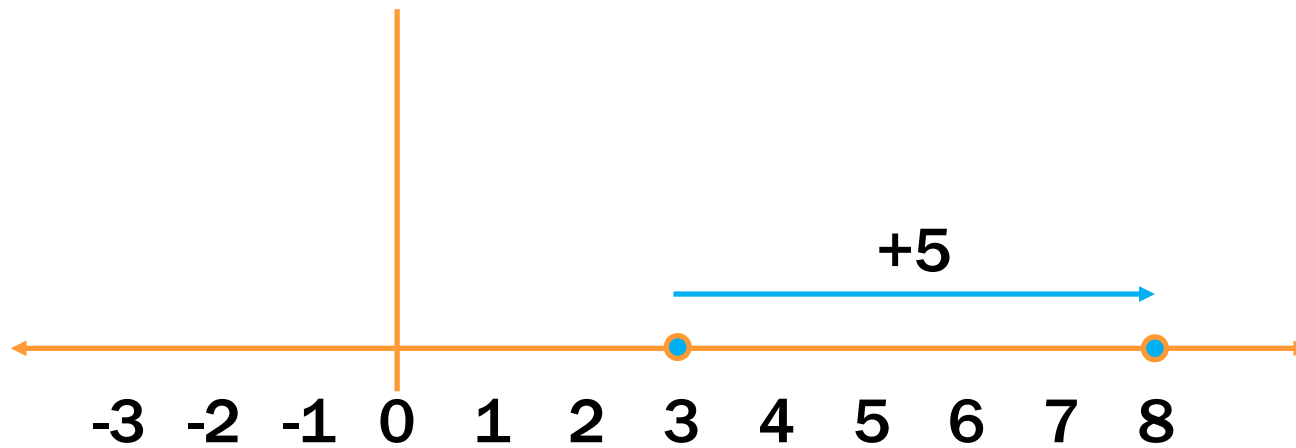there exist *unique* integers q, r with $0 \leq r < b$
such that $a = qb + r$.

To put it another way, if we divide b into a, we get a
unique quotient $q = a$ **div** b
and non-negative remainder $r = a$ **mod** b

$$a = (a \textbf{ div } b)\ b + (a \textbf{ mod } b)$$

$$\forall a\ \forall b\ \Big((b > 0) \rightarrow \big(a = (a \textbf{ div } b)b + (a \textbf{ mod } b) \big) \Big)$$
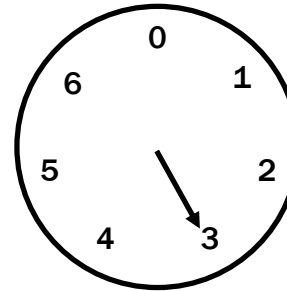
Ordinary arithmetic

$$3 + 5 = 8$$
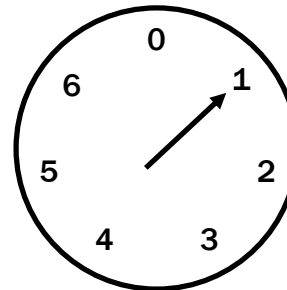
Arithmetic on a Clock

$$3 + 5 = 8$$

$$8 = 7 \cdot 1 + 1$$
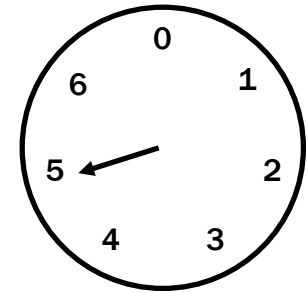$$15 = 7 \cdot 2 + 1$$
$$22 = 7 \cdot 3 + 1$$

If $a = 7q + r$, then $r \; (= a \bmod b)$ is
where you <u>stop</u> after taking a steps on the clock

Arithmetic, mod 7

(a + b) mod 7

(a × b) mod 7

+	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

x	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

Modular Arithmetic

Definition: "a is congruent to b modulo m"

For a, b, m with $m > 0$
$$a \equiv_m b \ := \ m \mid (a - b)$$

New notion of "sameness" that will help us understand modular arithmetic

Modular Arithmetic

Definition: "a is congruent to b modulo m"

For a, b, m with $m > 0$
$$a \equiv_m b \;:=\; m \mid (a - b)$$

The standard math notation is

$$a \equiv b \pmod{m}$$

A chain of equivalences is written

$$a \equiv b \equiv c \equiv d \pmod{m}$$

Many students find this confusing,
so we will use $\equiv_m$ instead.

Modular Arithmetic

Definition: "a is congruent to b modulo m"

For a, b, m with $m > 0$
$$a \equiv_m b \ := \ m \mid (a - b)$$

Check Your Understanding. What do each of these mean? When are they true?

$-1 \equiv_5 19$

This statement is true. 19 - (-1) = 20 which is divisible by 5

$x \equiv_2 0$

This statement is the same as saying "x is even"; so, any x that is even (including negative even numbers) will work.

$y \equiv_7 2$

This statement is true for y in { ..., -12, -5, 2, 9, 16, ...}. In other words, all y of the form 2+7k for k an integer.

The $\mathrm{mod}\ m$ function vs the $\equiv_m$ predicate

– **The $\mathrm{mod}\ m$ function maps any integer a to a remainder $a \bmod m \in \{0, 1, \ldots, m-1\}$.**
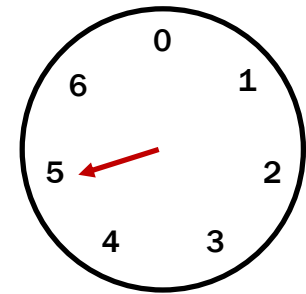
  Tells you where it lands on the clock.

– **Imagine grouping together all integers that have the same value of the $\mathrm{mod}\ m$ function.**

  They must differ by a multiple of m ($q_1 m + r$ vs $q_2 m + r$)

– **The $\equiv_m$ predicate compares integers a, b to see if if they differ by a multiple of m.**

  If they differ by a multiple of m, then walking from one to the other leaves you at the same spot on the clock.

Modular Arithmetic: A Property

> Let a, b, m be integers with $m > 0$.
> Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

This claim is an $\leftrightarrow$ ("iff")

Proof Plan:

1. $(a \bmod m = b \bmod m) \rightarrow (a \equiv_m b)$??
2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$??
3. $(a \bmod m = b \bmod m) \rightarrow (a \equiv_m b) \wedge$
      $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$ Intro $\wedge$: 1, 2
4. $(a \equiv_m b) \leftrightarrow (a \bmod m = b \bmod m)$ Biconditional: 3

Modular Arithmetic: A Property

> Let a, b, m be integers with $m > 0$.
> Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

1. $(a \bmod m = b \bmod m) \rightarrow (a \equiv_m b)$??

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

1.1. $a \bmod m = b \bmod m$ Assumption

1.? $a \equiv_m b$??

1. $(a \bmod m = b \bmod m) \rightarrow (a \equiv_m b)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

    1.1. $a \bmod m = b \bmod m$ Assumption

    1.? $m \mid a - b$??
    1.? $a \equiv_m b$ Def of $\equiv$
  1. $(a \bmod m = b \bmod m) \rightarrow (a \equiv_m b)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

1.1. $a \bmod m = b \bmod m$ Assumption

1.? $\exists q \ (a - b = qm)$??

1.? $m \mid a - b$ Def of |

1.? $a \equiv_m b$ Def of ≡

1. $(a \bmod m = b \bmod m) \rightarrow (a \equiv_m b)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

1.1. $a \bmod m = b \bmod m$ Assumption

1.2. $a = (a \operatorname{div} m)\, m + (a \bmod m)$ Apply Division

1.3. $b = (b \operatorname{div} m)\, m + (b \bmod m)$ Apply Division

1.? $\exists q\, (a - b = qm)$??

1.? $m \mid a - b$ Def of $\mid$

1.? $a \equiv_m b$ Def of $\equiv$

1. $(a \bmod m = b \bmod m) \to (a \equiv_m b)$ Direct Proof

Modular Arithmetic: A Property

> Let a, b, m be integers with $m > 0$.
> Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

1.1. $a \bmod m = b \bmod m$	Assumption
1.2. $a = (a \operatorname{div} m)\, m + (a \bmod m)$	Apply Division
1.3. $b = (b \operatorname{div} m)\, m + (b \bmod m)$	Apply Division
1.4. $a - b = \big((a \operatorname{div} m) - (b \operatorname{div} m)\big)\, m$	Algebra
1.5. $\exists q\, (a - b = qm)$	Intro $\exists$
1.6. $m \mid a - b$	Def of $\mid$
1.7. $a \equiv_m b$	Def of $\equiv$
1. $(a \bmod m = b \bmod m) \to (a \equiv_m b)$	Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \bmod m = b \bmod m$.

Assumption

Apply Division
Apply Division

Algebra

Intro $\exists$
Def of $|$
Def of $\equiv$

Therefore, $a \equiv_m b$.

Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \bmod m = b \bmod m$.

By the Division Theorem, we can write
$a = (a \operatorname{div} m)\, m + (a \bmod m)$ and
$b = (b \operatorname{div} m)\, m + (b \bmod m)$.

Therefore, $a \equiv_m b$.

Assumption

Apply Division
Apply Division

Algebra

Intro ∃
Def of |
Def of ≡
Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \bmod m = b \bmod m$.

Assumption

By the Division Theorem, we can write
$a = (a \operatorname{div} m)\, m + (a \bmod m)$ and
$b = (b \operatorname{div} m)\, m + (b \bmod m)$.

Apply Division
Apply Division

Subtracting these we can see that
$$a - b = \big((a \operatorname{div} m) - (b \operatorname{div} m)\big)m + $$
$$\big((a \bmod m) - (b \bmod m)\big)$$
$$= \big((a \operatorname{div} m) - (b \operatorname{div} m)\big) m$$

Algebra

since $(a \bmod m) - (b \bmod m) = 0$.

Intro $\exists$

…

Def of |

Therefore, $a \equiv_m b$.

Def of $\equiv$

Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \bmod m = b \bmod m$. Assumption

By the Division Theorem, we can write
$a = (a \operatorname{div} m)\, m + (a \bmod m)$ and
$b = (b \operatorname{div} m)\, m + (b \bmod m)$.

Apply Division
Apply Division

Subtracting these we can see that
$$a - b = \big((a \operatorname{div} m) - (b \operatorname{div} m)\big)m +$$
$$((a \bmod m) - (b \bmod m))$$
$$= \big((a \operatorname{div} m) - (b \operatorname{div} m)\big)m$$
since $(a \bmod m) - (b \bmod m) = 0$.

Algebra

Intro ∃

Therefore, by definition of divides, $m \mid (a - b)$
and so $a \equiv_m b$, by definition of congruent.

Def of |
Def of ≡
Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$ **??**

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$ Assumption

2.? $a \bmod m = b \bmod m$??
2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$	Assumption
2.2. $m \mid a - b$	Def of $\mid$

2.? $a \bmod m = b \bmod m$??

