CSE 311 Section 07

Structural Induction, Set Theory

Administrivia

Announcements & Reminders

- Midterm
  - Good job!
  - Please don't talk about the midterm!! Not everyone has taken it yet ☺

- HW5 Regrade Requests
  - Regrade request window open as usual for Part 1 & 2
  - If something was graded incorrectly, submit a regrade request

- HW6
  - Due Wednesday 2/21 @ 11:59pm

Sets

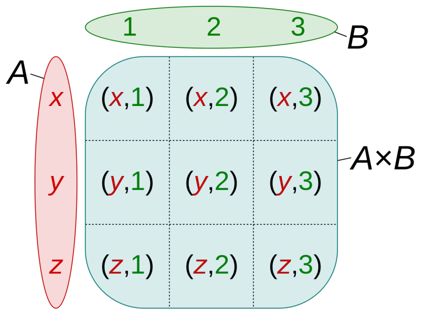
Sets

- A set is an **unordered** group of **distinct** elements
  - Set variable names are capital letters, with lower-case letters for elements

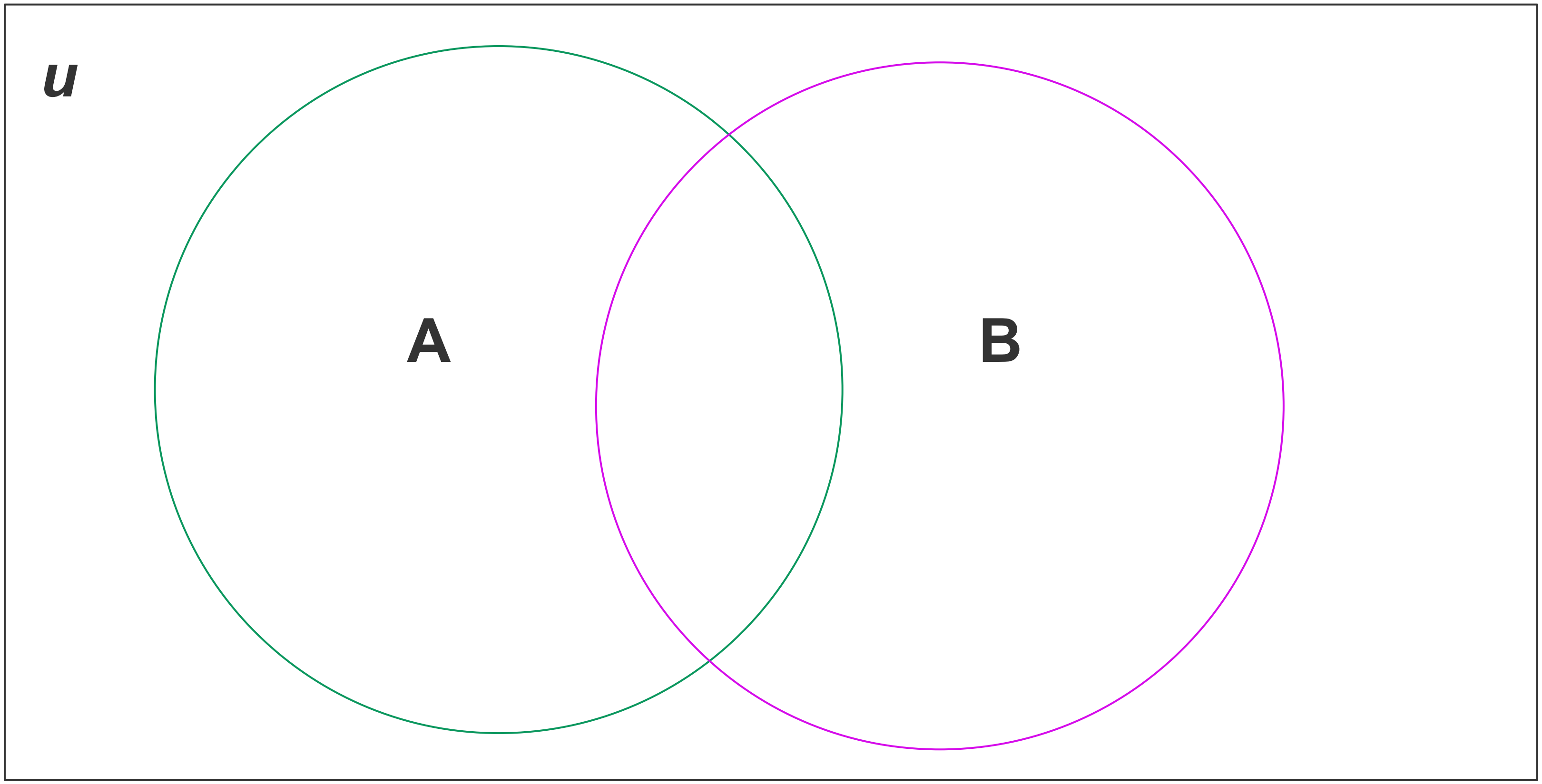
- Set Notation:
  - $a \in A$: "a is in A" or "a is an element of A"
  - $A \subseteq B$: "A is a subset of B", every element of A is also in B
  - $\emptyset$: "empty set", a unique set containing no elements
  - $\mathcal{P}(A)$: "power set of A", the set of all subsets of A including the empty set and A itself

Set Operators

- Subset: $A \subseteq B \equiv \forall x(x \in A \rightarrow x \in B)$
- Equality: $A = B \equiv \forall x(x \in A \leftrightarrow x \in B) \equiv A \subseteq B \wedge B \subseteq A$
- Union: $A \cup B = \{x : x \in A \vee x \in B\}$
- Intersection: $A \cap B = \{x : x \in A \wedge x \in B\}$
- Complement: $\overline{A} = \{x : x \notin A\}$
- Difference: $A \backslash \mathrm{B} = \{x : x \in A \wedge x \notin B\}$
- Cartesian Product: $A \times B = \{(a, b) : a \in A \wedge b \in B\}$

