Discrete Maths Notes

October 19, 2023

Contents

1 Logic

1.1 Propositional Logic

1.1.1 Basics

A __proposition__ is a statement that is either true or false.
Prepositions will be represented mathematically with capital letters A, B, C...
These prepositions are then are connected into more complex compound prepositions using *connectives*. Connectives are statements like "and, implies, if-then" and are represented mathematically with the symbols below.

❶ It's not always easy to determine if they're true/false.

Connectives			
Symbol	Name	English Term(s)	Reading
$\wedge$	AND	And, But, Also	A and B
$\vee$	OR	-	A or B
$\implies$	IMPLICATION	If A, Then B If A, then B A implies B A, therefore B A only if B B follows from A A is a sufficient condition for B B is a necessary condition for A	A implies B
$\iff$	BICONDITIONAL	If & only if A is necessary and sufficient for B	A if and only if B
$\neg$	NEGATION	Not...	Not A

❶ *A Biconditional can also be thought of* $(A \implies B) \wedge (B \implies A)$
Negation may sometimes be represented as A⫫ or $\overline{A}$

1.1.2 Terminology

$A \wedge B$ - <u>conjunction</u> of <u>conjuncts</u> A and B
$A \vee B$ - <u>disjunction</u> of <u>disjuncts</u> A and B
$A \implies B$ - A is the <u>hypothesis</u>/antecedent and B is the <u>conclusion</u>/consequence

1.1.3 Examples

1 Compound Proposition

If <u>all humans are mortal</u>_{prp A} and <u>all Greeks are human</u>_{prp B}
then <u>all Greeks are mortal</u>_{prp C} can be represented as $A \wedge B \implies C$

2 Negation

Chocolate is sweet $\rightarrow$ Chocolate is <u>not</u> sweet

Peter is tall and thin $\rightarrow$ <u>Peter is short or fat</u>

The river is shallow or polluted $\rightarrow$ <u>The river is deep and polluted.</u>

❶ Short <u>and</u> fat would be incorrect!

❶ Not shallow <u>or</u> not polluted would be incorrect!

3 Implication: `hypothesis` and `conclusion`

If `the rain continues` then `the river will flood`

A sufficient condition for a `network failure` is that the `central switch goes down`

`The avocados are ripe` only if `they are dark and soft`

`A good diet` is a necessary condition for `a healthy cat`

1.1.4 Satifiability, Tautology, Contradiction

A proposition is <u>satisfiable</u> if it is true for *at least one* combination of boolean values.

❶ You don't need a whole truth table for this, just look for one!

A <u>Boolean Satisfiability Problem (SAT)</u> is checking for satisfiability in a propositional logic formula.

A <u>Tautology</u> is a proposition that is <u>always true</u>
  _{ex} $A \vee \neg A$

A <u>Contradiction</u> is a proposition that is always false.
  _{ex} $A \wedge \neg A$

4

1.2 Truth Tables

1.2.1 Basics

Truth Tables are used for determining all the possible outputs of a complex compound propostion.

<u>The Columns</u> Are for the prepositions, <u>intermediate compound prepositions</u> and the whole compound preposition.

> ❶ The intrmt' prepositions are optional steps to make solving easier, use as needed.

<u>The Rows</u> Are to contain the different sets of possible truth values for each proposition. You will have 2^p rows where p is the number of propositions (then $+1$ for the header).

⚠ The connectives in a compound propositional logic problem follow an order of precedence (the PEMDAS of logic) in the following order;
$\neg$, $\wedge$, $\vee$, $\implies$, $\iff$

1.2.2 Connective Outputs

Negation

A	$\neg A$
T	F
F	T

And

A	B	$A \wedge B$
T	T	T
T	F	F
F	T	F
F	F	F

Or

A	B	$A \vee B$
T	T	T
T	F	T
F	T	T
F	F	F

Implication

A	B	$A \implies B$
T	T	T
T	F	F
F	T	T
F	F	T

Biconditional

A	B	$A \iff B$
T	T	T
T	F	F
F	T	F
F	F	T

An implication is true when the hypothesis is false or the conclusion is true.

A Biconditional is true when the two prepositions have the same value.

Out of all these outputs, the most unintuitive is the 3rd implication output ($F, T \implies T$). The easiest way to understand this output is with the proposition "If it is raining, then the ground is wet"; now say you step outside and it is not raining, but the ground is wet. The entire statement isn't false or incorrect, but the first part of it still has a false value.
The only way to make an implication false is when the hypothesis is true but the conclusion is false.

1.2.3 Examples

$A \implies B \iff B \implies A$				
A	B	$A \implies B$	$B \implies A$	——
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

$A \wedge \neg B \implies \neg C$				
A	B	C	$A \wedge \neg B$	——
T	T	T	F	T
T	T	F	F	T
T	F	T	T	F
T	F	F	T	T
F	T	T	F	T
F	T	F	F	T
F	F	T	F	T
F	F	F	F	T

❶ *Remember, columns like $A \implies B$ are optional in-between steps to help solve each problem.*

1.2.4 Exercise: Finding Tautologies, Satisfiable & Contradicting Props'

Indicate whether each of the following is a tautology, satisfiable but not a tautology or a contradiction;

$A \implies B$

$A \implies A$

$A \implies \neg B \vee \neg C$

$A \vee B \implies B$

$(A \wedge B) \implies (A \vee B)$

$A \vee \neg A \implies B \wedge \neg B$

(Answers and explanations on the next page...)

ⓘ Notice how none of these rely on drawing out a whole truth table! Focus on trying to find a way to get each proposition to output true and a way to get it to output false!

$A \implies B$
Satisfiable but not a tautology
Just knowing the properties of an implication you should know there's way to get true outputs and a false output.

$A \implies A$
Tautology
Only would be $T \implies T$ or $F \implies F$, both of which result in true.

$A \implies \neg B \vee \neg C$
Satisfiable but not a tautology
Instead of making a long unpleasant truth table, it's easiest to start by simply looking for one true and one false possible output.
We can make the left side true simply by making A false, since all that remains is an or statement we now have a true output.
We can just as easily find a false output for this proposition with $A = T$, $B = T(\neg B = F)$ to make the implication false, then we can just make $\neg C$ false to make the or output false.

$A \vee B \implies B$
Satisfiable but not a tautology
If we make B true then the biconditional will always be true regardless of A.
There is only one way to make an implication false, so if we can set up A and B to result in that false output, it won't be a tautology. If we make A true and B false it will make the implication false!

$(A \wedge B) \implies (A \vee B)$
Tautology
Remember the only way to make an implication false is if the hypothesis is true and the conclusion is false. There is absolutely no way to do this because of the and/or setup!

$A \vee \neg A \implies B \wedge \neg B$
Contradiction
The left side is always true and the right side is always false. So the result of the implication is always false!