2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

    2.1. $a \equiv_m b$ Assumption

    2.2. $m \mid a - b$ Def of $\equiv$

    2.3. $\exists q \ (a - b = qm)$ Def of $\mid$

    2.? $a \bmod m = b \bmod m$??

2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$ Direct Proof

Modular Arithmetic: A Property

> Let a, b, m be integers with $m > 0$.
> Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$	Assumption
2.2. $m \mid a - b$	Def of $\equiv$
2.3. $\exists q \, (a - b = qm)$	Def of $\mid$
2.4. $a - b = km$	Elim $\exists$

2.? $a \bmod m = b \bmod m$	**??**
2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$	Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$ Assumption

2.2. $m \mid a - b$ Def of $\equiv$

2.3. $\exists q \, (a - b = qm)$ Def of $\mid$

2.4. $a - b = km$ Elim $\exists$

2.5. $a = (a \operatorname{div} m)\, m + (a \bmod m)$ Apply Division

2.? $a \bmod m = b \bmod m$??

2. $(a \equiv_m b) \to (a \bmod m = b \bmod m)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$ Assumption

2.2. $m \mid a - b$ Def of $\equiv$

2.3. $\exists q \, (a - b = qm)$ Def of $\mid$

2.4. $a - b = km$ Elim $\exists$

2.5. $a = (a \operatorname{div} m)\, m + (a \bmod m)$ Apply Division

2.6. $b = (a \operatorname{div} m - k)\, m + (a \bmod m)$ Algebra

2.? $a \bmod m = b \bmod m$??

2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$ Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$	Assumption
2.2. $m \mid a - b$	Def of $\equiv$
2.3. $\exists q \, (a - b = qm)$	Def of $\mid$
2.4. $a - b = km$	Elim $\exists$
2.5. $a = (a \operatorname{div} m) \, m + (a \bmod m)$	Apply Division
2.6. $b = (a \operatorname{div} m - k) \, m + (a \bmod m)$	Algebra
2.7. $b \operatorname{div} m = (a \operatorname{div} m - k) \wedge$	Apply DivUnique
$b \bmod m = a \bmod m$	
2.? $a \bmod m = b \bmod m$	??
2. $(a \equiv_m b) \rightarrow (a \bmod m = b \bmod m)$	Direct Proof

Modular Arithmetic: A Property

> Let a, b, m be integers with $m > 0$.
> Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

2.1. $a \equiv_m b$	Assumption
2.2. $m \mid a - b$	Def of $\equiv$
2.3. $\exists q\ (a - b = qm)$	Def of $\mid$
2.4. $a - b = km$	Elim $\exists$
2.5. $a = (a \operatorname{div} m)\, m + (a \bmod m)$	Apply Division
2.6. $b = (a \operatorname{div} m - k)\, m + (a \bmod m)$	Algebra
2.7. $b \operatorname{div} m = (a \operatorname{div} m - k) \wedge$ $\quad b \bmod m = a \bmod m$	Apply DivUnique
2.8. $a \bmod m = b \bmod m$	Elim $\wedge$
2. $(a \equiv_m b) \to (a \bmod m = b \bmod m)$	Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \equiv_m b$.

Therefore, $a \bmod m = b \bmod m$.

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \equiv_m b$.

Assumption

Then, $m \mid (a - b)$ by the definition of congruence.
So, $a - b = km$ for some integer k by the definition of
divides. Equivalently, $a = b + km$.

Def of $\equiv$
Def of $\mid$
Elim $\exists$

Apply Division

Algebra

Apply DivUnique
Elim $\exists$

Therefore, $a \bmod m = b \bmod m$.

Direct Proof

Modular Arithmetic: A Property

> Let a, b, m be integers with $m > 0$.
> Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \equiv_m b$.

Then, $m \mid (a - b)$ by the definition of congruence.
So, $a - b = km$ for some integer k by the definition of divides. Equivalently, $a = b + km$.

By the Division Theorem, we have $a = (a \text{ div } m)\, m + (a \bmod m)$, with $0 \le (a \bmod m) < m$.

Therefore, $a \bmod m = b \bmod m$.

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \equiv_m b$.

Assumption

Then, $m \mid (a - b)$ by the definition of congruence.
So, $a - b = km$ for some integer k by the definition of divides. Equivalently, $a = b + km$.

Def of $\equiv$
Def of $\mid$
Elim $\exists$

By the Division Theorem, we have $a = (a \text{ div } m)\, m + (a \bmod m)$, with $0 \leq (a \bmod m) < m$.

Apply Division

Combining these, we have $(a \text{ div } m)m + (a \bmod m) = a = b + km$. Solving for b gives $b = (a \text{ div } m)\, m + (a \bmod m) - km = \big((a \text{ div } m) - k\big)m + (a \bmod m)$.

Algebra

Apply DivUnique
Elim $\exists$

Therefore, $a \bmod m = b \bmod m$.

Direct Proof

Modular Arithmetic: A Property

Let a, b, m be integers with $m > 0$.
Then, $a \equiv_m b$ if and only if $a \bmod m = b \bmod m$.

Suppose that $a \equiv_m b$.

Then, $m \mid (a - b)$ by the definition of congruence.
So, $a - b = km$ for some integer k by the definition of divides. Equivalently, $a = b + km$.

By the Division Theorem, we have $a = (a \operatorname{div} m)\, m + (a \bmod m)$, with $0 \leq (a \bmod m) < m$.

Combining these, we have $(a \operatorname{div} m)m + (a \bmod m) = a = b + km$. Solving for b gives b $= (a \operatorname{div} m)\, m + (a \bmod m) - km = \big((a \operatorname{div} m) - k\big)m + (a \bmod m)$.

By the uniqueness property in the Division Theorem, we must have $b \bmod m = a \bmod m$.

Assumption

Def of ≡
Def of |
Elim ∃

Apply Division

Algebra

Apply DivUnique
Elim ∃

Direct Proof

Recall: Familiar Properties of "="

- **If $a = b$ and $b = c$, then $a = c$.**
  - i.e., if $a = b = c$, then $a = c$

- **If $a = b$ and $c = d$, then $a + c = b + d$.**
  - since $c = c$ is true, we can "$+\,c$" to both sides

- **If $a = b$ and $c = d$, then $ac = bd$.**
  - since $c = c$ is true, we can "$\times\,c$" to both sides

These facts allow us to use algebra to solve problems

The Algebra Rule

$$\boxed{\text{Algebra}} \quad \frac{x_1 = y_1 \;\ldots\; x_n = y_n}{\therefore \; x = y}$$

- **Algebra rule applies these properties:**
  - adding equations
  - multiplying equations by *a constant*   **Note**: no division (since domain is integers)

- **But also uses knowledge of**
  - arithmetic with constants
  - commutativity of multiplication (**e.g.,** $yx = xy$)
  - distributivity (**e.g.,** $a(b+c) = ab + bc$)

Recall: Familiar Properties of "="

- **If $a = b$ and $b = c$, then $a = c$.**
  - i.e., if $a = b = c$, then $a = c$

- **If $a = b$ and $c = d$, then $a + c = b + d$.**
  - since $c = c$ is true, we can "$+ c$" to both sides

- **If $a = b$ and $c = d$, then $ac = bd$.**
  - since $c = c$ is true, we can "$\times c$" to both sides

Same facts apply to "$\leq$"
with non-negative numbers

What about "$\equiv_m$"?

Modular Arithmetic: Basic Property

> Let a, b, c and m be integers with $m > 0$.
> If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

1. $(a \equiv_m b \land b \equiv_m c) \to (a \equiv_m c)$ **??**

Modular Arithmetic: Basic Property

Let a, b, c and m be integers with $m > 0$.
If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Suppose that $a \equiv_m b$ and $b \equiv_m c$.

Therefore, $a \equiv_m c$.

Modular Arithmetic: Basic Property

Let a, b, c and m be integers with $m > 0$.
If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Suppose that $a \equiv_m b$ and $b \equiv_m c$.

Assumption

By the definition of congruence, we know that
$m \mid (a - b)$ and $m \mid (b - c)$.

Elim $\wedge$

Def of $\equiv$

??

Therefore, $a \equiv_m c$.

Direct Proof

Modular Arithmetic: Basic Property

Let a, b, c and m be integers with $m > 0$.
If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Suppose that $a \equiv_m b$ and $b \equiv_m c$.

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (b - c)$. By the definition of divides, we know that $a - b = km$ and $b - c = jm$ for some integers k and j.

Therefore, $a \equiv_m c$.

Assumption

Elim $\wedge$

Def of $\equiv$

Def of $\mid$

Elim $\exists$

??

Direct Proof

Modular Arithmetic: Basic Property

Let a, b, c and m be integers with $m > 0$.
If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Suppose that $a \equiv_m b$ and $b \equiv_m c$.
Assumption

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (b - c)$. By the definition of divides, we know that $a - b = km$ and $b - c = jm$ for some integers k and j.

Elim $\wedge$
Def of $\equiv$
Def of $\mid$
Elim $\exists$

??
Def of $\equiv$

Therefore, we have $m \mid (a - c)$, so $a \equiv_m c$ by the definition of congruence.
Direct Proof

Modular Arithmetic: Basic Property

Let a, b, c and m be integers with $m > 0$.
If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Suppose that $a \equiv_m b$ and $b \equiv_m c$. **Assumption**

By the definition of congruence, we know that
$m \mid (a - b)$ and $m \mid (b - c)$. By the definition of
divides, we know that $a - b = km$ and $b - c = jm$
for some integers k and j.

Elim $\wedge$

Def of $\equiv$

Def of $\mid$

Elim $\exists$

??

… $\exists q \ (a - c = mq)$ … **Def of** $\mid$

Therefore, by the definition of divides, we have **Def of** $\equiv$
shown that $m \mid (a - c)$, and then, $a \equiv_m c$ by the **Direct Proof**
definition of congruence.

Modular Arithmetic: Basic Property

Let a, b, c and m be integers with $m > 0$.
If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Suppose that $a \equiv_m b$ and $b \equiv_m c$.

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (b - c)$. By the definition of divides, we know that $a - b = km$ and $b - c = jm$ for some integers k and j.

Adding these, gives $a - c = km + jm = (k + j)m$.

Therefore, by the definition of divides, we have shown that $m \mid (a - c)$, and then, $a \equiv_m c$ by the definition of congruence.

Assumption

Elim $\wedge$

Def of $\equiv$

Def of $\mid$

Elim $\exists$

Algebra

Intro $\exists$

Def of $\mid$

Def of $\equiv$

Direct Proof

Modular Arithmetic: Addition Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

1. $(a \equiv_m b \land c \equiv_m d) \rightarrow (a + c \equiv_m b + d)$ **??**

Modular Arithmetic: Addition Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$.   Assumption

Therefore, $a + c \equiv_m b + d$.   ??

Direct Proof

Modular Arithmetic: Addition Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$. | Assumption

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (c - d)$. | Elim $\wedge$
Def of $\equiv$

??