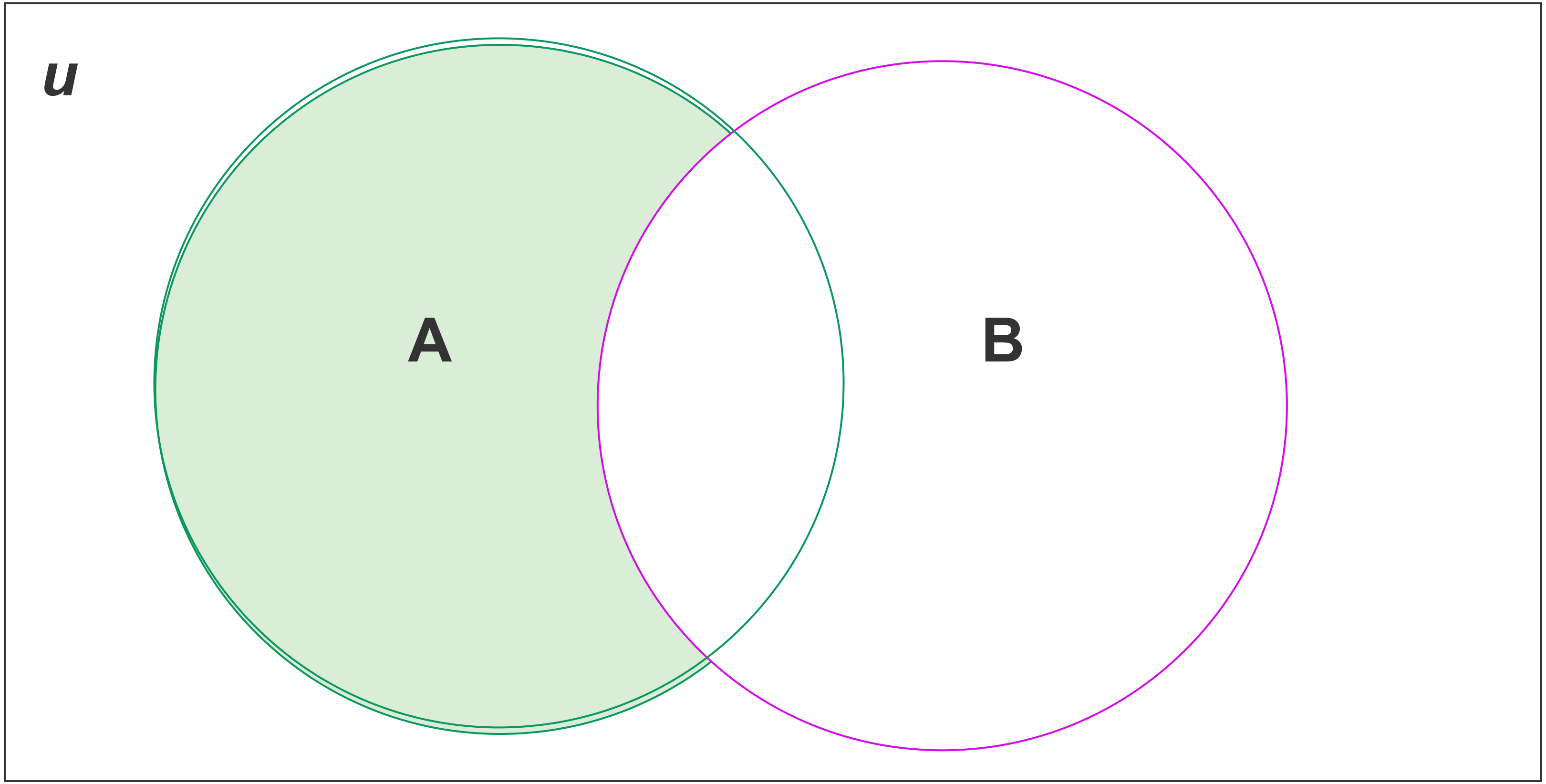


Understand Sets Visually!



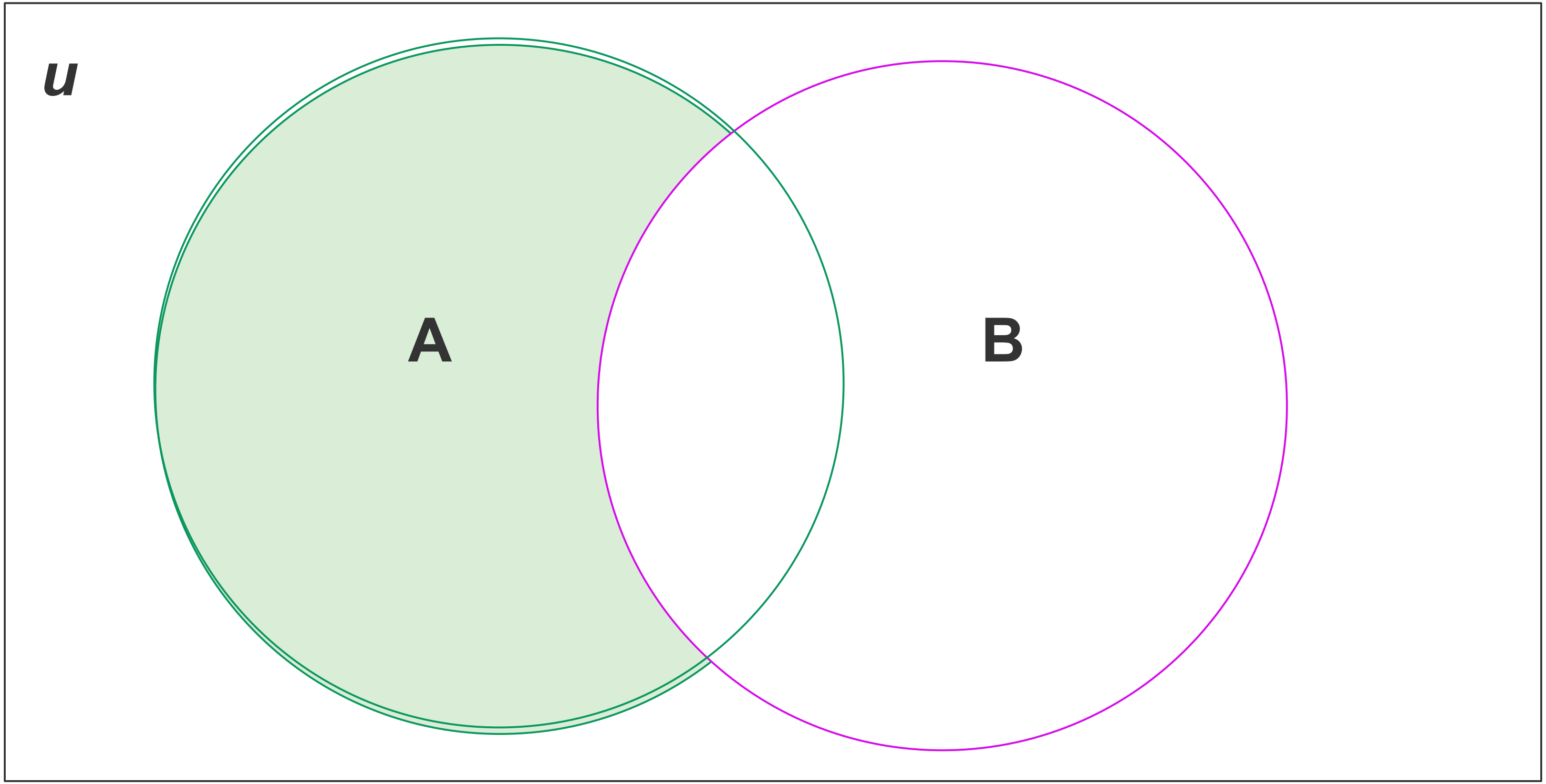
Understand Sets Visually!

What Set Operation is this?