1.3 Equivalence

1.3.1 Introduction to Equivalence

📕 Two (compound) propositions P and Q are **logically equivalent** when their truth values always match (Meaning they'll have the same truth table!).
Equivalence is denoted by $P \equiv Q$.

Equivalence relates heavily to the concept of Tautologies;

  P and Q are equivalent when $P \iff Q$ is a tautology.

  A proposition P is a tautology iff (if and only if) it is equivalent to T (true), i.e $P \equiv T$

Examples

1

Given the implication $A \implies B$, are the following equivalent?

  The contrapositive: $\neg B \implies \neg A$

  The converse: $B \implies A$

A	B	$A \implies B$	$\neg B \implies \neg A$	$B \implies A$
T	T	T	T	T
T	F	F	F	T
F	T	T	T	F
F	F	T	T	T

Looking at the table we can see that $A \implies B$ and $\neg B \implies \neg A$ are equivalent.

Now, what about $\neg A \lor B$?

A	B	$A \implies B$	$\neg A \lor B$
T	T	T	T
T	F	F	F
F	T	T	T
F	F	T	T

Yep! $\neg A \lor B \equiv A \implies B$.

ⓘ This is actually one of the equivalence laws you'll see in the next section!

Understanding equivalent boolean expressions is very important in computer science (for code) and chip design (for logic gates). Consider the code below;

```
if(x > 0 || (x <= 0 && y > 100))
```

Lets see if we can change this expression to something equivalent but simplified.
Let A be `x > 0` and let B be `y > 100`
Now we can compare the truth values of $A \vee (\neg A \wedge B)$ and $A \vee B$.

A	B	$A \vee (\neg A \wedge B)$	$A \vee B$
T	T	T	T
T	F	T	T
F	T	T	T
F	F	F	F

They're equivalent! We can reduce the if statement's expression to simply;

```
if(x > 0 || y > 100)
```

1.3.2 Equivalence Laws

For more complex propositions it is impractical to create a set of massive truth tables to check for equivalence. So instead we utilize equivalence laws to directly prove equivalence.

Nine Equivalence Laws;

Many of these are pretty self-explanatory

Double Negation Law: $\neg(\neg A) \equiv A$

Identity Laws: $A \wedge T \equiv A \qquad A \vee F \equiv A$

Domination Laws: $A \vee T \equiv T \qquad A \wedge F \equiv F$

Commutative Laws: $A \wedge B \equiv B \wedge A \qquad A \vee B \equiv B \vee A$

Associative Laws: $(A \wedge B) \wedge C \equiv A \wedge (B \wedge C) \qquad (A \vee B) \vee C \equiv A \vee (B \vee C)$

Idempotent Laws: $A \wedge A \equiv A \qquad A \vee A \equiv A$

Distributive Laws: $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C) \qquad A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$

❶ Very similar to the algebriac distributive law

DeMorgan's Laws: $\neg(A \wedge B) \equiv \neg A \vee \neg B \qquad \neg(A \vee B) \equiv \neg A \wedge \neg B$

Implication Laws: $A \implies B \equiv \neg B \implies \neg A \equiv \neg A \vee B$

Examples

₁ Prove $A \vee (\neg A \wedge B) \equiv A \vee B$

$$
\begin{aligned}
A \vee (\neg A \wedge B) &\equiv (A \vee \neg A) \wedge (A \vee B) \quad \text{(Distributive)} \\
&\equiv T \wedge (A \vee B) \\
&\equiv A \vee B \quad\quad\quad\quad\quad\quad \text{(Identity)}
\end{aligned}
$$

❖ *In solving these, the goal should be to reduce the # of letters in the propositions. Focus on the side of an equivalence with more going on and try to reduce it down since the more complex proposition will have more oppurtunities to utilize the different equivalence laws.*

₂ Simplify $A \wedge \neg(A \wedge B)$

$$
\begin{aligned}
A \wedge \neg(A \wedge B) &\equiv A \wedge (\neg A \vee \neg B) \quad\quad\; \text{(DeMorgan's)} \\
&\equiv (A \wedge \neg A) \vee (A \wedge \neg B) \quad \text{(Distributive)} \\
&\equiv F \vee (A \wedge \neg B) \\
&\equiv A \wedge \neg B \quad\quad\quad\quad\quad\quad \text{(Identity)}
\end{aligned}
$$

❖ *You don't have to name the laws you're using in the homework, the simple $\equiv$ down the middle format for each step is fine.*

₃ Show that $(A \wedge B) \implies (A \vee B)$ is a tautology.

$$
\begin{aligned}
(A \wedge B) \implies (A \vee B) &\equiv \neg(A \wedge B) \vee (A \vee B) \quad\;\; \text{(Implication)} \\
&\equiv (\neg A \vee \neg B) \vee (A \vee B) \quad \text{(DeMorgan's)} \\
&\equiv \neg A \vee \neg B \vee A \vee B \quad\quad\;\, \text{(Associative)} \\
&\equiv \neg A \vee A \vee \neg B \vee B \quad\quad\;\, \text{(Commutative)} \\
&\equiv (\neg A \vee A) \vee (\neg B \vee B) \quad \text{(Associative)} \\
&\equiv T \vee T \\
&\equiv T \quad\quad\quad\quad\quad\quad\quad\quad\;\; \text{(Idempotent)}
\end{aligned}
$$

1.4 Arguments

📓 An **argument** is a sequence of propositions in which the conjunction of the initial propositions implies the final proposition
An argument can be represented as;
$P_1 \wedge P_2 \wedge P_3 ... \wedge P_n \implies Q$

Examples

If George Washington was the first president of the United States, then John Adams was the first vice president. George Washington was the first president of the United States. Therefore John Adams was the first vice president.

> ❯ Let A be "George Washington was the first president of the United States."

> ❯ Let B be "John Adams was the first vice president."

> ❯ $(A \implies B) \wedge A \implies B$

If Martina is the author of the book, then the book is fiction. But the book is nonfiction. Therefore Martina is not the author.

> ❯ Let A be "Martina is the author of the book."

> ❯ Let B be "The book is fiction."

> ❯ $(A \implies B) \wedge \neg B \implies \neg A$

The dog has a shiny coat and loves to bark. Consequently, the dog loves to bark.

> ❯ Let A be "The dog has a shiny coat."

> ❯ Let B be "The dog loves to bark."

> ❯ $A \wedge B \implies B$

1.4.1 Valid Arguments / Inference Rules

📖 An argument is **valid** if and only if **its conclusion is never false while its premises are true.**

We can't use a truth table to validate an argument since it only shows the truth values for the statement as a whole, instead we need to use new **Inference Rules**

Inference Rules

$$\begin{array}{c} P \\ P \implies Q \\ \hline \therefore Q \end{array} \qquad \begin{array}{c} P \implies Q \\ \neg Q \\ \hline \therefore \neg P \end{array}$$

Ex: If George Washington...   Ex: If Martina...

$$\begin{array}{c} P \wedge Q \\ \hline \therefore P \end{array} \qquad \begin{array}{c} P \\ \hline \therefore P \vee Q \end{array}$$

$$\begin{array}{c} P \\ Q \\ \hline \therefore P \wedge Q \end{array}$$

Ex: Paul is a good swimmer. Paul is a good runner.