Therefore, $a + c \equiv_m b + d$. | Direct Proof

Modular Arithmetic: Addition Property

> Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$. *Assumption*

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (c - d)$. By the definition of divides, we know that $a - b = km$ and $c - d = jm$ for some integers k and j. *Elim $\wedge$*
Def of $\equiv$
Def of $\mid$
Elim $\exists$

Therefore, $a + c \equiv_m b + d$. *??*

Direct Proof

Modular Arithmetic: Addition Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$.

Assumption

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (c - d)$. By the definition of divides, we know that $a - b = km$ and $c - d = jm$ for some integers k and j.

Elim $\wedge$
Def of $\equiv$
Def of $\mid$
Elim $\exists$

Therefore, we have $m \mid (a + c) - (b + d)$, so we can see that $a + c \equiv_m b + d$ by the definition of congruence.

??
Def of $\equiv$
Direct Proof

Modular Arithmetic: Addition Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$. Assumption

By the definition of congruence, we know that Elim $\wedge$
$m \mid (a - b)$ and $m \mid (c - d)$. By the definition of Def of $\equiv$
divides, we know that $a - b = km$ and $c - d = jm$ Def of $\mid$
for some integers k and j. Elim $\exists$

... $\exists q \,((a+c) - (b+d) = mq)$ ... ??
Therefore, by the definition of divides, we have Def of $\mid$
shown $m \mid (a + c) - (b + d)$, and then, we have Def of $\equiv$
$a + c \equiv_m b + d$ by the definition of congruence. Direct Proof

Modular Arithmetic: Addition Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$.

Assumption

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (c - d)$. By the definition of divides, we know that $a - b = km$ and $c - d = jm$ for some integers k and j.

Elim $\wedge$
Def of $\equiv$
Def of $\mid$
Elim $\exists$

Adding these, gives $(a + c) - (b + d) = (a - b) + (c - d) = km + jm = (k + j)m$.

Algebra

Therefore, by the definition of divides, we have shown $m \mid (a + c) - (b + d)$, and then, we have $a + c \equiv_m b + d$ by the definition of congruence.

Intro $\exists$
Def of $\mid$
Def of $\equiv$

Direct Proof

Modular Arithmetic: Multiplication Property

Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $ac \equiv_m bd$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$.

By the definition of congruence, we know that $m \mid (a - b)$ and $m \mid (c - d)$. By the definition of divides, we know that $a - b = jm$ and $c - d = km$ for some integers j and k.

... $\exists q \, (ac - bd = mq)$ ...

Therefore, $m \mid ac - bd$ by the definition of divides, so $ac \equiv_m bd$ by the definition of congruence.

Assumption

Def of $\equiv$
Def of $\mid$
Elim $\exists$

Algebra

Intro $\exists$
Def of $\mid$
Def of $\equiv$

Direct Proof

Modular Arithmetic: Multiplication Property

> Let m be a positive integer. If $a \equiv_m b$ and $c \equiv_m d$, then $ac \equiv_m bd$.

Suppose that $a \equiv_m b$ and $c \equiv_m d$.   Assumption

By the definition of congruence, we know that
$m \mid (a - b)$ and $m \mid (c - d)$. By the definition of   Def of $\equiv$
divides, we know that $a - b = jm$ and $c - d = km$ Def of $\mid$
for some integers j and k.   Elim $\exists$

Equivalently, $a = b + jm$ and $c = d + km$.   Algebra
Multiplying these gives $ac = (b + jm)(d + km) =$
$bd + bkm + djm + jkm = bd + (bk + dj + jk)m$,   Intro $\exists$
so $ac - bd = (bk + dj + jk)m$.   Def of $\mid$

Therefore, $m \mid ac - bd$ by the definition of divides,   Def of $\equiv$
so $ac \equiv_m bd$ by the definition of congruence.   Direct Proof

Modular Arithmetic: Properties

If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.

Corollary: If $a \equiv_m b$, then $a + c \equiv_m b + c$.

If $a \equiv_m b$ and $c \equiv_m d$, then $ac \equiv_m bd$.

Corollary: If $a \equiv_m b$, then $ac \equiv_m bc$.

These properties are **sufficient** to allow us to do algebra with congruences

Another Property of "=" Used in Algebra

Can "plug in" (a.k.a. substitute)
the known value of a variable

Example: **given** $2y + 3x = 25$ **and** $x = 7y$,
follows that $2y + 21y = 25$.

We can call this "Algebra",
but it's a more primitive rule.

(See the reference sheet.)

Another Property of "=" Used in Algebra

Example: **given** $2y + 3x = 25$ **and** $x = 7y$,
follows that $2y + 21y = 25$**.**

This is <u>also</u> true of *congruences*!

(But we don't have the tools to prove it yet....)

Example: **given** $2y + 3x \equiv_m 25$ **and** $x \equiv_m 7y$,
follows that $2y + 21y \equiv_m 25$**.**

GCD

First GCD Fact

For every positive integer *a*,
if *a* > 0, then gcd(*a, 0*) = *a*.

$$\forall a\ ((a > 0) \rightarrow (a = \gcd(a, 0))$$

We proved this in Topic 2….

Let *a* be arbitrary. Suppose *a* > 0. Show that a satisfies the definition of gcd(*a, 0*).

Useful GCD Fact

> Let a and b be positive integers.
> We have gcd(a, b) = gcd(b, a mod b)

Proof Idea:
We will show that every number dividing a and b also divides b and a mod b.
I.e., $d|a$ and $d|b$ iff $d|b$ and $d|(a \bmod b)$.

Hence, their set of common divisors are the same,
which means that their greatest common divisor is the same.

Useful GCD Fact

> Let a and b be positive integers.
> We have gcd(a, b) = gcd(b, a mod b)

Proof (of $d|a$ and $d|b$ iff $d|b$ and $d|(a \bmod b)$):
By the Division Theorem, $a = qb + (a \bmod b)$ for some integer $q = a \operatorname{div} b$.

Suppose $d \mid b$ and $d \mid (a \bmod b)$.
Then $b = md$ and $(a \bmod b) = nd$ for some integers m and n.
Therefore $a = qb + (a \bmod b) = qmd + nd = (qm + n)d$.
So $d \mid a$ by the definition of divides.

Suppose $d \mid a$ and $d \mid b$.
Then $a = kd$ and $b = jd$ for some integers k and j.
Therefore $(a \bmod b) = a - qb = kd - qjd = (k - qj)d$.
So, $d \mid (a \bmod b)$ by the definition of divides.

Since they have the same common divisors, $\gcd(a, b) = \gcd(b, a \bmod b)$. ∎

Euclid's Algorithm

gcd(a, b) = gcd(b, a mod b)   gcd(a, 0) = a

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

Note: gcd(b, a) = gcd(a, b)

Euclid's Algorithm

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

gcd(660,126) =

Euclid's Algorithm

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

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

Bézout's theorem

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

$$(a > 0 \land b > 0) \to \exists s \, \exists t \, (\gcd(a,b) = sa + tb)$$

$$\forall a \, \forall b \, ((a > 0 \land b > 0) \to \exists s \, \exists t \, (\gcd(a,b) = sa + tb))$$

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):

$$\overset{\text{a}\quad\text{b}}{\gcd(35, 27)} = \overset{\text{b}\quad\text{a mod b = r}}{\gcd(27, 35 \bmod 27)} = \overset{\text{b}\quad\text{r}}{\gcd(27, 8)}$$

a = q * b + r
$$35 = 1 * 27 + 8$$

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):

$$
\begin{array}{lll}
 a \quad b & b \quad a \bmod b = r & b \quad r \\
\gcd(35, 27) = \gcd(27, 35 \bmod 27) = \gcd(27, 8) \\
 = \gcd(8, 27 \bmod 8) = \gcd(8, 3) \\
 = \gcd(3, 8 \bmod 3) = \gcd(3, 2) \\
 = \gcd(2, 3 \bmod 2) = \gcd(2, 1) \\
 = \gcd(1, 2 \bmod 1) = \gcd(1, 0)
\end{array}
$$

$a = q * b + r$
$35 = 1 * 27 + 8$
$27 = 3 * 8\ \ + 3$
$8 = 2 * 3\ \ + 2$
$3 = 1 * 2\ \ + \boxed{1}$

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):

a = q * b + r
$35 = 1 * 27 + 8$
$27 = 3 * 8\ \ + 3$
$8\ \ = 2 * 3\ \ + 2$
$3\ \ = 1 * 2\ \ + 1$

r = a − q * b

$8 = 35 - 1 * 27$

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):

$a = q * b + r$	$r = a - q * b$
$35 = 1 * 27 + 8$	$8 = 35 - 1 * 27$
$27 = 3 * 8 + 3$	$3 = 27 - 3 * 8$
$8 = 2 * 3 + 2$	$2 = 8 - 2 * 3$
$3 = 1 * 2 + ⓵$	$⓵ = 3 - 1 * 2$

Extended Euclidean algorithm

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

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

Step 3 (Backward Substitute Equations):

$$8 = 35 - 1 * 27$$

$$3 = 27 - 3 * 8$$

$$2 = 8 - 2 * 3$$

$$\boxed{1 = 3 - 1 * 2}$$

Extended Euclidean algorithm

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

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

Step 3 (Backward Substitute Equations):

Plug in the def of 2

$8 = 35 - 1 * 27$

$1 = 3 - 1 * (8 - 2 * 3)$
$ = 3 - 8 + 2 * 3$ Re-arrange into
$ = (-1) * 8 + 3 * 3$ 3's and 8's

$3 = 27 - 3 * 8$

$2 = 8 - 2 * 3$

$\boxed{1 = 3 - 1 * 2}$

Extended Euclidean algorithm

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

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

Step 3 (Backward Substitute Equations):

Plug in the def of 2

$$8 = 35 - 1 * 27$$

$$
\begin{aligned}
1 &= 3 - 1 * (8 - 2 * 3) \\
&= 3 - 8 + 2 * 3 \\
&= (-1) * 8 + 3 * 3
\end{aligned}
$$

Re-arrange into 3's and 8's

$$3 = 27 - 3 * 8$$

Plug in the def of 3

$$
\begin{aligned}
&= (-1) * 8 + 3 * (27 - 3 * 8) \\
&= (-1) * 8 + 3 * 27 + (-9) * 8 \\
&= 3 * 27 + (-10) * 8
\end{aligned}
$$

$$2 = 8 - 2 * 3$$

$$\boxed{1 = 3 - 1 * 2}$$

Re-arrange into 8's and 27'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):

$8 = 35 - 1 * 27$

$3 = 27 - 3 * 8$

$2 = 8 - 2 * 3$

$1 = 3 - 1 * 2$

Re-arrange into
27's and 35's

$1 = 3 - 1 * (8 - 2 * 3)$

$ = 3 - 8 + 2 * 3$ Re-arrange into

$ = (-1) * 8 + 3 * 3$ 3's and 8's

Plug in the def of 3

$ = (-1) * 8 + 3 * (27 - 3 * 8)$

$ = (-1) * 8 + 3 * 27 + (-9) * 8$

$ = 3 * 27 + (-10) * 8$ Re-arrange into

8's and 27's

$ = 3 * 27 + (-10) * (35 - 1 * 27)$

$ = 3 * 27 + (-10) * 35 + 10 * 27$

$ = 13 * 27 + (-10) * 35$

Multiplicative inverse $\bmod m$

Let $0 \leq a, b < m$. Then, b is the *multiplicative inverse of a (modulo m)* iff $ab \equiv_m 1$.