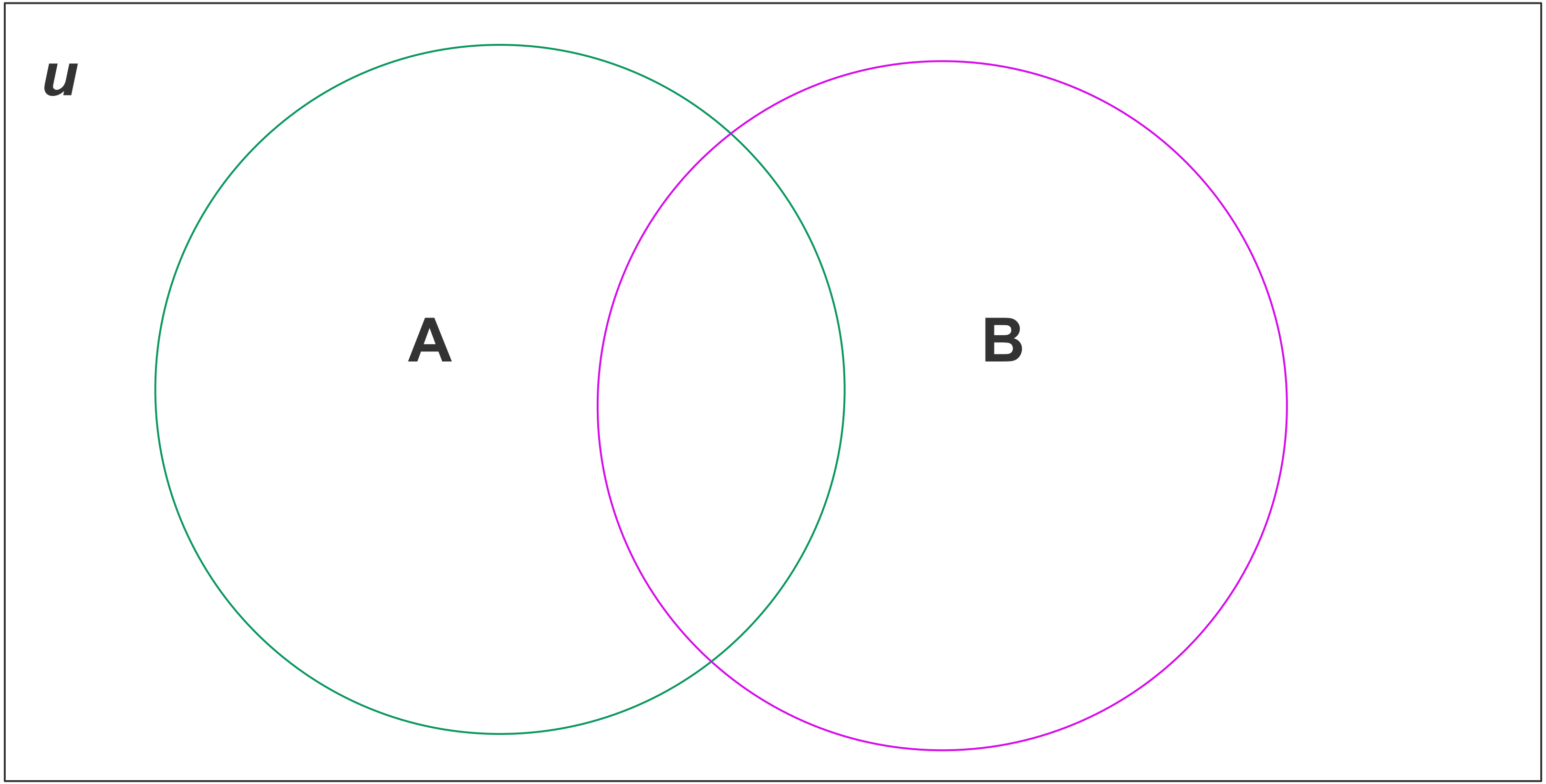


Understand Sets Visually!

What Set Operation is this?
Difference: A \ B

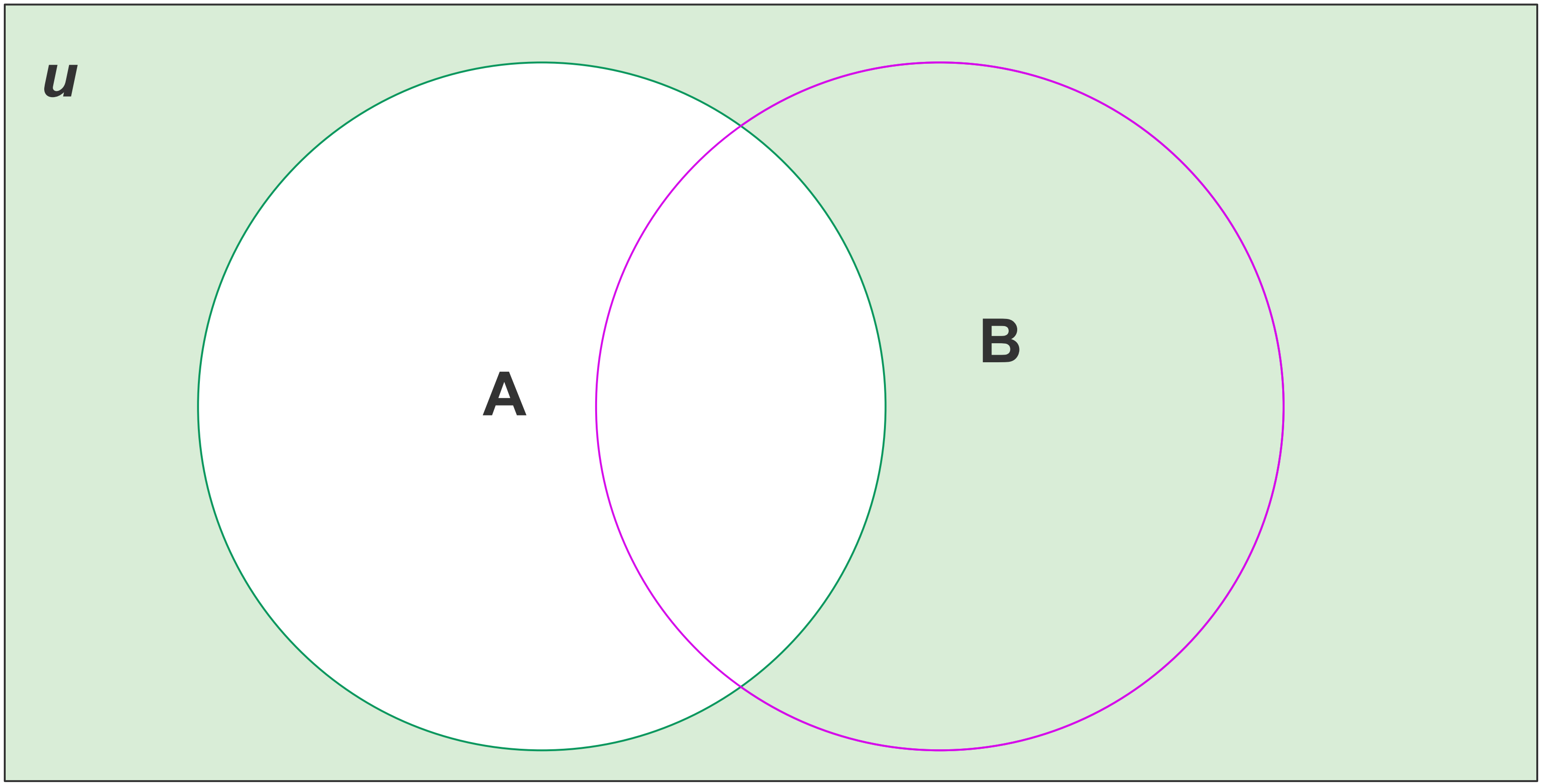


Understand Sets Visually!