Therefore Paul is a good swimmer and a good runner

ℹ️ *The items above the line can be combined/transformed into a new proof step defined below the line*

Examples (finding conclusions)

1. If the car was involved in the hit-and-run, then the paint would be chipped. But the paint is not chipped.

   > "Car was involved in a hit-and-run" $\rightarrow P$

   > "Paint would be chipped" $\rightarrow Q$

   > "The paint is not chipped" $\rightarrow \neg Q$

   > Conclusion: The car was not involved in a hit-and-run. From the second rule!

2. If the bill was sent today, then you will be paid tomorrow. You will be paid tomorrow.

   > Nothing can be concluded from this. ☺

3. If the program is efficient$_P$, it executes quickly$_Q$. Either the program is efficient$_P$, or it has a bug$_R$. However, the program does not execute quickly$_{\neg Q}$.

   > "If the program is efficient" $\rightarrow P$

   > "it executes quickly" $\rightarrow Q$

   > "it has a bug" $\rightarrow R$

   > "the program does not execute quickly" $\rightarrow \neg Q$

   > We start by knowing $P \implies Q$ and $P \vee R$ and $\neg Q$...

   > $(P \implies Q)$ and $\neg Q$ can imply $\neg P$

   > We need to transform $P \vee R$ to use it: $P \vee R \equiv \neg(\neg P) \vee R \equiv \neg P \implies R$

   > $\neg P \implies R$ and $\neg P$ (the first implication we isolated) now implies R by the first inference rule.

1.4.2 Proving a Valid Argument

Assuming the premises are true, apply a sequence of premises and derivation rules, which include the equivalence laws and inference.

General Steps

1. Identify all the premises (might need some transformations).

2. Think backwards. Start from what you want and then seek supporting premises, current results, and necessary equivalence laws and inference rules, until you reach the given premises.

   > ❶ Start with the RHS of the argument on the bottom of the list and work your way up

3. Write the proof sequence, where each step is either one premise or derived from previous step(s) using equivalence laws or inference rules.

Examples

₁ Prove $(A \implies B) \land (\neg C \lor A) \land C \implies B$

> ❶ *This one is already in its standard form - so we just need to identify each part of the standard $P_1 \land P_2 \land P_3... \land P_n \implies Q$ form. At the end of this we want to prove that B is true.*

$(A \implies B) \quad \land \quad (\neg C \lor A) \qquad \land \quad C \quad \implies B$

$\qquad\qquad \downarrow \quad \land \quad \overset{\text{Implication Law}}{C \implies A} \quad \land \quad C \quad \downarrow$

$(A \implies B) \quad \land \quad \overset{\text{first inference rule law}}{A} \qquad\qquad\qquad \downarrow$

$\qquad\qquad\quad \overset{\text{also by first inf rule}}{B} \qquad\qquad\qquad\qquad \implies B$

> ❶ *This is not usually how you would format these proofs, this table was to give you an idea of the actual process. The actual proof would look like the following;*
>
> *1. $A \implies B$*
>
> *2. $\neg C \lor A$*
>
> *3. C*
>
> *4. $C \implies A$ (2, Implication)*
>
> *5. A (3,4)*
>
> *6. B (1,5)*
>
> *You need to put every step in a seperate (numbered) line, starting with each component of the argument and then the transformations you do with the reason given. You dont need to name the law used but you need to mention the steps you combined to acheive the next part.*

Prove $A \wedge (B \implies C) \wedge ((A \wedge B) \implies (D \vee \neg C)) \wedge B \implies D$

    ⓘ *For this one focus on step 3 ($D \vee \neg C$) as your point to figure out this argument since its the only portion that has D in it.*

1. A

2. $B \implies C$

3. $(A \wedge B) \implies (D \vee \neg C)$

4. B

5. $A \wedge B$ (1,4)

6. $D \vee \neg C$ (3,5)

7. $C \implies D$ (6, Commutative, Implication) - Communicative used to swap C, D

8. C (2,4)

9. D (7,8)

Prove $(A \implies B) \wedge (\neg C \vee A) \wedge C \implies A \wedge B$

    ⓘ *For this one notice that the right-hand side isn't a single letter anymore. We now need to focus on proving the whole $A \wedge B$ statement. So this problem is actually solved a bit backwards, start by writing the last steps ($A, B, A \wedge B$) and then go up and figure out how you can prove A.*

1. $A \implies B$
2. $\neg C \vee A$
3. C
4. $C \implies A$ (2, Implication)
5. A (3,4)
6. B (1,5)
7. $A \wedge B$ (5,6)

    ⓘ *If instead this problem was looking for $A \vee B$, you could just prove either A or B to make the entire statement valid.*

1.4.3 The Deduction Method

Now, what if the conclusion is in implication form?
There are two ways of solving for this form, the main one being **The Deduction Method...**

Suppose the argument has the form:

$$P_1 \land P_2 \land P_3... \land P_n \implies (R \implies S)$$

where the conclusion itself is an implication. <u>We can add R as an additional premise and then imply S</u>. In other words, we can have the argument:

$$P_1 \land P_2 \land P_3... \land P_n \land R \implies S$$

Examples

₁ Prove $(A \implies B) \land (B \implies C) \implies (A \implies C)$

> ⓘ *Start with C at the bottom. Now the only way to validate C is if B is true, so make B step 4 and find the proper relations to make B true.*

Deduction: $(A \implies B) \land (B \implies C) \land A \implies C$
1. $A \implies B$
2. $B \implies C$
3. A
4. B \qquad\qquad (1,3)
5. C \qquad\qquad (2,4)

₂ Prove $\neg(A \land \neg B) \land (B \implies C) \implies (A \implies C)$
Deduction: $\neg(A \land \neg B) \land (B \implies C) \land A \implies C$
1. $\neg(A \land \neg B)$
2. $B \implies C$
3. A
4. $\neg A \lor B$ \qquad (1, DeMorgan's)
5. $A \implies B$ \qquad (4, Implication)
6. B \qquad\qquad (3,5)
7. C \qquad\qquad (2,6)

1.5 Predicate Logic

📕 A **predicate** represents the properties of/relations among objects.
Examples:
   n *is a perfect square*
   x *is greater than* y

Often propositional logic is not enough!

There are several cases where propositional logic won't help us reach needed conclusions or information;

Suppose we know that "All CS students must take CSCI 358". We cannot conclude that "Alice must take CSCI 358 where Alice is a CS student" using our current propositional logic knowledge.

Statements that hold many objects must be enumerated;

   ❯ Example:

   * If Alice is a CS student, then Alice must take CSCI358.

   * If Bob is a CS student, then Bob must take CSCI358.

   * If Chris is a CS student, then Chris must take CSCI358.