X	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

mod 7

X	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9
2	0	2	4	6	8	0	2	4	6	8
3	0	3	6	9	2	5	8	1	4	7
4	0	4	8	2	6	0	4	8	2	6
5	0	5	0	5	0	5	0	5	0	5
6	0	6	2	8	4	0	6	2	8	4
7	0	7	4	1	8	5	2	9	6	3
8	0	8	6	4	2	0	8	6	4	2
9	0	9	8	7	6	5	4	3	2	1

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**):**
$$1 \equiv_m sa \text{ since } m \mid 1 - sa \text{ (since } 1 - sa = tm)$$

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

These inverses let us solve modular equations...

Recall: Properties of Modular Arithmetic

If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

If $a \equiv_m b$, then $a + c \equiv_m b + c$.

If $a \equiv_m b$, then $ac \equiv_m bc$.

These properties are sufficient to allow
us to do algebra with congruences

Multiplicative Inverses and Algebra

Adding to both sides easily reversible:

$$-c \quad x \equiv_m y \quad +c$$
$$x + c \equiv_m y + c$$

The same is not true of multiplication...

unless we have a multiplicative inverse $cd \equiv_m 1$

$$\times d \quad x \equiv_m y \quad \times c$$
$$cx \equiv_m cy$$

Recall: 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):**

$$1 \equiv_m sa \text{ since } m \mid 1 - sa \text{ (since } 1 - sa = tm)$$

We can compute multiplicative inverses with the Extended Euclidean algorithm

These inverses let us solve modular equations...

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Suppose we can show that 15 is the multiplicative inverse of 7 modulo 26, i.e., that $15 \cdot 7 \equiv_{26} 1$

Then, we can multiply on both sides by 15 to see that

$$x \equiv_{26} 1x \equiv_{26} 15 \cdot 7x \equiv_{26} 15 \cdot 3 \equiv_{26} 45 \equiv_{26} 19$$

So, if we are given that $7x \equiv_{26} 3$, then we have shown that $x \equiv_{26} 19$.

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

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

$$26 = 3*7 + 5$$
$$7 = 1*5 + 2$$
$$5 = 2*2 + 1$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

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

$$26 = 3 * 7 + 5 \qquad 5 = 26 - 3 * 7$$
$$7 = 1 * 5 + 2 \qquad 2 = 7 - 1 * 5$$
$$5 = 2 * 2 + 1 \qquad 1 = 5 - 2 * 2$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

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

$$
\begin{aligned}
26 &= 3 * 7 + 5 &\quad 5 &= 26 - 3 * 7 \\
7 &= 1 * 5 + 2 &\quad 2 &= 7 - 1 * 5 \\
5 &= 2 * 2 + 1 &\quad 1 &= 5 - 2 * 2
\end{aligned}
$$

$$
\begin{aligned}
1 &= \quad 5 \quad - 2 * (7 - 1 * 5) \\
&= (-2) * 7 \quad + 3 * 5 \\
&= (-2) * 7 \quad + 3 * (26 - 3 * 7) \\
&= (-11) * 7 \quad + 3 * 26
\end{aligned}
$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

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

$$26 = 3 * 7 + 5 \qquad 5 = 26 - 3 * 7$$
$$7 = 1 * 5 + 2 \qquad 2 = 7 - 1 * 5$$
$$5 = 2 * 2 + 1 \qquad 1 = 5 - 2 * 2$$

$$1 = 5 - 2 * (7 - 1 * 5)$$
$$= (-2) * 7 + 3 * 5$$
$$= (-2) * 7 + 3 * (26 - 3 * 7)$$
$$= (-11) * 7 + 3 * 26$$

"**the**" multiplicative inverse

Now $(-11) \bmod 26 = 15$.   (-11 is also "a" multiplicative inverse)

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

We saw before that... if we are given that $7x \equiv_{26} 3$, then we have shown that $x \equiv_{26} 19$.

$$7x \equiv_{26} 3 \quad \Rightarrow \quad x \equiv_{26} 19$$

But these steps are *all* reversible...

Example: Solve a Modular Equation

$$7x \equiv_{26} 3 \quad \Rightarrow \quad 15 \cdot 7x \equiv_{26} 15 \cdot 3$$

multiply both sides by 15

$$\Rightarrow \quad x \equiv_{26} 19$$

since $15 \cdot 7 \equiv_{26} 1$ **and** $15 \cdot 3 \equiv_{26} 19$

$$x \equiv_{26} 19 \quad \Rightarrow \quad 7x \equiv_{26} 7 \cdot 19$$

multiply both sides by 7

$$\Rightarrow \quad 7x \equiv_{26} 3$$

since $7 \cdot 19 \equiv_{26} 3$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

We saw before that... if we are given that $7x \equiv_{26} 3$, then we have shown that $x \equiv_{26} 19$.

$$7x \equiv_{26} 3 \quad \Rightarrow \quad x \equiv_{26} 19$$

But all of these steps are reversible...

$$x \equiv_{26} 19 \quad \Rightarrow \quad 7x \equiv_{26} 7 \cdot 19$$

So $7x \equiv_{26} 3$ iff $x \equiv_{26} 19$

Hence, the solutions are all numbers of the form $19 + 26k$ for some integer

Beware the "Backward Proof"

- ## Many classes teach doing proofs backward:

$$15x + 5 = 5(-3x - 1) \qquad \text{start with we want to } \textit{prove}$$

$$(15x + 5)^2 = [5(-3x - 1)]^2 \qquad \text{do some manipulations}$$

$$\dots \qquad \dots$$

$$0 = 0 \qquad \text{end with an obvious truth}$$

- ## This is proof of nothing: "$P \to \mathrm{T}$" is a tautology
  - – it is true regardless of whether P is true

- ## See notes on the Resources page

Solving Modular Equations in "Standard Form"

Solve: $7x \equiv_{26} 3$ (of the form $Ax \equiv_m B$ for some A and B)

Step 1. Find multiplicative inverse of **7** modulo **26**

$$1 \;=\; \ldots \;=\; (-11) * 7 \;+\; 3 * 26$$

Since $(-11) \bmod 26 = 15$, **the inverse of** 7 **is** 15.

Step 2. Multiply both sides and simplify

Multiplying by 15, we get $x \equiv_{26} 15 \cdot 7x \equiv_{26} 15 \cdot 3 \equiv_{26} 19.$

Step 3. State the full set of solutions

So, the solutions are $19 + 26k$ for any integer k

(must be of the form $a + mk$ with $0 \le a < m$)

Example: Not in "Standard Form"

Solve: $7(x - 3) \equiv_{26} 8 + 2x$

What about equation not in standard form?

Example: Not in "Standard Form"

Solve: $7(x - 3) \equiv_{26} 8 + 2x$

Rewrite it in standard form:

$$7x - 21 \equiv_{26} 7(x - 3) \equiv_{26} 8 + 2x$$

move 2x **to the other side**

$$5x - 21 \equiv_{26} 8$$

move -21 **to the other side**

$$5x \equiv_{26} 29 \equiv_{26} 3$$

These steps are all reversible, so the solutions are the same.

Induction

Mathematical Induction

Method for proving claims about non-negative integers

- **A new logical inference rule!**
  - It only applies over the non-negative numbers
  - The idea is to **use** the special structure of these numbers to prove things more easily

Prove $\forall k \, ((a \equiv_m b) \to (a^k \equiv_m b^k))$

Let k be an arbitrary *non-negative* integer.

Suppose that $a \equiv_m b$.

We know $((a \equiv_m b) \land (a \equiv_m b)) \to (a^2 \equiv_m b^2)$ by multiplying congruences. So, applying this repeatedly, we have:

$$((a \equiv_m b) \land (a \equiv_m b)) \to (a^2 \equiv_m b^2)$$
$$\left((a^2 \equiv_m b^2) \land (a \equiv_m b)\right) \to (a^3 \equiv_m b^3)$$

$$\ldots$$
$$\left((a^{k-1} \equiv_m b^{k-1}) \land (a \equiv_m b)\right) \to (a^k \equiv_m b^k)$$

The "…"s is a problem! We don't have a proof rule that allows us to say "do this over and over".

But there is such a rule for non-negative numbers!

Domain: Non-Negative Numbers

$$\text{Induction} \quad \frac{P(0) \quad \forall k \, (P(k) \longrightarrow P(k+1))}{\therefore \, \forall n \; P(n)}$$

Induction Is A Rule of Inference

Domain: Non-Negative Numbers

$$P(0)$$
$$\forall k \ (P(k) \longrightarrow P(k+1))$$

Induction

$$\therefore \forall n \ P(n)$$

How do the givens prove P(3)?

Induction Is A Rule of Inference

Domain: Non-Negative Numbers

$$P(0)$$
$$\forall k \ (P(k) \longrightarrow P(k+1))$$

Induction

$$\therefore \forall n \ P(n)$$

How do the givens prove P(5)?

$$P(0) \quad P(1) \quad P(2) \quad P(3) \quad P(4) \quad P(5)$$

P(0)→P(1)   P(1)→P(2)   P(2)→P(3)   P(3)→P(4)   P(4)→P(5)

First, we have **P(0)**.

Since P(n) → P(n+1) for all n, we have **P(0)** → **P(1)**.

   Since **P(0)** is true and **P(0)** → **P(1)**, by Modus Ponens, **P(1)** is true.

Since P(n) → P(n+1) for all n, we have **P(1)** → **P(2)**.

   Since **P(1)** is true and **P(1)** → **P(2)**, by Modus Ponens, **P(2)** is true.

Using The Induction Rule In A Formal Proof

$$\text{Induction} \quad \dfrac{P(0) \quad \forall k \ (P(k) \longrightarrow P(k+1))}{\therefore \ \forall n \ P(n)}$$

Using The Induction Rule In A Formal Proof

$$\text{Induction} \quad \frac{P(0) \quad \forall k \ (P(k) \longrightarrow P(k+1))}{\therefore \ \forall n \ P(n)}$$

1. P(0)

2. $\forall$k (P(k) $\rightarrow$ P(k+1))   ??
3. $\forall$n P(n)   Induction: 1, 2

Using The Induction Rule In A Formal Proof

$$\text{Induction} \quad \frac{P(0) \quad \forall k \ (P(k) \longrightarrow P(k+1))}{\therefore \ \forall n \ P(n)}$$

1. P(0)

    Let k be an arbitrary integer ≥ 0

    2.1 P(k) $\longrightarrow$ P(k+1)   ??

2. $\forall k \ (P(k) \longrightarrow P(k+1))$ Intro $\forall$

3. $\forall n \ P(n)$ Induction: 1, 2

Using The Induction Rule In A Formal Proof

$$\text{Induction} \quad \frac{P(0) \quad \forall k\ (P(k) \longrightarrow P(k+1))}{\therefore\ \forall n\ P(n)}$$

1. P(0)

    Let k be an arbitrary integer ≥ 0

        2.1.1.   P(k)   Assumption

        2.1.2.   ...