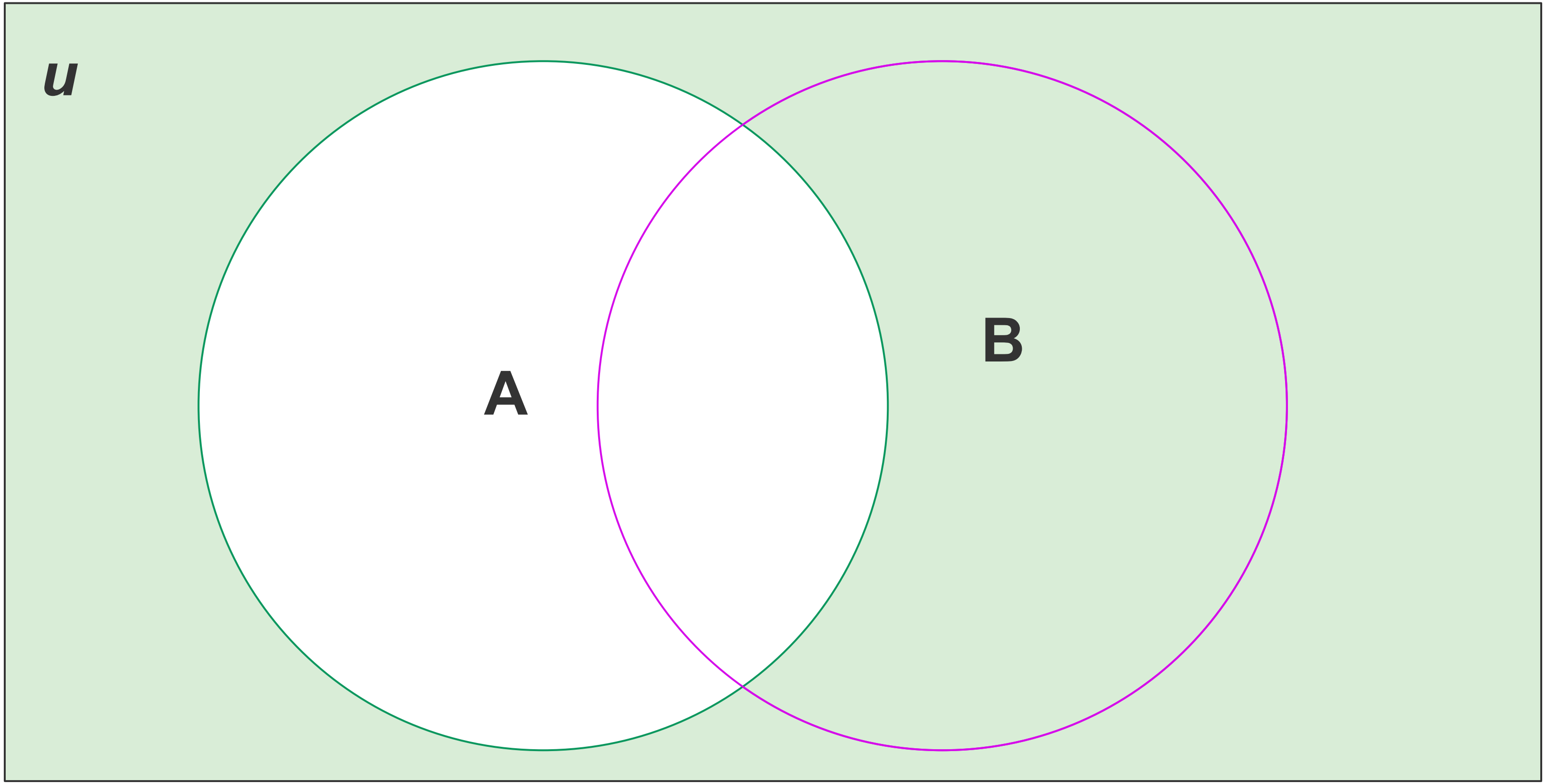
Understand Sets Visually!

What Set Operation is this?

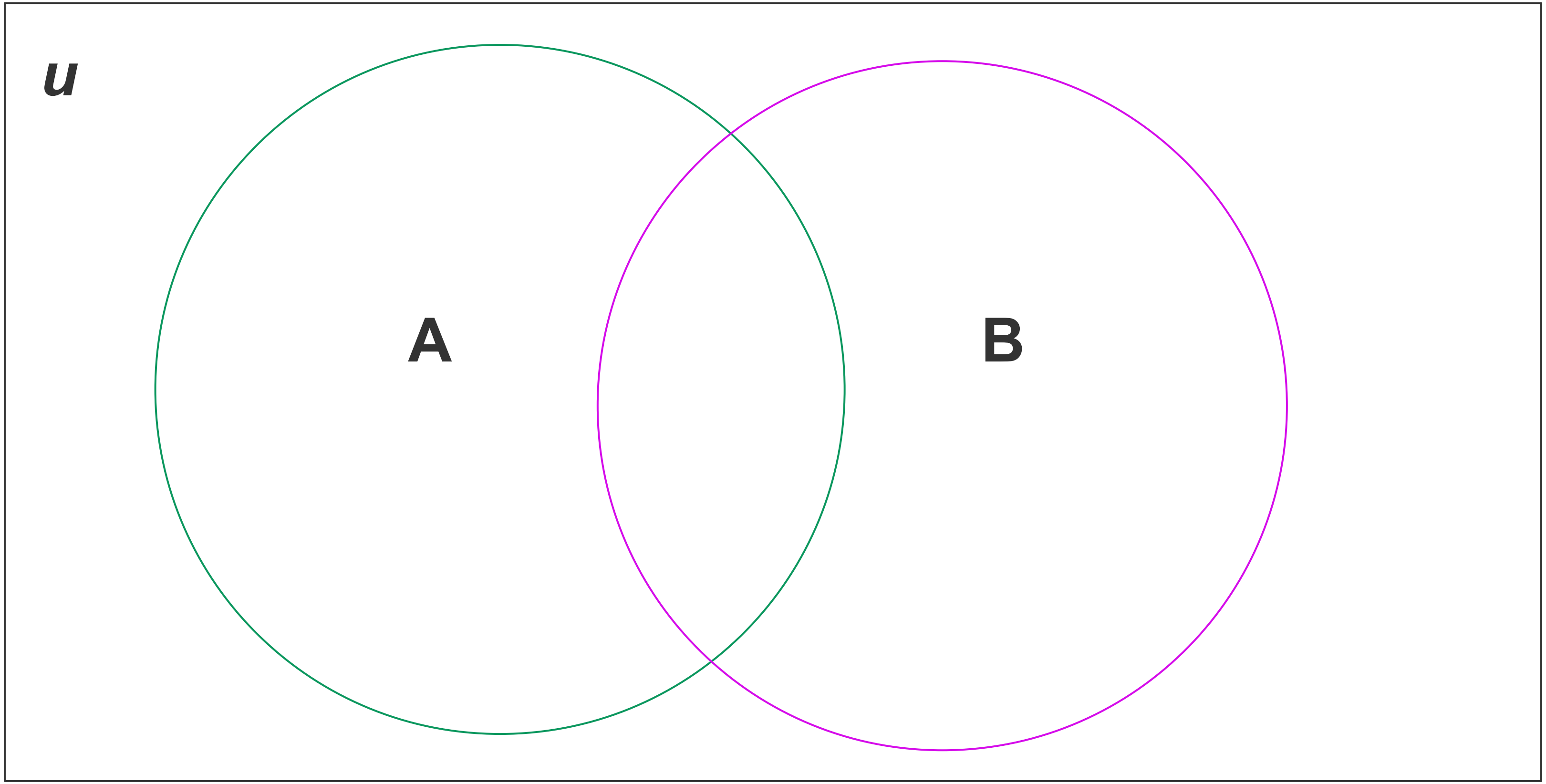


Understand Sets Visually!