   * ...

   ❯ Solution: make statements with variables

   * If x is a CS student, then x must take CSCI358.

Statements that define the property of a group of objects;

   ❯ Example:

   * All new cars must be registered.

   * Some of the new CS students graduate with honor.

   ❯ Solution: Make statements with quantifiers:

   * Universal Quantifier - the property is satisfied by all members of the group.

   * Existential Quantifier - at least one member of the group satisfies the property.

1.5.1   Predicate representation

Predicates are represented like functions in other branches of maths;
e.g $P(x)$ represents some predicate such as "x is a perfect square".

Note that predicates can involve multiple variables, e.g Q(x,y) is "x is greater than y."

The two main quantifiers are represented with $\forall$ and $\exists$

- Universal Quantifier: $\forall$
  - › Read as "for all," "for every," "for each," or "for any."
  - › Ex: $\forall x, x > 0$ is read as "for any number x, x is greater than 0."
- Existential Quantifier: $\exists$
  - › Read as "there exists one," "there is," "for at least one," or "for some."
  - › Example: $\exists x, x > 0$ is read as "there exists a number x such that x is greater than zero."

  ❶ When $\forall x P(x)$ or $\exists x P(x)$ is used, the domain must be specified.

Truth Values of Predicates

Predicate	True When...	False When...	Examples
$\forall x P(x)$	If P(x) is true for **every** x in the domain	If there is **any** x in the domain such that P(x) is false	$P(x)$ is $x + 1 > x$, $\forall P(x)$ is true for the domain consisting of all real numbers.
			$Q(x)$ is $x < 2$. $\forall x Q(x)$ is false for the domain consisting of all real numbers because $Q(3)$ is false. $x = 3$ is a counterexample of $\forall x Q(x)$
$\exists x P(x)$	There's is an x **anywhere** such that P(x) is true.	P(x) is false for **every** x	$P(x)$ is $x > 3$. $\exists x P(x)$ is true for the domain consisting of all real numbers. Because when x=4, P(4) is true.
			$Q(x)$ is $X = x + 1$. $existsxQ(x)$ is false for the domain consisting of all real numbers. Because $Q(x)$ is false for every real number x

The quantifiers $\forall$ and $\exists$ have higher precedence than all logical connectives from propositional logic.

For Example:
  $\forall x P(x) \land Q(x)$ means $(\forall x P(x)) \land Q(x)$ rather than $\forall x (P(x) \land Q(x))$

The margin note at the top right reads:
❶ Once we plug in a value for x, the predicate becomes a proposition

Negating Quantified Expressions

$$\neg \forall x P(x) \equiv \exists x \neg P(x)$$

Example:

> ➢ Every CS Student Must take CSCI385.

> ➢ Negation: There is a CS student who doesn't have to take CSCI358

$$\neg \exists x P(x) \equiv \forall x \neg P(x)$$

Example:

> ➢ There is s student in this class who has taken CSCI 262.

> ➢ Negation: Every student in this class has **not** taken CS262

1.5.2 Translating to Logical Expressions

Converting statements to expressions

English to Logical Expressions

₁ Every parrot is beautiful

Translation:

> ❯ Assume that the domain consists of all parrots.
>
> > * Let $B(x)$ denote "x is beautiful"
> > * Then $\forall x \; B(x)$
>
> ❯ Assume that the domain consists of all animals.
>
> > * Let $P(x)$ denote "x is a parrot".
> > * Let $B(x)$ denote "x is beautiful"
> > * Then $\forall x \; (P(x) \implies B(x))$

> ℹ
> $\forall x (P(x) \wedge B(x))$
> would be incor-
> rect

₂ There exists a beautiful parrot

Translation:

> ❯ Assume that the domain consists of all parrots.
>
> > * Let $B(x)$ denote "x is beautiful
> > * Then $\exists x \; B(x)$
>
> ❯ Assume that the domain consists of all animals.
>
> > * Let $P(x)$ denote "x is a parrot"
> > * Let $B(x)$ denote "x is beautiful"
> > * Then $\exists x \; (P(x) \wedge B(x))$

> ❶ $\exists x \; (P(x) \implies B(x))$ *is an incorrect solution. If x is not a parrot then P(x) is false, since P(x) is attached to the start of the implication it would make the entire expression true (when it should be false)*

3 Let $P(x)$ denote "x speaks Russian"
and let $Q(x)$ denote "x knows the computer language C++"
Let the domain consist of all students at Mines.
Translate the following into logical expressions;

There is a student at Mines who speaks Russian and knows C++

> $\exists x \ (P(x) \land Q(x))$

There is a student at Mines who speaks Russian but doesn't know C++

> $\exists \ (P(x) \land \neg Q(x))$

Every student at Mines either speaks Russian or knows C++

> $\forall x \ (P(x) \lor Q(x))$

No student as Mines speaks Russian or knows C++

> $\forall x \ (\neg P(x) \land \neg Q(x))$

> or $\neg \exists x \ (P(x) \lor Q(x))$

Nested quantifiers

More than one quantifier may be needed to represent the meaning of a statemente in predicate logic.

1 Every real number has its corresponding negative

Assume that the domain consists of all real numbers

Let $P(x, y)$ denote "$x + y = 0$"

Then we can write $\forall x \ \exists y \ P(x, y)$

2 There is a person who loves everybody.

Assume that the domain consists of all people

Let $L(x, y)$ denote "x loves y".

Then we can write $\exists x \ \forall y \ L(x, y)$

Order of quantifiers

When quantifiers are of the **same** type, the order <u>doesn't</u> matter.

Example:

> ❯ Assume that the domain consists of all real numbers.

> ❯ Let $P(x, y)$ denote "$x + y = y + x$"

> ❯ $\forall x \; \forall y \; P(x, y)$ represents "For every real number x, for every real number y, x + y = y + x."

> ❯ $\forall y \; \forall x \; P(x, y)$ represents "For every real number y, for every real number x, x + y = y + x."

> ❯ $\forall x \; \forall y \; P(x, y)$ and $\forall y \; \forall x \; P(x, y)$ have the same meaning!

When quantifiers are of **different** types, the order <u>does</u> matter!

Example:

> ❯ Assume that the domain consists of all real numbers.

> ❯ Let $Q(x, y)$ denote "$x + y = 0$"

> ❯ $\forall x \; \exists y \; Q(x, y)$ represents "For every real number x, there is a real number y, such that x + y = 0."

> ❯ $\exists y \; forall x \; Q(x, y)$ represents "There is a real number y, such that for every real number x, x + y = 0."

> ❯ $\forall x \; \exists y \; Q(x, y)$ and $\exists y \; forall x \; Q(x, y)$ have different meanings!