        2.1.3.   P(k+1)

    2.1 P(k) $\longrightarrow$ P(k+1)   Direct Proof

2. $\forall$k (P(k) $\rightarrow$ P(k+1))   Intro $\forall$

3. $\forall$n P(n)   Induction: 1, 2

Translating to an English Proof

$$\text{Induction} \quad \frac{P(0) \quad \forall k \, (P(k) \longrightarrow P(k+1))}{\therefore \, \forall n \; P(n)}$$

1. Prove P(0) — **Base Case**

Let k be an arbitrary integer ≥ 0
  2.1.1. Suppose that P(k) is true — **Inductive Hypothesis**

  2.1.2. ...
  2.1.3. Prove P(k+1) is true — **Inductive Step**

2.1 P(k) → P(k+1)   Direct Proof
2. ∀k (P(k) → P(k+1))   Intro ∀
3. ∀n P(n)   Induction: 1, 2

Conclusion

Translating to an English Proof

1. Prove P(0)	**Base Case**
Let k be an arbitrary integer ≥ 0	**Inductive**
2.1.1. Suppose that P(k) is true	**Hypothesis**
2.1.2. ...	**Inductive**
2.1.3. Prove P(k+1) is true	**Step**
2.1 P(k) → P(k+1)	Direct Proof
2. ∀k (P(k) → P(k+1))	Intro ∀
3. ∀n P(n)	Induction: 1, 2

Conclusion

Induction English Proof Template

[...Define P(n)...]

We will show that $P(n)$ is true for every $n \geq 0$ by induction.

Base Case: *[...proof of $P(0)$ here...]*

Induction Hypothesis:
   Suppose that $P(k)$ is true for an arbitrary $k \geq 0$.

Induction Step:
   [...proof of $P(k+1)$ here...]
   *The proof of $P(k+1)$ **must** invoke the IH somewhere.*

So, the claim is true by induction.

Inductive Proofs In 5 Easy Steps

Proof:

1. "Let $P(n)$ be... . We will show that $P(n)$ is true for every $n \geq 0$ by Induction."

2. "Base Case:" Prove $P(0)$

3. "Inductive Hypothesis:

   Suppose $P(k)$ is true for an arbitrary integer $k \geq 0$"

4. "Inductive Step:" Prove that $P(k+1)$ is true.

   Use the goal to figure out what you need.

   Make sure you are using I.H. and point out where you are using it. (Don't assume $P(k+1)$!!)

5. "Conclusion: Result follows by induction"

What is $1 + 2 + 4 + ... + 2^n$?

- 1 $=$ 1
- $1 + 2$ $=$ 3
- $1 + 2 + 4$ $=$ 7
- $1 + 2 + 4 + 8$ $=$ 15
- $1 + 2 + 4 + 8 + 16$ $=$ 31

It sure looks like this sum is $2^{n+1} - 1$

How can we prove it?

We could prove it for $n = 1, n = 2, n = 3, ...$ but that would literally take forever.

Good that we have induction!

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

1. Let P(n) be "$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1$". We will show P(n) is **true** for all non-negative numbers by induction.

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

1. Let $P(n)$ be "$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1$". We will show $P(n)$ is true for all non-negative numbers by induction.
2. Base Case ($n=0$): $2^0 = 1 = 2 - 1 = 2^{0+1} - 1$ so $P(0)$ is true.

Prove $1 + 2 + 4 + \dots + 2^n = 2^{n+1} - 1$

1. Let $P(n)$ be "$2^0 + 2^1 + \dots + 2^n = 2^{n+1} - 1$". We will show $P(n)$ is true for all non-negative numbers by induction.

2. **Base Case** ($n=0$): $2^0 = 1 = 2 - 1 = 2^{0+1} - 1$ so $P(0)$ is true.

3. **Induction Hypothesis:** Suppose that $P(k)$ is true for some arbitrary integer $k \geq 0$, i.e., that $\boxed{2^0 + 2^1 + \dots + 2^k = 2^{k+1} - 1.}$

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

1. Let $P(n)$ be "$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1$". **We will show** $P(n)$ **is true for all non-negative numbers by induction.**

2. **Base Case** $(n=0)$: $2^0 = 1 = 2 - 1 = 2^{0+1} - 1$ **so** $P(0)$ **is true.**

3. **Induction Hypothesis: Suppose that** $P(k)$ **is true for some arbitrary integer** $k \geq 0$**, i.e., that** $2^0 + 2^1 + \ldots + 2^k = 2^{k+1} - 1$**.**

4. **Induction Step:**

    > **Goal: Show** $P(k+1)$**, i.e. show** $2^0 + 2^1 + \ldots + 2^k + 2^{k+1} = 2^{k+2} - 1$

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

1. Let $P(n)$ be "$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1$". We will show $P(n)$ is **true for all non-negative numbers by induction.**

2. **Base Case** $(n=0)$: $2^0 = 1 = 2 - 1 = 2^{0+1} - 1$ **so** $P(0)$ **is true.**

3. **Induction Hypothesis: Suppose that** $P(k)$ **is true for some arbitrary integer** $k \geq 0$, **i.e., that** $2^0 + 2^1 + \ldots + 2^k = 2^{k+1} - 1$.

4. **Induction Step:**

    $2^0 + 2^1 + \ldots + 2^k = 2^{k+1} - 1$ **by IH**

    Adding 2^{k+1} **to both sides, we get:**

    $2^0 + 2^1 + \ldots + 2^k + 2^{k+1} = 2^{k+1} + 2^{k+1} - 1$

    Note that $2^{k+1} + 2^{k+1} = 2(2^{k+1}) = 2^{k+2}$.

    So, we have $2^0 + 2^1 + \ldots + 2^k + 2^{k+1} = 2^{k+2} - 1$, **which is exactly** $P(k+1)$.

Calculation Block

We can do the same with equality:

$$2^0 + 2^1 + \ldots + 2^k + 2^{k+1}$$
$$= (2^0 + 2^1 + \ldots + 2^k) + 2^{k+1}$$
$$= (2^{k+1} - 1) + 2^{k+1} \qquad \text{since } 2^0 + 2^1 + \ldots + 2^k = 2^{k+1} - 1$$
$$= 2(2^{k+1}) - 1 \qquad \qquad \text{(better: "by the IH")}$$
$$= 2^{k+2} - 1$$

Explanations appear on in right column

- "since" means we substituted LHS for RHS
- ordinary algebra (on integers) does not need explanation
- "def of" will be used to apply the definition of a function
  - e.g., replacing f(x) by y when we have f defined as f(x) := y

Calculation Block

We can do the same with equality:

$$2^0 + 2^1 + \ldots + 2^k + 2^{k+1}$$
$$= (2^0 + 2^1 + \ldots + 2^k) + 2^{k+1}$$
$$= (2^{k+1} - 1) + 2^{k+1} \qquad \text{by the IH}$$
$$= 2(2^{k+1}) - 1$$
$$= 2^{k+2} - 1$$

Entire block shows $2^0 + 2^1 + \ldots + 2^k + 2^{k+1} = 2^{k+2} - 1$

– this is the transitivity property of "="

Can also do calculation with "<" and "≤"

– don't mix directions: ">" and "<" in one block is ><

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

1. Let $P(n)$ be "$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1$". We will show $P(n)$ is **true** for all non-negative numbers by induction.

2. **Base Case** (n=0): $2^0 = 1 = 2 - 1 = 2^{0+1} - 1$ so $P(0)$ is true.

3. **Induction Hypothesis: Suppose that** $P(k)$ **is true for some arbitrary integer** $k \geq 0$, **i.e., that** $2^0 + 2^1 + \ldots + 2^k = 2^{k+1} - 1$.

4. **Induction Step:**

   We can calculate

   $$\boxed{\textbf{Show } 2^0 + 2^1 + \ldots + 2^k + 2^{k+1} = 2^{k+2} - 1}$$

   $$
   \begin{aligned}
   2^0 + 2^1 + \ldots + 2^k + 2^{k+1} &= (2^0 + 2^1 + \ldots + 2^k) + 2^{k+1} \\
   &= (2^{k+1} - 1) + 2^{k+1} \qquad \textbf{by the IH} \\
   &= 2(2^{k+1}) - 1 \\
   &= 2^{k+2} - 1,
   \end{aligned}
   $$

   which is exactly $P(k+1)$.

   The entire inductive step is one calculation!

   We will rely heavily on calculation going forward…

Prove $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

1. Let $P(n)$ be "$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1$". We will show $P(n)$ is true for all non-negative numbers by induction.

2. Base Case ($n=0$): $2^0 = 1 = 2 - 1 = 2^{0+1} - 1$ so $P(0)$ is true.

3. Induction Hypothesis: Suppose that $P(k)$ is true for some arbitrary integer $k \geq 0$, i.e., that $2^0 + 2^1 + \ldots + 2^k = 2^{k+1} - 1$.

4. Induction Step:

   We can calculate

$$
\begin{aligned}
2^0 + 2^1 + \ldots + 2^k + 2^{k+1} &= (2^0 + 2^1 + \ldots + 2^k) + 2^{k+1} \\
&= (2^{k+1} - 1) + 2^{k+1} \qquad \text{by the IH} \\
&= 2(2^{k+1}) - 1 \\
&= 2^{k+2} - 1,
\end{aligned}
$$

   which is exactly $P(k+1)$.

5. Thus $P(n)$ is true for all $n \geq 0$, by induction.

Recall: Inductive Proofs In 5 Easy Steps

Proof:

1. "Let $P(n)$ be... . We will show that $P(n)$ is true for every $n \geq 0$ by Induction."

2. "Base Case:" Prove $P(0)$

3. "Inductive Hypothesis:

   Suppose $P(k)$ is true for an arbitrary integer $k \geq 0$"

4. "Inductive Step:" Prove that $P(k + 1)$ is true.

   Use the goal to figure out what you need.

   Make sure you are using I.H. and point out where you are using it. (Don't assume $P(k + 1)$!!)