What Set Operation is this?
A complement: $\overline{A}$

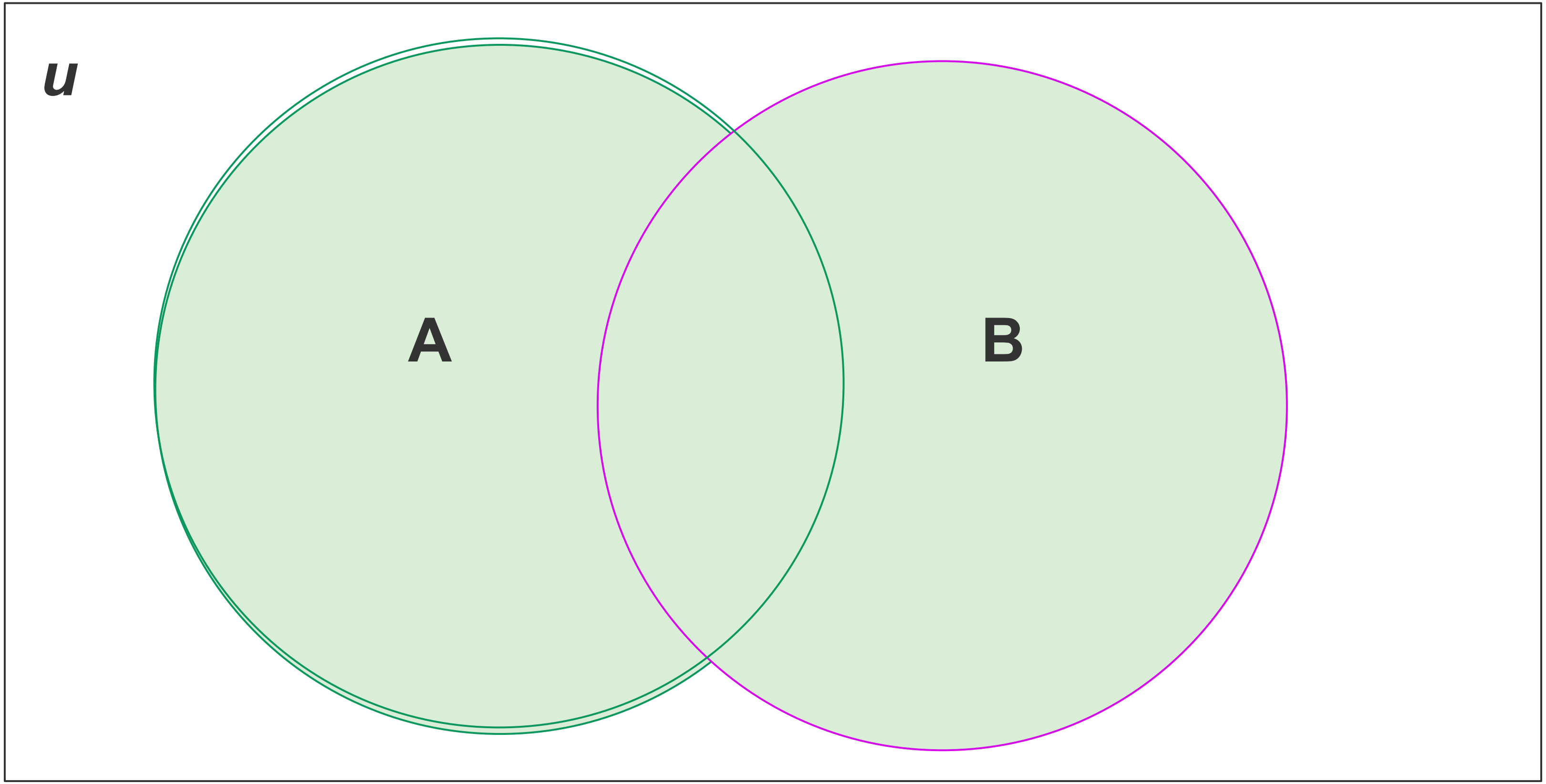


Understand Sets Visually!



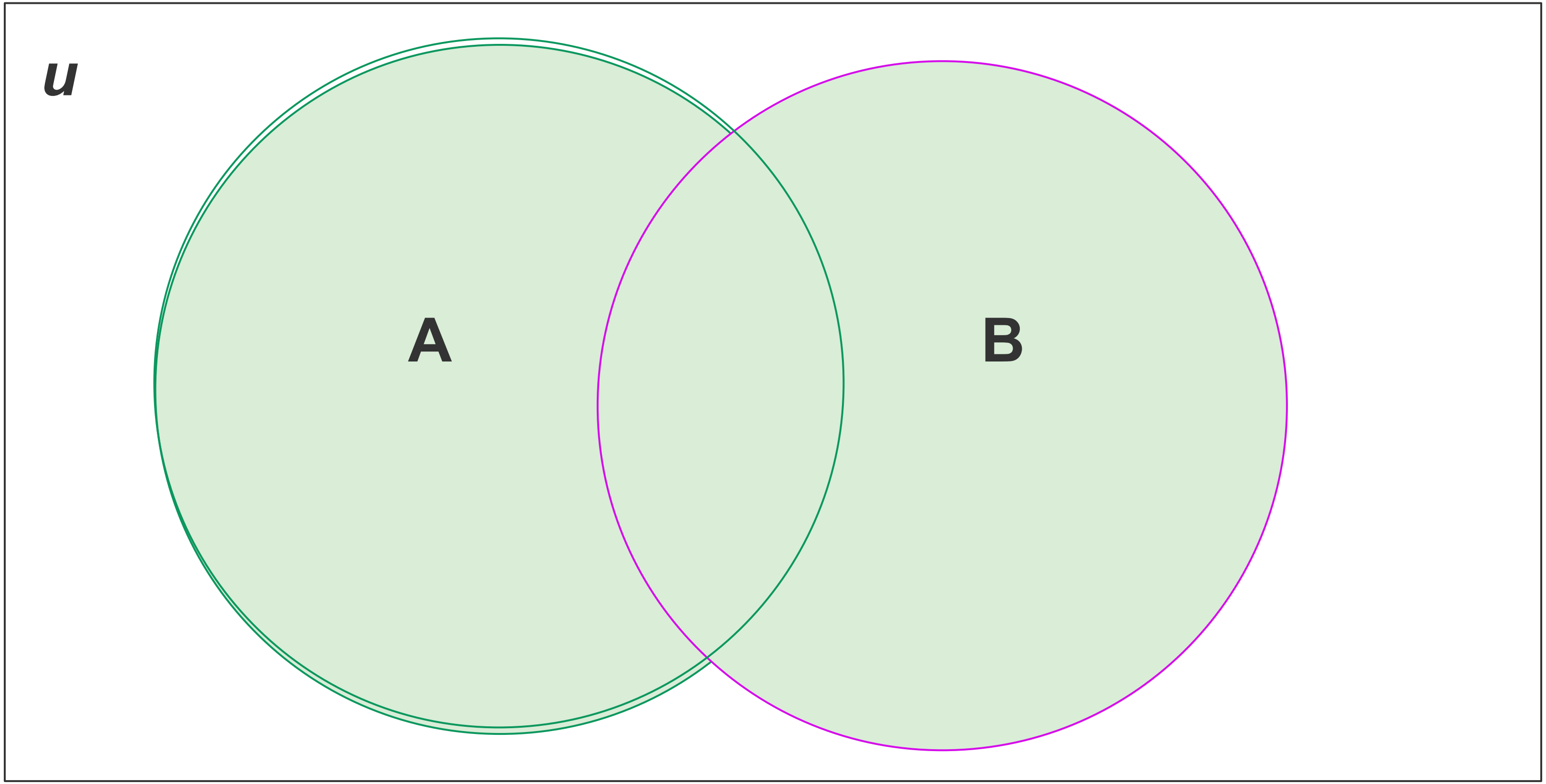
Understand Sets Visually!