Ex: Let $Q(x, y, z)$ be "$x + y = z$" and assume that the domain consists of all real numbers.

$$\forall x \; \forall y \; \exists z \; Q(x, y, z) \not\equiv \exists z \; \forall x \; \forall y \; Q(x, y, z)$$

1.5.3 Translation Examples

English to Logical Expression Examples

₁ Given the two unique statements;

1. John loves only Mary (If John loves any person, then that person is Mary.)

2. Only John loves Mary (If any person loves Mary, then that person is John.)

Let $J(x)$ be "x is John". let M(x) be "x is Mary". Let L(x,y) be "x loves y". The domain consists of all people.

1. John loves only Mary.

   ❯ For any person x, if x is John, then if it loves any person y, then y is Mary.

   ❯ $\forall x \ (J(x) \implies \forall y(L(x,y) \implies M(y)))$
   Or $\forall x \ \forall y \ (J(x) \wedge L(x,y) \implies M(y))$

2. Only John loves Mary.

   ❯ For any person x, if x is Mary, then if any person y loves x, then y is John.

   ❯ $\forall x \ (M(x) \implies \forall(L(y,x) \implies J(y)))$
   Or $\forall x \ \forall y \ (M(x) \wedge L(y,x) \implies J(y))$

₂ Given that; D(x) is "x is a dog". R(x) is "x is a rabbit". C(x,y) is "x chases y". The domain consists of all animals.
Translate the following;

1. All dogs chase rabbits.

   ❯ For any animal, if it is a dog, then for any other animal, if that animal is a rabbit, then the dog chases it.

   ❯ $\forall x \ (D(x) \implies \forall y(R(y) \implies C(x,y)))$

2. Some dogs chase all rabbits.

   ❯ There is some animal that is a dog and, for any other thing, if that animal is a rabbit, then the dog chases it.

   ❯ $\exists x \ \forall y(D(x) \wedge (R(y) \implies C(x,y)))$

3. Only dogs chase rabbits.

   ❯ For any animals, if it is a rabbit then, if any animal chases it, that animal is a dog.

   ❯ $\forall y \ (R(y) \implies \forall x(C(x,y) \implies D(x)))$

   ❯ or: For any two animals, if one is a rabbit and the other chases it, then the other is a dog.

   ❯ $\forall y \ \forall x \ (R(y) \wedge C(x,y) \implies D(x))$

Mathematical Statements to Logical Expression Examples

1 Translate "The sum of two positive integers is always positive."

Assume that the domain consists of all integers.

> For every two integers, if they are both positive, then the sum of them is positive.

> $\forall x \ \forall y \ (x > 0 \land y > 0 \implies (x + y > 0))$

Assume that the domain consists of all positive integers.

> For every two positive integers, the sum of them is positive.

> $\forall x \ \forall y \ (x + y > 0)$

2 Translate "The difference of two positive integers is not necessarily positive."

Assume that the domain consists of all integers.

> It is not the case that, for every two integers, if they are both positive, then the difference of them is positive.

> $\neg \forall x \ \forall y \ (x > 0 \land y > 0 \implies (x - y > 0))$
> Or: $\exists x \ \exists y ((x > 0 \land y > 0) \land (x - y \leq 0))$

Logical Expressions to English Examples

1 Translate: $\forall x (C(x) \lor \exists y (C(y) \land F(x, y)))$, where $C(x)$ denotes "x has a computer," $F(x, y)$ denotes "x and y are friends," and the domain consists of all Mines students.

For every student x at Mines, x has a computer or there is a student y such that y has a computer and x and y are friends.

Simplifies to: Every student at Mines has a computer or a friend that has a computer.

2 Translate: $\exists x \ \forall y \ \forall z ((F(x, y) \land F(x, z) \land (y \neq z)) \implies \neg F(y, z))$, where $F(x, y)$ denotes "x and y are friends," and the domain consists of all students at Mines.

$\exists x \ \forall y \ \forall z ((F(x, y) \land F(x, z) \land (y \neq z)) \implies \neg F(y, z))$ means that if students x and y are friends, and students x and z are friends, and y and z are not the same student, then y and z are not friends.

There is a student x such that for every student y and every student z other than y, if x and y are friends, x and z are friends, then y and z are not friends.

There is a Mines student none of whose friends are friends.

1.5.4   Negating Nested Quantifiers

General Principle: Moving a $\neg$ across a quantifier changes the kind of quantifier.

What is the negation of $\forall x \; \exists y (xy = 1)$?

> Let $P(x)$ denote $\exists y (xy = 1)$
>
> Then we know how to negate $\forall x \; P(x)$
>
> $\neg \forall x \; P(x) \equiv \exists x \; \neg P(x)$
>
> In addition, $\neg P(x) \equiv \neg \exists y \; (xy = 1) \equiv \forall y \; (xy \neq 1)$
>
> Therefore, the negation of $\forall x \; \exists y \; (xy = 1)$ is $\exists x \; \forall y (xy \neq 1)$

1.5.5 Arguments In Predicate Logic

Just like propositional logic, we need to utilize inference rules to prove these arguments.

Basic examples: (Assume the domain is all Mines CS students.)

1. All the CS students must take CSCI 358. Thus some CS students must take CSCI 358.

    > $\forall x \ P(x) \implies \exists x \ P(x)$ ✓

    > If every element has property P, then some element has property P

2. All the CS students must take CSCI 358. Thus Alice must take CSCI 358, where Alice is a CS student

    > $\forall x \ P(x) \implies P(a)$, where a is a constant ✓

    > If every element has property P, then a particular element has property P.

    ❶ The $P(a)$ here represents a "constant" Alice whom is a particular element within the domain

3. All the CS students must take CSCI 261 and CSCI 358. Thus all the CS students must take CSCI 261 and all the CS students must take CSCI 358.

    > $\forall x \ (P(x) \wedge Q(x)) \implies \forall x \ P(x) \wedge \forall x \ Q(x)$ ✓

    > If both P and Q are true for all the elements, then P is true for all elements and Q is true for all elements (duh)

4. Some CS students graduate with honors. Thus all the CS students graduate with honors.

    > $\exists x \ P(x) \implies \forall x \ P(x)$ ✗

    > If some element has property P, then all the elements have property P.

    ❶ This one also just doesn't make intuitive sense. Try rewriting arguments like this into English and see if they make sense!

All the equivalence laws and inference rules still hold!

- Double Negation Law: ¬(¬A) ≡ A
- Identity Laws: A ∧ T ≡ A   A ∨ F ≡ A
- Domination Laws: A ∨ T ≡ T   A ∧ F ≡ F
- Commutative Laws: A ∧ B ≡ B ∧ A   A ∨ B ≡ B ∨ A
- Associative Laws: (A ∧ B) ∧ C ≡ A ∧ (B ∧ C) (A ∨ B) ∨ C ≡ A ∨ (B ∨ C)
- Idempotent Laws: A ∧ A ≡ A   A ∨ A ≡ A
- Distributive Laws: A ∨ (B ∧ C) ≡ (A ∨ B) ∧ (A ∨ C)
  - A ∧ (B ∨ C) ≡ (A ∧ B) ∨ (A ∧ C)
- DeMorgan's Laws: ¬ (A ∧ B) ≡ ¬A ∨ ¬B
  - ¬ (A ∨ B) ≡ ¬A ∧ ¬B
- Implication Laws: A→B ≡ ¬B→ ¬A ≡ ¬A ∨ B

- P, P →Q can imply Q
- P →Q, ¬Q can imply ¬P
- P, Q can imply P ∧ Q
- P ∧ Q can imply P, Q
- P can imply P ∨ Q

However, these rules are not enough, predicates will utilize <u>four new inference rules</u>;

Universal Instantiation

$\forall x \; P(x)$ can imply $P(c)$, where c is a particular element or any arbitrary element in the domain.