5. "Conclusion: Result follows by induction"

Prove $1 + 2 + 3 + \dots + n = n(n+1)/2$

Prove that $\sum_{i=0}^{n} i = n(n+1)/2$

Summation Notation

$\sum_{i=0}^{n} i = 0 + 1 + 2 + 3 + \dots + n$

Prove $\sum_{i=0}^{n} i = n(n+1)/2$

1. Let P(n) be "$\sum_{i=0}^{n} i = n(n+1)/2$". **We will show** P(n) is true for all non-negative numbers by induction.

Prove $\sum_{i=0}^{n} i = n(n+1)/2$

1. Let P(n) be "$\sum_{i=0}^{n} i = n(n+1)/2$". **We will show** P(n) is true for all non-negative numbers by induction.

2. **Base Case** (n=0): $\sum_{i=0}^{0} i = 0 = 0(0+1)/2$, so P(0) **is true.**

Prove $\sum_{i=0}^{n} i = n(n+1)/2$

1. **Let** P(n) **be** "$\sum_{i=0}^{n} i = n(n+1)/2$". **We will show** P(n) **is true for all non-negative numbers by induction.**

2. **Base Case** (n=0): $\sum_{i=0}^{0} i = 0 = 0(0+1)/2$, so P(0) **is true.**

3. **Induction Hypothesis: Suppose that** P(k) **is true for some arbitrary integer** k ≥ 0**. I.e., suppose** $\sum_{i=0}^{k} i = k(k+1)/2$

"some" or "an"
not <u>any</u>!

Prove $\sum_{i=0}^{n} i = n(n+1)/2$

1. **Let** P(n) **be** "$\sum_{i=0}^{n} i = n(n+1)/2$". **We will show** P(n) **is true for all non-negative numbers by induction.**

2. **Base Case** (n=0): $\sum_{i=0}^{0} i = 0 = 0(0+1)/2$, so P(0) **is true.**

3. **Induction Hypothesis: Suppose that** P(k) **is true for some arbitrary integer** k ≥ 0. **I.e., suppose** $\sum_{i=0}^{k} i = k(k+1)/2$

4. **Induction Step:**

Goal: Show P(k+1), **i.e.,** $\sum_{i=0}^{k+1} i = (k+1)(k+2)/2$

Prove $\sum_{i=0}^{n} i = n(n+1)/2$

1. Let P(n) be "$\sum_{i=0}^{n} i = n(n+1)/2$". **We will show** P(n) is **true for all non-negative numbers by induction.**

2. **Base Case** (n=0): $\sum_{i=0}^{0} i = 0 = 0(0+1)/2$, so P(0) **is true.**

3. **Induction Hypothesis: Suppose that** P(k) **is true for some arbitrary integer** k ≥ 0. **I.e., suppose** $\sum_{i=0}^{k} i = k(k+1)/2$

4. **Induction Step: We can see that**

$$\sum_{i=0}^{k+1} i = (\sum_{i=0}^{k} i) + (k+1)$$
$$= k(k+1)/2 + (k+1) \qquad \text{by the IH}$$
$$= (k+1)(k/2+1)$$
$$= (k+1)(k+2)/2$$

which is exactly P(k+1)**.**

Prove $\sum_{i=0}^{n} i = n(n+1)/2$

1. Let P(n) be "$\sum_{i=0}^{n} i = n(n+1)/2$". **We will show** P(n) is true for all non-negative numbers by induction.

2. **Base Case** (n=0): $\sum_{i=0}^{0} i = 0 = 0(0+1)/2$, so P(0) is true.

3. **Induction Hypothesis: Suppose that** P(k) **is true for some arbitrary integer** k ≥ 0. **I.e., suppose** $\sum_{i=0}^{k} i = k(k+1)/2$

4. **Induction Step: We can see that**

$$\sum_{i=0}^{k+1} i = (\sum_{i=0}^{k} i) + (k+1)$$
$$= k(k+1)/2 + (k+1) \qquad \text{by the IH}$$
$$= (k+1)(k/2+1)$$
$$= (k+1)(k+2)/2$$

   which is exactly P(k+1).

5. **Thus** P(n) **is true for all** n ≥ 0, **by induction.**

Induction: Changing the starting point

- **What if we want to prove that $P(n)$ is true for all integers $n \geq b$ for some integer b?**

- **Define predicate $Q(k) = P(k + b)$ for all k.**
  - **Then $\forall n \, Q(n) \equiv \forall n \geq b \ P(n)$**

- **Ordinary induction for Q:**
  - **Prove $Q(0) \equiv P(b)$**
  - **Prove**
    $$\forall k \left(Q(k) \longrightarrow Q(k+1) \right) \equiv \forall k \geq b \left(P(k) \longrightarrow P(k+1) \right)$$

Inductive Proofs In 5 Easy Steps

1. "Let $P(n)$ be... . We will show that $P(n)$ is true for all integers $n \geq b$ by induction."

2. "Base Case:" Prove $P(b)$

3. "Inductive Hypothesis:

   Assume $P(k)$ is true for an arbitrary integer $k \geq b$"

4. "Inductive Step:" Prove that $P(k+1)$ is true:

   Use the goal to figure out what you need.

   Make sure you are using I.H. and point out where you are using it. (Don't assume $P(k+1)$!!)

5. "Conclusion: $P(n)$ is true for all integers $n \geq b$"

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

1. Let P(n) be "$3^n \geq n^2+3$". We will show P(n) is true for all integers $n \geq 2$ by induction.

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

1. Let P(n) be "$3^n \geq n^2+3$". We will show P(n) is true for all integers $n \geq 2$ by induction.

2. Base Case (n=2): $3^2 = 9 \geq 7 = 4+3 = 2^2+3$ so P(2) is true.

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

1. Let $P(n)$ be "$3^n \geq n^2+3$". We will show $P(n)$ is true for all integers $n \geq 2$ by induction.

2. Base Case ($n=2$): $3^2 = 9 \geq 7 = 4+3 = 2^2+3$ so $P(2)$ is true.

3. Inductive Hypothesis: Suppose that $P(k)$ is true for some arbitrary integer $k \geq 2$. I.e., suppose $3^k \geq k^2+3$.

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

1. Let P(n) be "$3^n \geq n^2+3$". We will show P(n) is true for all integers n ≥ 2 by induction.

2. Base Case (n=2): $3^2 = 9 \geq 7 = 4+3 = 2^2+3$ so P(2) is true.

3. Inductive Hypothesis: Suppose that P(k) is true for some arbitrary integer k ≥ 2. I.e., suppose $3^k \geq k^2+3$.

4. Inductive Step:

Goal: Show P(k+1), **i.e. show** $3^{k+1} \geq (k+1)^2+3=k^2+2k+4$

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

1. Let P(n) be "$3^n \geq n^2+3$". We will show P(n) is true for all integers $n \geq 2$ by induction.

2. Base Case (n=2): $3^2 = 9 \geq 7 = 4+3 = 2^2+3$ so P(2) is true.

3. Inductive Hypothesis: Suppose that P(k) is true for some arbitrary integer $k \geq 2$. I.e., suppose $3^k \geq k^2+3$.

4. Inductive Step: We can see that

$$3^{k+1} = 3(3^k)$$
$$\geq 3(k^2+3) \qquad \text{by the IH}$$
$$= k^2+2k^2+9$$
$$\geq k^2+2k+9 \qquad \text{since } k^2 \geq k$$
$$\geq k^2+2k+4 \qquad \text{since } 9 \geq 4$$
$$= (k+1)^2+3$$

Therefore P(k+1) is true.

Prove $3^n \geq n^2 + 3$ for all $n \geq 2$

1. Let $P(n)$ be "$3^n \geq n^2+3$". We will show $P(n)$ is true for all integers $n \geq 2$ by induction.

2. Base Case ($n=2$): $3^2 = 9 \geq 7 = 4+3 = 2^2+3$ so $P(2)$ is true.

3. Inductive Hypothesis: Suppose that $P(k)$ is true for some arbitrary integer $k \geq 2$. I.e., suppose $3^k \geq k^2+3$.

4. Inductive Step: We can see that

$$3^{k+1} = 3(3^k) \geq 3(k^2+3) \qquad \text{by the IH}$$

$$= k^2+2k^2+9$$

$$\geq k^2+2k+9 \qquad \text{since } k^2 \geq k$$

$$\geq k^2+2k+4 \qquad \text{since } 9 \geq 4$$

$$= (k+1)^2+3$$

Therefore $P(k+1)$ is true.

5. Thus $P(n)$ is true for all integers $n \geq 2$, by induction.

Induction: Adding Base Cases

- **What if we want to prove that $P(n)$ is true for all integers $n \geq b$ for some integer b but the inductive step only works for $n \geq c$?**

- **Add proofs of $P(b), P(b+1), ..., P(c-1)$**
  - will call these extra "base cases"

- **Formally, we are using the fact that**

  $$P(b) \land \cdots \land P(c-1) \land \forall n \, ((c \leq n) \to P(n))$$
  $$\equiv \forall n \, ((b \leq n) \to P(n))$$

Inductive Proofs In 5 Easy Steps

1. "Let $P(n)$ be... . We will show that $P(n)$ is true for all integers $n \geq b$ by induction."

2. "Base Case:" Prove $P(b)$, ..., $P(c)$

3. "Inductive Hypothesis:

    Assume $P(k)$ is true for an arbitrary integer $k \geq c$"

4. "Inductive Step:" Prove that $P(k+1)$ is true:

    Use the goal to figure out what you need.

    Make sure you are using I.H. and point out where you are using it. (Don't assume $P(k+1)$!!)

5. "Conclusion: $P(n)$ is true for all integers $n \geq b$"

Recursive Definitions of Functions

Familiar Recursive Definitions

Suppose that $h : \mathbb{N} \to \mathbb{R}$.

Then we have familiar summation notation:

$\sum_{i=0}^{0} h(i) := h(0)$

$\sum_{i=0}^{n+1} h(i) := \left(\sum_{i=0}^{n} h(i)\right) + h(n+1)$ **for** $n \geq 0$

There is also product notation:

$\prod_{i=0}^{0} h(i) := h(0)$

$\prod_{i=0}^{n+1} h(i) := \left(\prod_{i=0}^{n} h(i)\right) \cdot h(n+1)$ **for** $n \geq 0$

Recursive definitions of functions

- $0! := 1;\ (n+1)! := (n+1) \cdot n!$ **for all** $n \geq 0$.

- $F(0) := 0;\ F(n+1) := F(n) + 1$ **for all** $n \geq 0$.

- $G(0) := 1;\ G(n+1) := 2 \cdot G(n)$ **for all** $n \geq 0$.

- $H(0) := 1;\ H(n+1) := 2^{H(n)}$ **for all** $n \geq 0$.