What Set Operation is this?



Understand Sets Visually!

What Set Operation is this?
Union: A U B



Problem 2 – How Many Elements?

For each of these, how many elements are in the set? If the set has infinitely many elements, say ∞.

a) $A = \{1, 2, 3, 2\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$

c) $C = A \times (B \cup \{7\})$

d) $D = \emptyset$

e) $E = \{\emptyset\}$

f) $F = \mathcal{P}(\{\emptyset\})$

Work this problem with the people around you, and then we'll go over it together!

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$

c) $C = A \times (B \cup \{7\})$

d) $D = \emptyset$

e) $E = \{\emptyset\}$

f) $F = \mathcal{P}(\{\emptyset\})$

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$ $3, A = \{1, 2, 3\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$

c) $C = A \times (B \cup \{7\})$

d) $D = \emptyset$

e) $E = \{\emptyset\}$

f) $F = \mathcal{P}(\{\emptyset\})$

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$ $3, A = \{1, 2, 3\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$ $2, B = \{\emptyset, \{\emptyset\}\}$

c) $C = A \times (B \cup \{7\})$

d) $D = \emptyset$

e) $E = \{\emptyset\}$

f) $F = \mathcal{P}(\{\emptyset\})$

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$ $3, A = \{1, 2, 3\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$ $2, B = \{\emptyset, \{\emptyset\}\}$

c) $C = A \times (B \cup \{7\})$ $9, C = \{1, 2, 3\} \times \{7, \emptyset, \{\emptyset\}\}$

d) $D = \emptyset$

e) $E = \{\emptyset\}$

f) $F = \mathcal{P}(\{\emptyset\})$

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$ $3, A = \{1, 2, 3\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$ $2, B = \{\emptyset, \{\emptyset\}\}$

c) $C = A \times (B \cup \{7\})$ $9, C = \{1, 2, 3\} \times \{7, \emptyset, \{\emptyset\}\}$

d) $D = \emptyset$ 0

e) $E = \{\emptyset\}$

f) $F = \mathcal{P}(\{\emptyset\})$

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$ $\quad 3, A = \{1, 2, 3\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$ $\quad 2, B = \{\emptyset, \{\emptyset\}\}$

c) $C = A \times (B \cup \{7\})$ $\quad 9, C = \{1, 2, 3\} \times \{7, \emptyset, \{\emptyset\}\}$

d) $D = \emptyset$ $\quad 0$

e) $E = \{\emptyset\}$ $\quad 1$

f) $F = \mathcal{P}(\{\emptyset\})$

Problem 2 – How Many Elements?

a) $A = \{1, 2, 3, 2\}$ $3, A = \{1, 2, 3\}$

b) $B = \{\{\}, \{\{\}\}, \{\{\}, \{\}\}, \{\{\}, \{\}, \{\}\}, \ldots\}$ $2, B = \{\emptyset, \{\emptyset\}\}$

c) $C = A \times (B \cup \{7\})$ $9, C = \{1, 2, 3\} \times \{7, \emptyset, \{\emptyset\}\}$

d) $D = \emptyset$ 0

e) $E = \{\emptyset\}$ 1

f) $F = \mathcal{P}(\{\emptyset\})$ $2, F = \{\emptyset, \{\emptyset\}\}$

Note: the empty set is **not an element of every set** but a **subset** of every set

Set Proofs

Subset Proofs

One of the most common types of proofs you will be asked to write involving sets is a subset proof. That is, you will be asked to prove that $A \subseteq B$. We always approach these proofs with the same proof skeleton:

Let x be an arbitrary element of A, so $x \in A$.
… some steps using set definitions to show that x must also be in B…
Thus, $x \in B$
Since x was arbitrary, $A \subseteq B$.

Set Equality Proofs

Another common type of set proof is proving that $A = B$. The trick here is that this is secretly just two subset proofs! We need to show both that $A \subseteq B$ and $B \subseteq A$. Again, we will always use the same proof skeleton:

Let x be an arbitrary element of A, so $x \in A$.
$\ldots$ Thus, $x \in B$
Since x was arbitrary, $A \subseteq B$.
Let y be an arbitrary element of B, so $y \in B$.
$\ldots$ Thus, $y \in A$
Since y was arbitrary, $\mathrm{B} \subseteq A$.
As we have shown both that $A \subseteq B$ and $B \subseteq A$, therefore $A = B$.

Problem 4 – Set Equality

a) Prove that $A \cap (A \cup B) = A$ for any sets A, B.

Problem 4 – Set Equality

a) Prove that $A \cap (A \cup B) = A$ for any sets A, B.

Let x be an arbitrary element of $A \cap (A \cup B)$.
$\ldots$
Since x was arbitrary, $A \cap (A \cup B) \subseteq A$.

Now let y be an arbitrary member of A. Then $y \in A$. So certainly $y \in A$ or $y \in B$.
$\ldots$
Since y was arbitrary, $A \subseteq A \cap (A \cup B)$.

Therefore $A \cap (A \cup B) = A$, by containment in both directions.

Problem 4 – Set Equality

a) Prove that $A \cap (A \cup B) = A$ for any sets A, B.

Let x be an arbitrary element of $A \cap (A \cup B)$.
Then by definition of intersection, $x \in A$ and $x \in A \cup B$. So certainly, $x \in A$.
Since x was arbitrary, $A \cap (A \cup B) \subseteq A$.

Now let y be an arbitrary member of A. Then $y \in A$. So certainly $y \in A$ or $y \in B$.
$\ldots$
Since y was arbitrary, $A \subseteq A \cap (A \cup B)$.

Therefore $A \cap (A \cup B) = A$, by containment in both directions.

Problem 4 – Set Equality

a) Prove that $A \cap (A \cup B) = A$ for any sets A, B.

Let x be an arbitrary element of $A \cap (A \cup B)$.
Then by definition of intersection, $x \in A$ and $x \in A \cup B$. So certainly, $x \in A$.
Since x was arbitrary, $A \cap (A \cup B) \subseteq A$.