> ❯ If for any x in which $P(x)$ is true, then $P(c)$ must also be true. More intuitively thought as the fact that if all elements in the domain have property "P", then any element (particular or arbitrary within the domain) must also have this property.

Existential Instantiation

$\exists x \; P(x)$ can imply $P(a)$, where a is a particular element not previously used in a proof sequence

> ❯ $\exists x P(x)$ means there must be some element in the domain that has property "P". Even though we don't know *exactly* what that element is, we can use the letter "a" to represent this particular element.

Universal Generalization

$P(c)$ can imply $\forall x \; P(x)$, where c is an arbitrary element in the domain.

> ❯ If any/every arbitrary element in the domain has property P ($P(c)$ always true), we can obviously say $\forall x P(x)$

Existential Generalization

$P(a)$ can imply $\exists x \; P(x)$, where a is a particular element.

> ❯ If a particular element in the domain has property "P", we can obviously say $\exists x \; P(x)$

❶ *The first two rules can be used to remove the quantifiers in front of the predicates. The last two rules can be used to add quantifiers to the front of predicates*

Let's look at all of these in more depth...

Universal Instantiation (UI)

$\forall x \; P(x)$ can imply $P(c)$, where c is a particular element or any arbitrary element in the domain.

Restrictions: If c is a variable, it cannot be already in $P(x)$

An **incorrect** use of UI would be saying $\forall x \; \exists y \; P(x, y)$ implies $\exists y \; P(y, y)$

For example, in the integer domain, if $P(x, y)$ means "$y > x$" then $\forall x \; \exists y \; P(x, y)$ is true, but $\exists y \; P(y, y)$ is false (y can't be greater than y ☺).

<u>Example</u>

Prove the following argument is valid: "All CS students must take CSCI358. Alice is a CS student. Therefore Alice must take CSCI 358." The domain consists of all Mines students.

Let $C(x)$ be "x is a CS student." s is a constant symbol. $D(x)$ is "x has to take CSCI358"

The argument would be: $\forall x (C(x) \implies D(x)) \land C(s) \implies D(s)$

1. $\forall x \; (C(x) \implies D(x))$

2. $C(s)$

3. $C(s) \implies D(s)$ (1,UI)

4. $D(s)$ \qquad\qquad (2,3)

Existential Instantiation (EI)

$\exists x\ P(x)$ can imply $P(a)$, where a is a particular element not previously used in a proof sequence

In English: If P is true for some element of the domain, we can give that element a specific notation.

Restrictions: a must not be used before!

Incorrect Uses of EI:

> $\exists\ P(x, a)$ CANNOT imply $P(a, a)$
> For example: in the integer domain, let $P(x, y)$ denote $x > y$ and $a = 1$

> $\forall x\ \exists y\ Q(x, y)$ CANNOT imply $\forall x Q(x, a)$
> For example: in the integer domain, let $Q(x, y)$ denote that $x > y$.

Example: $\forall x\ (P(x) \implies Q(x)) \land \exists y\ P(y) \implies Q(a)$

$$
\begin{array}{lll}
1. & \forall x\ (P(x) \implies Q(x)) & \\
2. & \exists y\ P(y) & \\
3. & P(a) & \text{(2,EI)} \\
4. & P(a) \implies Q(a) & \text{(1,UI)} \\
5 & Q(a) & \text{(3,4)}
\end{array}
$$

TODO: Update these incorrect use examples, what do these mean?

Universal Generalization (UG)

$P(c)$ can imply $\forall x\ P(x)$, where c is an arbitrary element in the domain.

In English: If $P(c)$ is true and c is arbitrary, then we can conclude $\forall x\ P(x)$

No weird restrictions or common misuses.

<u>Example</u>: $\forall x\ (P(x) \implies Q(x)) \wedge \forall x\ P(x) \implies \forall x\ Q(x)$

1.	$\forall x\ (P(x) \implies Q(x))$	
2.	$\forall x\ P(x)$	
3.	$P(c) \implies Q(c)$	(1,UI)
4.	$P(c)$	(2,UI)
5.	$Q(c)$	(3,4)
6.	$\forall x\ Q(x)$	(5,UG)

Existential Generalization (EG)

$P(a)$ can imply $\exists x \ P(x)$, where a is a particular element.

In English: Something has been named as having property P, so we can say that there exists something that has property P.

Restrictions: x must not appear in $P(a)$

Incorrect Uses of EG:

> $P(z, y)$ CANNOT imply $\exists y P(y, y)$
  For example: In the positive integer domain, let $P(x, y)$ mean that $y > x$, and a stands for 0, then $y > 0$ does not mean $y > y$

Example: $\forall x \ P(x) \implies \exists x \ P(x)$

$$
\begin{array}{lll}
1. & \forall x \ P(x) & \\
2. & P(a) & (1,\text{UI}) \\
3. \ \exists x \ P(x) & (2,\text{EG}) &
\end{array}
$$

Proving a Valid Predicate Logic Argument (examples)

General Steps:

1. Strip off the quantifiers.

2. Work with the separate statements.

3. Insert quantifiers, as necessary.

₁ $\forall x \ (P(x) \land Q(x)) \implies \forall x \ P(x) \land \forall x \ Q(x)$

1.	$\forall x \ (P(x) \land Q(x))$	
2.	$P(c) \land Q(c)$	(1,UI) ❶ You can use the same arbitrary element c for both P and Q
3.	$P(c)$	(2)
4.	$Q(c)$	(2)
5.	$\forall x \ P(x)$	(3,UG)
6.	$\forall x \ Q(x)$	(4,UG)
7.	$\forall x \ P(x) \land \forall x \ Q(x)$	(5,6)

₂ Prove the following argument is valid: "A student in this class has not attended any in-person classes. Everyone in this class passed the first exam. Therefore someone who passed the first exam has not attended any in-person classes."

Let C(x) be "x is in this class," B(x) be "x has attended in-person classes," and P(x) be "x passed the first exam." Let the domain consist of all Mines students.

$\exists x \ (C(x) \land \neg B(x)) \land \forall x \ (C(x) \implies P(x)) \implies \exists x \ (P(x) \land \neg B(x))$

❶ Its easier to figure this one out in reverse, follow the proof from the bottom up as a process of what we need preceeded by what we use to get it!