Prove $n! \leq n^n$ for all $n \geq 1$

1. Let $P(n)$ be "$n! \leq n^n$". We will show that $P(n)$ is true for all integers $n \geq 1$ by induction.

Prove $n! \leq n^n$ for all $n \geq 1$

1. Let P(n) be "$n! \leq n^n$". We will show that P(n) is true for all integers $n \geq 1$ by induction.

2. Base Case (n=1): $1! = 1 \cdot 0! = 1 \cdot 1 = 1 = 1^1$ so P(1) is true.

Prove $n! \leq n^n$ for all $n \geq 1$

1. Let P(n) be "$n! \leq n^n$". We will show that P(n) is true for all integers $n \geq 1$ by induction.

2. Base Case (n=1): $1! = 1 \cdot 0! = 1 \cdot 1 = 1 = 1^1$ so P(1) is true.

3. Inductive Hypothesis: Suppose that P(k) is true for some arbitrary integer $k \geq 1$. I.e., suppose $k! \leq k^k$.

Prove $n! \leq n^n$ for all $n \geq 1$

1. Let P(n) be "$n! \leq n^n$". We will show that P(n) is true for all integers n ≥ 1 by induction.

2. Base Case (n=1): $1! = 1 \cdot 0! = 1 \cdot 1 = 1 = 1^1$ so P(1) is true.

3. Inductive Hypothesis: Suppose that P(k) is true for some arbitrary integer k ≥ 1. I.e., suppose $k! \leq k^k$.

4. Inductive Step:

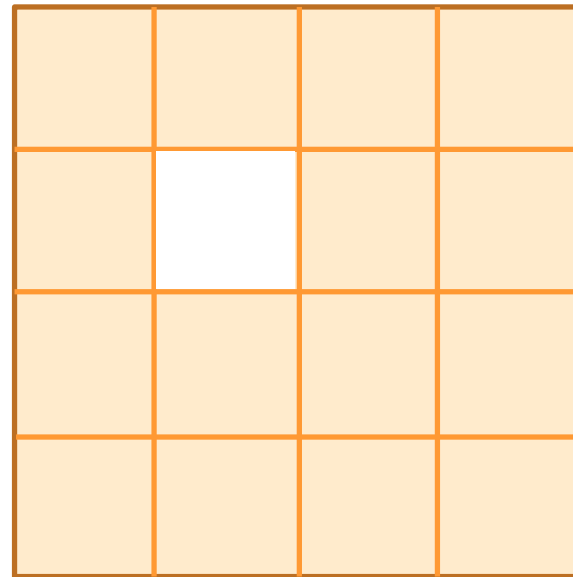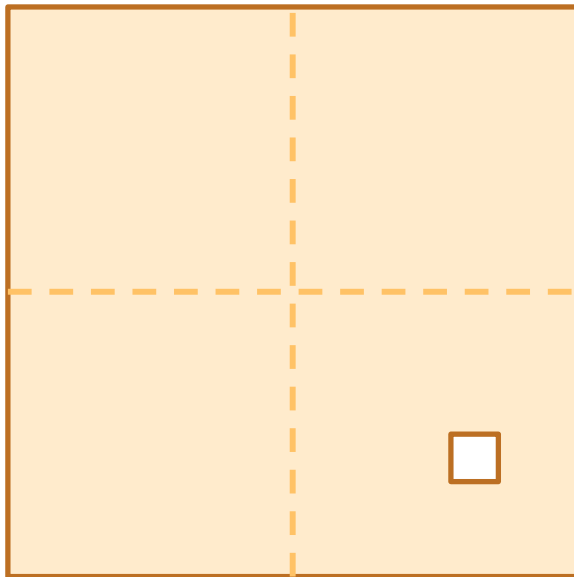Goal: Show P(k+1), i.e. show $(k+1)! \leq (k+1)^{k+1}$

Prove $n! \leq n^n$ for all $n \geq 1$

1. Let P(n) be "$n! \leq n^n$". We will show that P(n) is true for all integers $n \geq 1$ by induction.

2. Base Case (n=1): $1! = 1 \cdot 0! = 1 \cdot 1 = 1 = 1^1$ so P(1) is true.

3. Inductive Hypothesis: Suppose that P(k) is true for some arbitrary integer $k \geq 1$. I.e., suppose $k! \leq k^k$.
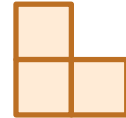
4. Inductive Step:

   We can calculate:

$$(k+1)! = (k+1) \cdot k! \qquad \text{by definition of !}$$
$$\leq (k+1) \cdot k^k \qquad \text{by the IH}$$
$$\leq (k+1) \cdot (k+1)^k \qquad \text{since } k \geq 0$$
$$= (k+1)^{k+1}$$

   Therefore P(k+1) is true.

Prove $n! \leq n^n$ for all $n \geq 1$

1. Let P(n) be "$n! \leq n^n$". We will show that P(n) is true for all integers $n \geq 1$ by induction.

2. Base Case (n=1): $1! = 1 \cdot 0! = 1 \cdot 1 = 1 = 1^1$ so P(1) is true.

3. Inductive Hypothesis: Suppose that P(k) is true for some arbitrary integer $k \geq 1$. I.e., suppose $k! \leq k^k$.

4. Inductive Step:

   We can calculate:

$$(k+1)! = (k+1) \cdot k! \qquad \text{by definition of !}$$
$$\leq (k+1) \cdot k^k \qquad \text{by the IH}$$
$$\leq (k+1) \cdot (k+1)^k \quad \text{since } k \geq 0$$
$$= (k+1)^{k+1}$$

   Therefore P(k+1) is true.

5. Thus P(n) is true for all $n \geq 1$, by induction.

Checkerboard Tiling

- **Prove that a $2^n \times 2^n$ checkerboard with one square removed can be tiled with:**

Checkerboard Tiling

1. Let $P(n)$ be any $2^n \times 2^n$ checkerboard with one square removed can be tiled with ⌐ .
   We prove $P(n)$ for all $n \geq 1$ by induction on n.
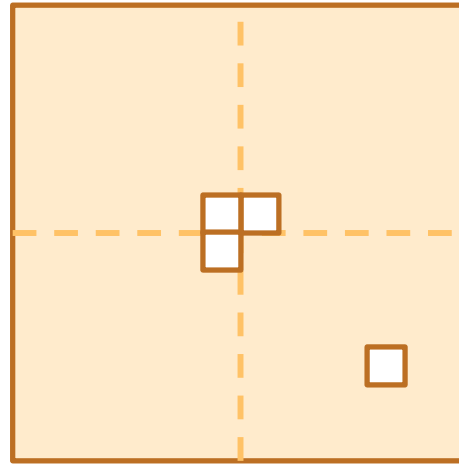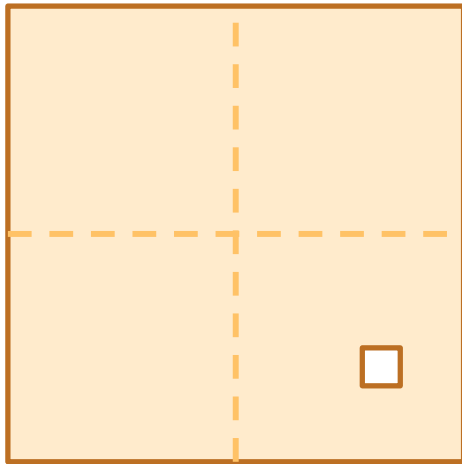
Checkerboard Tiling

1. Let $P(n)$ be any $2^n \times 2^n$ checkerboard with one square removed can be tiled with ▭.
    We prove $P(n)$ for all $n \geq 1$ by induction on n.

2. Base Case: $n=1$

Checkerboard Tiling

1. Let $P(n)$ be any $2^n \times 2^n$ checkerboard with one square removed can be tiled with ⌐ .
   We prove $P(n)$ for all $n \geq 1$ by induction on n.

2. Base Case: $n=1$

3. Inductive Hypothesis: Assume $P(k)$ for some
   arbitrary integer $k \geq 1$

Checkerboard Tiling

1. Let $P(n)$ be any $2^n \times 2^n$ checkerboard with one square removed can be tiled with ⌐ .
   We prove $P(n)$ for all $n \geq 1$ by induction on n.

2. Base Case: n=1

3. Inductive Hypothesis: Assume $P(k)$ for some arbitrary integer $k \geq 1$

4. Inductive Step: Prove $P(k+1)$

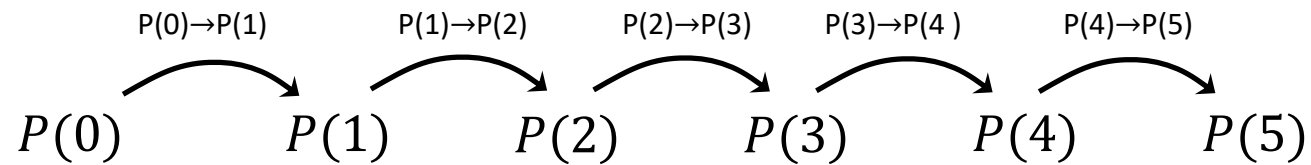Apply IH to each quadrant then fill with extra tile.

Induction Is A Rule of Inference

Domain: Non-Negative Numbers

$$P(0)$$
$$\forall k \ (P(k) \longrightarrow P(k+1))$$

Induction

$$\therefore \forall n \ P(n)$$

How do the givens prove P(5)?

P(0)→P(1)   P(1)→P(2)   P(2)→P(3)   P(3)→P(4)   P(4)→P(5)

$P(0)$ $P(1)$ $P(2)$ $P(3)$ $P(4)$ $P(5)$

Induction Is A Rule of Inference

Domain: Non-Negative Numbers

$$P(0)$$
$$\forall k \ (P(k) \longrightarrow P(k+1))$$

Induction

$$\therefore \forall n \ P(n)$$

How do the givens prove P(5)?

| P(0)→P(1) | P(1)→P(2) | P(2)→P(3) | P(3)→P(4) | P(4)→P(5) |

$$P(0) \qquad P(1) \qquad P(2) \qquad P(3) \qquad P(4) \qquad P(5)$$

First, we have P(0).
Since P(n) → P(n+1) for all n, we have P(0) → P(1).
   Since P(0) is true and P(0) → P(1), by Modus Ponens, P(1) is true.
Since P(n) → P(n+1) for all n, we have P(1) → P(2).
   Since P(1) is true and P(1) → P(2), by Modus Ponens, P(2) is true.

Strong Induction

$$P(0) \qquad \forall k\,(\,P(k) \longrightarrow P(k+1))$$

Induction
$$\therefore \forall n\, P(n)$$

$$P(0) \qquad \forall k\,\Big(\forall j\,\big((0 \le j \le k) \to P(j)\big) \to P(k+1)\Big)$$

Strong Induction
$$\therefore \forall n\, P(n)$$

Strong Induction

$$P(0) \qquad \forall k \left(\forall j \left(0 \leq j \leq k \to P(j) \right) \to P(k+1) \right)$$
$$\therefore \forall n\, P(n)$$

Strong induction for P follows from ordinary induction for Q where

$$Q(k) \; ::= \; \forall j \left(0 \leq j \leq k \to P(j) \right)$$

Note that $Q(0) = P(0)$ and $Q(k+1) \equiv Q(k) \wedge P(k+1)$ and $\forall n\, Q(n) \equiv \forall n\, P(n)$

Strong Inductive Proofs In 5 Easy Steps

1. "Let $P(n)$ be... . We will show that $P(n)$ is true for all integers $n \geq b$ by *strong* induction."

2. "Base Case:" Prove $P(b)$

3. "Inductive Hypothesis:

   Assume that for some arbitrary integer $k \geq b$,

   $P(j)$ is true for every integer j from b to k"

4. "Inductive Step:" Prove that $P(k+1)$ is true:

   Use the goal to figure out what you need.