Now let y be an arbitrary member of A. Then $y \in A$. So certainly $y \in A$ or $y \in B$.
Then by definition of union, $y \in A \cup B$.
$\ldots$
Since y was arbitrary, $A \subseteq A \cap (A \cup B)$.

Therefore $A \cap (A \cup B) = A$, by containment in both directions.

Problem 4 – Set Equality

a) Prove that $A \cap (A \cup B) = A$ for any sets A, B.

Let x be an arbitrary element of $A \cap (A \cup B)$.
Then by definition of intersection, $x \in A$ and $x \in A \cup B$. So certainly, $x \in A$.
Since x was arbitrary, $A \cap (A \cup B) \subseteq A$.

Now let y be an arbitrary member of A. Then $y \in A$. So certainly $y \in A$ or $y \in B$.
Then by definition of union, $y \in A \cup B$.
Since $y \in A$ and $y \in A \cup B$, then by definition of intersection, $y \in A \cap (A \cup B)$.
Since y was arbitrary, $A \subseteq A \cap (A \cup B)$.

Therefore $A \cap (A \cup B) = A$, by containment in both directions.

Structural Induction

Idea of Structural Induction

Every element is built up recursively…

So to show $P(s)$ for all $s \in S$…

Show $P(b)$ for all base case elements b.

Show for an arbitrary element not in the base case, if $P()$ holds for every named element in the recursive rule, then $P()$ holds for the new element (each recursive rule will be a case of this proof).

Structural Induction Template

Let $P(x)$ be “<predicate>”. We show $P(x)$ holds for all $x \in S$ by structural induction.

Base Case: Show $P(x)$
[Do that for every base cases x in S.]

Inductive Hypothesis: Suppose $P(x)$ for an arbitrary x
[Do that for every x listed as in S in the recursive rules.]

Inductive Step: Show $P()$ holds for y.
[You will need a separate case/step for every recursive rule.]

Therefore $P(x)$ holds for all $x \in S$ by the principle of induction.

Problem 5b – Structural Induction on Trees

Definition of Tree:
Basis Step: • is a Tree.
Recursive Step: If L is a Tree and R is a Tree then Tree(•, L, R) is a Tree

Definition of leaves():
leaves(•) = 1
leaves(Tree(•, L, R)) = leaves(L) + leaves(R)

Definition of size():
size(•) = 1
size(Tree(•, L, R)) =1 + size(L) + size(R)

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

Work on this problem with the people around you.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For x ∈ S, let P(x) be "".
We show P(x) holds for all x ∈ S by structural induction on x.

Base Case: Show P(x) (for all x in the basis rules)

Inductive Hypothesis: Suppose P(x) (for all x in the recursive rules), i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be "leaves(T) ≥ size(T)/2 + 1/2".
We show P(x) holds for all x ∈ S by structural induction on x.

<u>Base Case:</u> Show P(x) (for all x in the basis rules)

<u>Inductive Hypothesis:</u> Suppose P(x) (for all x in the recursive rules), i.e. (IH in terms of P(x))

<u>Inductive Step:</u> Goal: Show that P(?) holds. (IS goal in terms of P(?))

<u>Conclusion:</u> Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be "leaves(T) ≥ size(T)/2 + 1/2".
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> Show P(x) (for all x in the basis rules)

<u>Inductive Hypothesis:</u> Suppose P(x) (for all x in the recursive rules), i.e. (IH in terms of P(x))

<u>Inductive Step:</u> Goal: Show that P(?) holds. (IS goal in terms of P(?))

<u>Conclusion:</u> Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be "leaves(T) ≥ size(T)/2 + 1/2".
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(x) (for all x in the recursive rules),
i.e. (IH in terms of P(x))

<u>Inductive Step:</u> Goal: Show that P(?) holds. (IS goal in terms of P(?))

<u>Conclusion:</u> Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be "leaves(T) ≥ size(T)/2 + 1/2".
We show P(T) holds for all trees T by structural induction on T.

Base Case: P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

Inductive Hypothesis: Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

Base Case: P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

Inductive Hypothesis: Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

Conclusion: Therefore P(x) holds for all x ∈ S by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

Base Case: P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

Inductive Hypothesis: Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

Inductive Step: Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that $\text{leaves}(T) \geq \text{size}(T)/2 + 1/2$ for all Trees T

For a tree T, let $P(T)$ be "$\text{leaves}(T) \geq \text{size}(T)/2 + 1/2$".
We show $P(T)$ holds for all trees T by structural induction on T.

<u>Base Case:</u> $P(\bullet)$: By definition of $\text{leaves}(\bullet)$, $\text{leaves}(\bullet) = 1$ and $\text{size}(\bullet) = 1$.
So, $\text{leaves}(\bullet) = 1 \geq 1/2 + 1/2 = \text{size}(\bullet)/2 + 1/2$, so $P(\bullet)$ holds.

<u>Inductive Hypothesis:</u> Suppose $P(L)$ and $P(R)$ hold for some arbitrary trees L and R,
i.e., $\text{leaves}(L) \geq \text{size}(L)/2 + 1/2$, $\text{leaves}(R) \geq \text{size}(R)/2 + 1/2$

<u>Inductive Step:</u> Goal: Show $P(\text{Tree}(\bullet, L, R))$: $\text{leaves}(P(\text{Tree}(\bullet, L, R)) \geq \text{size}(P(\text{Tree}(\bullet, L, R))/2 + 1/2$

Again, as long as you can get this far, you will get the majority of points on the problem! Go for this skeleton first, and then think about what you need to do to complete the proof.

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be "leaves(T) ≥ size(T)/2 + 1/2".
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

leaves(Tree(•, L, R)) =

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

leaves(Tree(•, L, R)) = leaves(L) + leaves(R)   definition of leaves

Conclusion: Therefore P(T) holds for all trees T by the principle of induction.

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

leaves(Tree(•, L, R))	= leaves(L) + leaves(R)	definition of leaves
	≥ (size(L)/2 + 1/2) + (size(R)/2 + 1/2)	by Inductive Hypothesis

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that $\text{leaves}(T) \geq \text{size}(T)/2 + 1/2$ for all Trees T

For a tree T, let P(T) be "$\text{leaves}(T) \geq \text{size}(T)/2 + 1/2$".
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, $\text{leaves}(\bullet) = 1 \geq 1/2 + 1/2 = \text{size}(\bullet)/2 + 1/2$, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., $\text{leaves}(L) \geq \text{size}(L)/2 + 1/2$, $\text{leaves}(R) \geq \text{size}(R)/2 + 1/2$

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): $\text{leaves}(P(\text{Tree}(\bullet, L, R)) \geq \text{size}(P(\text{Tree}(\bullet, L, R))/2 + 1/2$

$\text{leaves}(\text{Tree}(\bullet, L, R))$	$= \text{leaves}(L) + \text{leaves}(R)$	definition of leaves
	$\geq (\text{size}(L)/2 + 1/2) + (\text{size}(R)/2 + 1/2)$	by Inductive Hypothesis
	$= (1/2 + \text{size}(L)/2 + \text{size}(R)/2) + 1/2$	