1.	$\exists x \ (C(x) \land \neg B(x))$	
2.	$\forall x \ (C(x) \implies P(x))$	
3.	$C(a) \land \neg B(a)$	(1,EI)
4.	$C(a)$	(3)
5.	$C(a) \implies P(a)$	(2, UI)
6.	$P(a)$	(4,5)
7.	$\neg B(a)$	(3)
8.	$P(a) \land \neg B(a)$	(6,7)
9.	$\exists x(P(x) \land \neg B(x))$	(8,EG)

> TODO: Add HW answers as examples later, HW had some weird situations that would be good to note.

1.6   Chapter 1 Cheatsheet

Connectives				
$\wedge$	$\vee$	$\Longrightarrow$	$\Longleftrightarrow$	$\neg$
AND	OR	IMPLIES	BICONDITIONAL	NEGATION
A and B	A or B	If A then B	A if and only if B	Not A
Both True	Either's True	$\star$	A = B	-

$\leftarrow$ true when

$\star$ - *An implication is only false when A is true and B is false.* ref truth tables (1.2.2)

Equivalence Laws	
Double Negation Laws	$\neg(\neg A) \equiv A$
Identity Laws	$A \wedge T \equiv A$ $A \vee F \equiv A$
Domination Laws	$A \vee T \equiv T$ $A \wedge F \equiv F$
Communicative Laws	$A \wedge B \equiv B \wedge A$ $A \vee B \equiv B \vee A$
Associative Laws	$(A \wedge B) \wedge C \equiv A \wedge (B \wedge C)$ $(A \vee B) \vee C \equiv A \vee (B \vee C)$
Idempotent Laws	$A \wedge A \equiv A$ $A \vee A \equiv A$
Distributive Laws	$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$ $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
DeMorgans Laws	$\neg(A \wedge B) \equiv \neg A \vee \neg B$ $\neg(A \vee B) \equiv \neg A \wedge \neg B$
Implication Laws	$A \implies B \equiv \neg B \implies \neg A$ $A \implies B \equiv \neg A \vee B$
Not a law, but helpful biconditional equivalence...	$A \iff B \equiv (A \implies B) \wedge (B \implies A)$

Inference Rules	
P $P \implies Q$ $\therefore Q$	Modus Ponens
$P \implies Q$ $\neg Q$ $\therefore \neg P$	Modus Tollens
$P \implies Q$ $Q \implies R$ $\therefore P \implies R$	Hypothetical Syllogism
$P \vee Q$ $\neg P$ $\therefore Q$	Disjunctive Syllogism
P Q $\therefore P \wedge Q$	Conjunction
$P \vee Q$ $\neg P \vee R$ $\therefore Q \vee R$	Resolution
P $\therefore P \vee Q$	Addition
$P \wedge Q$ $\therefore P$	Simplification ❶you're basically "pulling" p out to its own line

Quantified Inference Rules

❶ *These names don't really matter for this class*

> TODO: Quantified Statement Inf Rules (ref saved screenshot) + other misc helpful stuff like negations of quantifiers. Just look up and copy-paste important details ☺

2   Proofs

2.1   Proof Basics

Some New Terminology

> A **theorem** is a proposition that can be shown to be true.

> A **lemma** is a preliminary proposition useful for proving later propositions.

> A **corollary** is a proposition that can be established directly from a theorem.

> A **conjecture** is a proposition that is being proposed to be a true statement.

> Propositions that are simply accepted as true are called **axioms.**

> > ex: For all real numbers x and y, x + y = y + x

> > ex: There is a straight-line segment between every pair of points.

> A **proof** is a valid argument that establishes the truth of a statement.
> Can use axioms, premises (if any) and previously proved theorems.

Common Theorem Forms

T

> > ex: "$\sqrt{2}$ is not a rational number."

$\exists x T(x)$

> > ex: "There exists one integer n such that $n^2 + n + 41$ is composite"

> > > * You would have to find an element a in the domain such that $T(a)$ is true and then apply Existential Generalization

> > > * To disprove, prove that T(x) is false for all elements in the domain

$\forall x\ (P(x) \implies Q(x))$

> > ex: "For every integer n, if $3n + 2$ is odd, then n is odd."

> > > * You would have to show that $P(c) \implies Q(c)$, where c is an arbitrary element of the domain, and then apply Universal Generalization

> > > * Show that Q is true if P is true.

> > > * To disprove, find a element e such that $P(e)$ is true, but $Q(e)$ is false.

$\forall x\ (P \iff Q)$

> > ❶ Proving $(P \iff Q)$ is equivalent to proving $(P \implies Q) \wedge (Q \implies P)$

2.2   Proof Methods

There are four main proof methods;
  Direct Proof   Proof by Contraposition   Proof by Contradiction   Proof by Cases

---

2.2.1   Direct Proof

Directly show that if P is true, then Q must be true, using axioms, definitions, and previously proven theorems, together with inference rules.

Examples

₁ Prove that "If n is odd, then n^2 is odd."

Proof:

  Assume that n is odd, then $n = 2k + 1$, where k is some integer.

  We have $n^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$.

  Therefore, n^2 is an odd integer.

> ❶ n is odd when $n = 2(...) + 1$. We have to figure out how to transform n^2 into this form.

₂ Prove that "If m and n are both perfect squares, then mn is also a perfect square."

Proof:

  Assume that m and n are both perfect squares, then $m = s^2$ and $n = t^2$, where s and t are some integers.

  We have $mn = s^2t^2 = (st)^2$

  Therefore, mn is a perfect square.

2.2.2   Proof by Contraposition

Instead of proving $P \implies Q$, prove $\neg Q \implies \neg P$.
We do this because we can then utilize the implication law:

$$P \implies Q \equiv \neg Q \implies \neg Q$$

Examples

1 Prove that "For any integer n, if n^2 is even, then n is even."
Proof:
  Contraposition: "If n is odd, then n^2 is odd."

  (Reference example 1 of the direct proof)

  We have now proven this theorem.

2 Prove that "If $3n + 2$ is odd for an integer n, then n is odd."
Proof:
  Contraposition: "If n is even, then 3n + 2 is even"

  Assume that n is even, then $n = 2k$, where k is some integer.

  We have $3n + 2 = 6k + 2 = 2(3k + 1)$

  Therefore $3n + 2$ is even.

  We have proved the theorem "If $3n + 2$ is odd then n is odd."

3 Prove that "If r is irrational, then $\sqrt{r}$ is also irrational"
Proof:
  Contraposition: "if $\sqrt{r}$ is rational, then r is rational."

  Assume that $\sqrt{r}$ is rational.

  There exists integers p and q (no common factors), such that $\sqrt{r} = \frac{p}{q}$

  Squaring both sides gives $r = \frac{p^2}{q^2}$

  Since p^2 and q^2 are integers, r is also rational.

  This proves the theorem.

2.2.3   Proof by Contradiction

Assume we want to prove S is true.