   Make sure you are using I.H. (that $P(b), ..., P(k)$ are true) and point out where you are using it.
   (Don't assume $P(k+1)$!!)

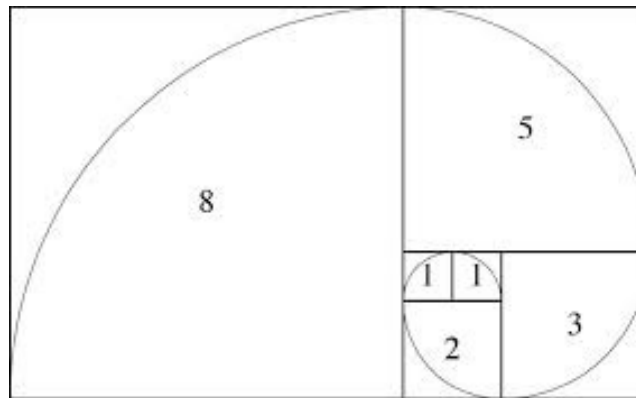5. "Conclusion: $P(n)$ is true for all integers $n \geq b$"

Fibonacci Numbers

$$f_0 := 0$$
$$f_1 := 1$$
$$f_{n+2} := f_{n+1} + f_n$$

Will need facts about f_{n-2} to reason about f_n

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by **strong** induction.

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by **strong** induction.

2. Base Case: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true.

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let P(n) be "$f_n < 2^n$". We prove that P(n) is true for all integers $n \geq 0$ by strong induction.

2. Base Case: $f_0 = 0 < 1 = 2^0$ so P(0) is true.

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 0$, we have $f_j < 2^j$ for every integer j from 0 to k.

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by strong induction.

2. Base Case: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true.

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 0$, we have $f_j < 2^j$ for every integer j from 0 to k.

4. Inductive Step:

$$f_{k+1} = f_k + f_{k-1} \qquad \text{def of f}$$

Oops! This is only true if $k + 1 \geq 2$!

Goal: Show $P(k+1)$; that is, $f_{k+1} < 2^{k+1}$

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by **strong** induction.

2. Base Cases: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true and
$$f_1 = 1 < 2 = 2^1 \text{ so } P(1) \text{ is true.}$$

$$\boxed{\begin{array}{ll} f_0 = 0 & f_1 = 1 \\ f_{n+2} = f_{n+1} + f_n \end{array}}$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by strong induction.

2. Base Cases: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true and
   $f_1 = 1 < 2 = 2^1$ so $P(1)$ is true.

3. Inductive Hypothesis: Assume that for **some** arbitrary integer $k \geq 1$, we have $f_j < 2^j$ for **every** integer j from 0 to k.

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by strong induction.

2. Base Cases: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true and
   $$f_1 = 1 < 2 = 2^1 \text{ so } P(1) \text{ is true.}$$

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 1$, we have $f_j < 2^j$ for every integer j from 0 to k.

4. Inductive Step:

Goal: Show $P(k+1)$; that is, $f_{k+1} < 2^{k+1}$

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by strong induction.

2. Base Cases: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true and
$$f_1 = 1 < 2 = 2^1 \text{ so } P(1) \text{ is true.}$$

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 1$, we have $f_j < 2^j$ for every integer j from 0 to k.

4. Inductive Step: We can calculate that

$$f_{k+1} = f_k + f_{k-1} \qquad \text{def of f (since } k+1 \geq 2)$$
$$< 2^k + 2^{k-1} \qquad \text{by IH (since } k-1 \geq 0)$$
$$< 2^k + 2^k$$
$$= 2 \cdot 2^k$$
$$= 2^{k+1}$$

so $P(k+1)$ is true.

$$\boxed{\begin{array}{ll} f_0 = 0 & f_1 = 1 \\ f_{n+2} = f_{n+1} + f_n & \end{array}}$$

Bounding Fibonacci: $f_n < 2^n$ for all $n \geq 0$

1. Let $P(n)$ be "$f_n < 2^n$". We prove that $P(n)$ is true for all integers $n \geq 0$ by strong induction.

2. Base Cases: $f_0 = 0 < 1 = 2^0$ so $P(0)$ is true and
   $$f_1 = 1 < 2 = 2^1 \text{ so } P(1) \text{ is true.}$$

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 1$, we have $f_j < 2^j$ for every integer j from 0 to k.

4. Inductive Step: We can calculate that

   $$f_{k+1} = f_k + f_{k-1} \qquad \text{def of f (since } k+1 \geq 2)$$
   $$< 2^k + 2^{k-1} \qquad \text{by IH (since } k-1 \geq 0)$$
   $$< 2^k + 2^k$$
   $$= 2^{k+1}$$

   so $P(k+1)$ is true.

5. Therefore, by strong induction, $f_n < 2^n$ for all integers $n \geq 0$.

Strong Inductive Proofs In 5 Easy Steps

1. "Let $P(n)$ be... . We will show that $P(n)$ is true for all integers $n \geq b$ by *strong* induction."

2. "Base Case:" Prove $P(b)$

3. "Inductive Hypothesis:

   Assume that for some arbitrary integer $k \geq b$,

   $P(j)$ is true for every integer j from b to k"

4. "Inductive Step:" Prove that $P(k+1)$ is true:

   Use the goal to figure out what you need.

   Make sure you are using I.H. (that $P(b), \ldots, P(k)$ are true) and point out where you are using it.
   (Don't assume $P(k+1)$!!)

5. "Conclusion: $P(n)$ is true for all integers $n \geq b$"

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. Let $P(n)$ be "$f_n \geq 2^{n/2 - 1}$". We prove that $P(n)$ is true for all integers $n \geq 2$ by **strong** induction.

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. Let $P(n)$ be "$f_n \geq 2^{n/2 - 1}$". We prove that $P(n)$ is true for all integers $n \geq 2$ by strong induction.

2. Base Case: $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2 - 1}$ so $P(2)$ holds

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. Let P(n) be "$f_n \geq 2^{n/2 - 1}$". We prove that P(n) is true for all integers n ≥ 2 by strong induction.

2. Base Case: $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2 - 1}$ so P(2) holds

3. Inductive Hypothesis: Assume that for some arbitrary integer k ≥ 2, P(j) is true for every integer j from 2 to k.

$$\boxed{\begin{array}{l} f_0 = 0 \quad f_1 = 1 \\ f_{n+2} = f_{n+1} + f_n \end{array}}$$

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. Let P(n) be "$f_n \geq 2^{n/2 - 1}$". We prove that P(n) is true for all integers n ≥ 2 by strong induction.

2. Base Case: $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2 - 1}$ so P(2) holds

3. Inductive Hypothesis: Assume that for some arbitrary integer k ≥ 2, P(j) is true for every integer j from 2 to k.

4. Inductive Step:

Goal: Show P(k+1); that is, $f_{k+1} \geq 2^{(k+1)/2 - 1}$

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. Let P(n) be "$f_n \geq 2^{n/2 - 1}$". We prove that P(n) is true for all integers $n \geq 2$ by strong induction.

2. Base Case: $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2 - 1}$ so P(2) holds

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 2$, P(j) is true for every integer j from 2 to k.

4. Inductive Step:

$$f_{k+1} = f_k + f_{k-1} \qquad \text{def of f (since k+1 $\geq$ 2)}$$

$$\geq 2^{k/2-1} + f_{k-1} \qquad \text{by the IH}$$

$$\geq 2^{k/2-1} + 2^{(k-1)/2-1} \qquad \text{by the IH}$$

Oops! This is only true if $k - 1 \geq 2$!

Goal: Show P(k+1); that is, $f_{k+1} \geq 2^{(k+1)/2 - 1}$

$$f_0 = 0 \qquad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci II: $f_n \geq 2^{n/2-1}$ for all $n \geq 2$

1. Let P(n) be "$f_n \geq 2^{n/2-1}$". We prove that P(n) is true for all integers n ≥ 2 by strong induction.

2. Base Cases: $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2-1}$ so P(2) holds
   $f_3 = f_2 + f_1 = 2 \geq 2^{1/2} = 2^{3/2-1}$ so P(3) holds

3. Inductive Hypothesis: Assume that for some arbitrary integer k ≥ 3, P(j) is true for every integer j from 2 to k.

4. Inductive Step:

Goal: Show P(k+1); that is, $f_{k+1} \geq 2^{(k+1)/2-1}$

$$f_0 = 0 \quad f_1 = 1$$
$$f_{n+2} = f_{n+1} + f_n$$

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. Let $P(n)$ be "$f_n \geq 2^{n/2 - 1}$". We prove that $P(n)$ is true for all integers $n \geq 2$ by strong induction.

2. Base Cases: $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2 - 1}$ so $P(2)$ holds
   $f_3 = f_2 + f_1 = 2 \geq 2^{1/2} = 2^{3/2 - 1}$ so $P(3)$ holds

3. Inductive Hypothesis: Assume that for some arbitrary integer $k \geq 3$, $P(j)$ is true for every integer j from 2 to k.

4. Inductive Step: We can calculate

$$f_{k+1} = f_k + f_{k-1} \qquad \text{def of f (since } k+1 \geq 4)$$
$$\geq 2^{k/2-1} + f_{k-1} \qquad \text{by the IH}$$
$$\geq 2^{k/2-1} + 2^{(k-1)/2-1} \qquad \text{by the IH (since } k-1 \geq 2)$$
$$\geq 2 \cdot 2^{(k-1)/2-1}$$
$$= 2^{(k+1)/2-1}$$

so $P(k+1)$ is true.

Bounding Fibonacci II: $f_n \geq 2^{n/2 - 1}$ for all $n \geq 2$

1. **Let** $P(n)$ **be** "$f_n \geq 2^{n/2 - 1}$". **We prove that** $P(n)$ **is true for all integers** $n \geq 2$ **by strong induction.**

2. **Base Cases:** $f_2 = f_1 + f_0 = 1 \geq 1 = 2^0 = 2^{2/2 - 1}$ **so** $P(2)$ **holds**
   $f_3 = f_2 + f_1 = 2 \geq 2^{1/2} = 2^{3/2 - 1}$ **so** $P(3)$ **holds**

3. **Inductive Hypothesis: Assume that for some arbitrary integer** $k \geq 3$, $P(j)$ **is true for every integer** j **from 2 to** k.

4. **Inductive Step: We can calculate**

   $\begin{aligned}
   f_{k+1} = f_k \ + \ f_{k-1} \qquad &\textbf{def of f (since } k+1 \geq 4\textbf{)} \\
   \geq 2^{k/2-1} + f_{k-1} \qquad &\textbf{by the IH} \\
   \geq 2^{k/2-1} + 2^{(k-1)/2-1} \qquad &\textbf{by the IH (since } k-1 \geq 2\textbf{)} \\
   \geq 2 \cdot 2^{(k-1)/2-1} = 2^{(k+1)/2 - 1} &
   \end{aligned}$

   so $P(k+1)$ **is true.**

5. **Therefore by strong induction,** $f_n \geq 2^{n/2 - 1}$ **for all integers** $n \geq 2$.