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

Base Case: P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

Inductive Hypothesis: Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

Inductive Step: Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

$$
\begin{aligned}
\text{leaves}(\text{Tree}(\bullet, L, R)) &= \text{leaves}(L) + \text{leaves}(R) && \text{definition of leaves} \\
&\geq (\text{size}(L)/2 + 1/2) + (\text{size}(R)/2 + 1/2) && \text{by Inductive Hypothesis} \\
&= (1/2 + \text{size}(L)/2 + \text{size}(R)/2) + 1/2 \\
&= (1 + \text{size}(L) + \text{size}(R)) / 2 + 1/2
\end{aligned}
$$

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be “leaves(T) ≥ size(T)/2 + 1/2”.
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

$$
\begin{aligned}
\text{leaves}(\text{Tree}(\bullet, L, R)) &= \text{leaves}(L) + \text{leaves}(R) && \text{definition of leaves} \\
&\geq (\text{size}(L)/2 + 1/2) + (\text{size}(R)/2 + 1/2) && \text{by Inductive Hypothesis} \\
&= (1/2 + \text{size}(L)/2 + \text{size}(R)/2) + 1/2 && \\
&= (1 + \text{size}(L) + \text{size}(R)) / 2 + 1/2 && \\
&= \text{size}(T)/2 + 1/2 && \text{definition of size}
\end{aligned}
$$

Conclusion: Therefore P(T) holds for all trees T by the principle of induction

Problem 5b - Structural Induction on Trees

Prove that leaves(T) ≥ size(T)/2 + 1/2 for all Trees T

For a tree T, let P(T) be "leaves(T) ≥ size(T)/2 + 1/2".
We show P(T) holds for all trees T by structural induction on T.

<u>Base Case:</u> P(•): By definition of leaves(•), leaves(•) = 1 and size(•) = 1.
So, leaves(•) = 1 ≥ 1/2 + 1/2 = size(•)/2 + 1/2, so P(•) holds.

<u>Inductive Hypothesis:</u> Suppose P(L) and P(R) hold for some arbitrary trees L and R,
i.e., leaves(L) ≥ size(L)/2 + 1/2, leaves(R) ≥ size(R)/2 + 1/2

<u>Inductive Step:</u> Goal: Show P(Tree(•, L, R)): leaves(P(Tree(•, L, R)) ≥ size(P(Tree(•, L, R))/2 + 1/2

leaves(Tree(•, L, R))	= leaves(L) + leaves(R)	definition of leaves
	≥ (size(L)/2 + 1/2) + (size(R)/2 + 1/2)	by Inductive Hypothesis
	= (1/2 + size(L)/2 + size(R)/2) + 1/2	
	= (1 + size(L) + size(R)) / 2 + 1/2	
	= size(T)/2 + 1/2	definition of size

So, P(Tree(•, L, R)) holds!
<u>Conclusion:</u> Therefore P(T) holds for all trees T by the principle of induction.

Problem 5a - Structural Induction on Strings

Definition of string:
Basis Step: "" is a string.
Recursive Step: If X is a string and c is a character then append(c, X) is a string.

Definition of **len**():
len("") = 0
len(append(c, X)) = 1 + len(X)

Definition of **double**():
double("") = ""
double(append(c, X)) = append(c, append(c, double(X)))

Prove that for any string X, len(double(X)) = 2len(X).

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For x ∈ S, let P(x) be "".
We show P(x) holds for all x ∈ S by structural induction on x.

Base Case: Show P(x) (for all x in the basis rules)

Inductive Hypothesis: Suppose P(x) (for all x in the recursive rules), i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be "len(double(X)) = 2len(X)".
We show P(x) holds for all x ∈ S by structural induction on x.

Base Case: Show P(x) (for all x in the basis rules)

Inductive Hypothesis: Suppose P(x) (for all x in the recursive rules), i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be "len(double(X)) = 2len(X)".
We prove P(X) for all strings X by structural induction on X

Base Case: Show P(x) (for all x in the basis rules)

Inductive Hypothesis: Suppose P(x) (for all x in the recursive rules), i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be "len(double(X)) = 2len(X)".
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(x) (for all x in the recursive rules),
i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. (IH in terms of P(x))

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show that P(?) holds. (IS goal in terms of P(?))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

Conclusion: Therefore P(x) holds for all x ∈ S by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

Conclusion: Therefore P(X) holds for all strings X by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be "len(double(X)) = 2len(X)".
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X))) =

Conclusion: Therefore P(X) holds for all strings X by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X))) = len(append(c, append(c, double(X)))) definition of double

Conclusion: Therefore P(X) holds for all strings X by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

<u>Base Case:</u> P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

<u>Inductive Hypothesis:</u> Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

<u>Inductive Step:</u> Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X)))	= len(append(c, append(c, double(X))))	definition of double
	= 1 + len(append(c, double(X)))	definition of len

Conclusion: Therefore P(X) holds for all strings X by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be "len(double(X)) = 2len(X)".
We prove P(X) for all strings X by structural induction on X

<u>Base Case:</u> P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

<u>Inductive Hypothesis:</u> Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

<u>Inductive Step:</u> Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X)))	= len(append(c, append(c, double(X))))	definition of double
	= 1 + len(append(c, double(X)))	definition of len
	= 1 + 1 + len(double(X))	definition of len

Conclusion: Therefore P(X) holds for all strings X by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X)))	= len(append(c, append(c, double(X))))	definition of double
	= 1 + len(append(c, double(X)))	definition of len
	= 1 + 1 + len(double(X))	definition of len
	= 2 + 2len(X)	by I.H.

Conclusion: Therefore P(X) holds for all strings X by structural induction

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be "len(double(X)) = 2len(X)".
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X))) = len(append(c, append(c, double(X))))	definition of double
= 1 + len(append(c, double(X)))	definition of len
= 1 + 1 + len(double(X))	definition of len
= 2 + 2len(X)	by I.H.
= 2(1 + len(X))	

Conclusion: Therefore P(X) holds for all strings X by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X)))	= len(append(c, append(c, double(X))))	definition of double
	= 1 + len(append(c, double(X)))	definition of len
	= 1 + 1 + len(double(X))	definition of len
	= 2 + 2len(X)	by I.H.
	= 2(1 + len(X))	
	= 2(len(append(c, X)))	definition of len

Conclusion: Therefore P(X) holds for all strings X by structural induction.

Problem 5a - Structural Induction on Strings

Prove that for any string X,
len(double(X)) = 2len(X)

For a string X, let P(X) be “len(double(X)) = 2len(X)”.
We prove P(X) for all strings X by structural induction on X

Base Case: P(""): By definition, len(double("")) = len("") = 0 = 2 · 0 = 2len(""), so P("") holds

Inductive Hypothesis: Suppose P(X) holds for some arbitrary string X,
i.e. len(double(X)) = 2len(X)

Inductive Step: Goal: Show P(append(c, X)) for any c: len(double(append(c, X))) = 2(len(append(c, X)))

len(double(append(c, X)))	= len(append(c, append(c, double(X))))	definition of double
	= 1 + len(append(c, double(X)))	definition of len
	= 1 + 1 + len(double(X))	definition of len
	= 2 + 2len(X)	by I.H.
	= 2(1 + len(X))	
	= 2(len(append(c, X)))	definition of len

So, P(append(c, X)) holds!

Conclusion: Therefore P(X) holds for all strings X by structural induction.

That’s All, Folks!

Thanks for coming to section this week!
Any questions?