Now, suppose we can find a contradiction C such that $\neg S \implies C$ is true.

Since C is false, but $\neg S \implies C$ is true, then S must be true.

Examples

1 Prove that '$\sqrt{2}$ is not a rational number."

Proof:

Assume that $\sqrt{2}$ is a rational number.

Then $\sqrt{2} = \frac{p}{q}$ where p and q have no common factors and $2 = \frac{p^2}{q^2}$ or $2q^2 = p^2$

Since p^2 is even, p is even (See example 1 of Proof by Contraposition). This means that 2 is a factor of p: hence 4 is a factor of p^2, and the equation $2q^2 = p^2$ can be written as $2q^2 = 4x$ for some integer x.

We have q^2 is even and thus q is even (Same Proof by Contraposition Example)

Now 2 is a factor of q and a factor of p, which contradicts that statement that p and q have no common factors.

Hence, $\sqrt{2}$ is not rational.

Proof by Contradiction (cont.)

For prepositions of the implication form $(P \implies Q)$...
We instead prove $\neg(P \implies Q) \implies$ T or $(P \wedge \neg Q) \implies$ F
So, how do you find a contradiction?

> Imply Q. Then assert $Q \wedge \neg Q$ as a contradiction.

> Imply $\neg P$. Then assert $P \wedge \neg P$ as a contradiction.

> Imply $R \wedge \neg R$ during the proof for some proposition R.

Examples
1 Prove that "If $3n + 2$ is odd for an integer n, then n is odd."
Proof:
  Assume to the contrary that $3n + 2$ is odd, and n is even.

  Since n is even, $n = 2k$, where k is some integer.

  We now have $3n + 2 = 6k + 2 = 2(3k + 1)$

  Thus $3n + 2$ is even, which contradicts the assumption $3n + 2$ is odd.

  Therefore, we have proved the theorem "If $3n + 2$ is odd, then n is odd.

2 Prove that "If a number added to itself gives itself, then the number is 0"
Proof:
  Assume to the contrary that $x + x = x$ and $x \neq 0$

  Then $2x = x$ and $x \neq 0$

  Because $x \neq 0$, we can divide both sides of the equation by x and arrive at $2 = 1$, which is a contradiction.

  Hence, $x + x = x \implies x = 0$

2.2.4 Proof By Cases

Assume that $P \equiv P_1 \vee P_2 \vee ... \vee P_n$.
Instead of proving $P \implies Q$, prove $(P_1 \implies Q) \wedge (P_2 \implies Q) \wedge ... \wedge (P_n \implies Q)$ We can do this because...

$$
\begin{aligned}
P_1 \vee P_2 \vee ... \vee P_n \implies Q \quad &\equiv \quad \neg(P_1 \vee P_2 \vee ... \vee P_n) \vee Q \\
&\equiv \quad (\neg P_1 \wedge \neg P_2 \wedge ... \wedge \neg P_n) \vee Q \\
&\equiv \quad (\neg P_1 \vee Q) \wedge (\neg P_2 \vee Q) \wedge ... \wedge (\neg P_n \vee Q) \\
&\equiv \quad (P_1 \implies Q) \wedge (P_2 \implies Q) \wedge ... \wedge (P_n \implies Q)
\end{aligned}
$$

Examples

1 Prove that "If n is an even integer, $4 \le n \le 12$ then n is the sum of two prime numbers."
Proof:
  We prove this for each value of n within the domain:

  > $n = 4 = 2 + 2$

  > $n = 6 = 3 + 3$

  > $n = 8 = 3 + 5$

  > $n = 10 = 5 + 5$

  > $n = 12 = 5 + 7$

  This completes the proof

2 Prove that "For any two numbers x and y, $\|x\|y\| = |xy|$."
Proof:
  There are 4 cases:

  1. $x \ge 0, \; y \ge 0$

     > $xy \ge 0$ and $|xy| = xy = |x||y|$

  2. $x \ge 0, \; y < 0$

     > $xy \le 0$ and $|xy| = -xy = x(-y) = |x||y|$

  3. $x < 0, \; y \ge 0$

     > $xy \le 0$ and $|xy| = -xy = (-x)y = |x||y|$

  4. $x < 0, \; y < 0$

     > $xy > 0$ and $|xy| = (-x)(-y) = |x||y|$

  Therefore, $|x||y| = |xy|$

2.3   Disproving a Statement & Proof Strategies

Disproving a Statement

Find a counterexample of the statement! For example, "Every positive integer is the sum of the squares of two positive integers"
Proof:
 3 cannot be written as the sum of squares of two integers.

 To show this, note that the only possible integers are 0 and 1. And it is not possible to write 3 by summing the squares of 0 and 1 (or just 1 twice).

What Makes a Good Proof?

- State your game plan.

- Keep a linear flow.*

- A proof is an essay, not a calculation!

- Avoid excessive symbolism.

- Revise and simplify.

- Introduce notation thoughtfully.

- Structure long proofs.

- Be wary of the "obvious"

ⓘ One of the examples was showing 2k was even!

Proof Strategies

- Understand the definitions

- Analyze the meaning of the hypothesis and the conclusion

- Prove the statement using one of the proof methods.

- Use forward and backward reasoning.

Example: Incorrect Proofs

The examples below contain proofs with some sort of highlighted issue related to the strategies and pitfalls mentioned above.

1. "The sum of two even numbers is a multiple of 4."
- Proof:
   - Let x and y be even numbers.
   - Then x = 2k and y = 2k, where k is an integer. ← x and y may not be equal.
   - So x + y = 2k + 2k = 4k, which is a multiple of 4.

2. "1/8 > 1/4."
- Proof:
   - 3 > 2
   - $3 \log_{10}(1/2) > 2 \log_{10}(1/2)$ ← The inequality symbol should be reversed when multiplying by a negative number.
   - $\log_{10}(1/2)^3 > \log_{10}(1/2)^2$
   - $(1/2)^3 > (1/2)^2$

3. "1¢ = \$1."
- Proof:
   - 1¢ = \$0.01 = (\$0.1)² = (10¢)² = \$1

   \$ ≠ \$²

43

2.4 Proof by Induction

If you can prove that you can get to $P(n + 1)$ from $P(n)$ and also prove that just $P(1)$ is true you can prove $\forall n\ P(n)$ since you're starting at 1 being true and saying anything following it will also be true.

> ❶ *if $P(1)$ is true and $\forall k\ (P(k) \implies P(k+1))$ is true. Then we just follow $P(1) \implies P(2)$, $P(2) \implies P(3)$ and so on...*

More specifically;

Let P be a predicate on positive integers. If...

1. Basis Step: $P(1)$ is true.

2. Inductive Step: $\forall k\ (P(k) \implies P(k + 1))$ is true.

   > ❯ You prove this by assuming P(k) is true for an arbitrary positive integer k and show that P(k+1) is true.

P(n) is true for all positive integers.

General Steps / Template

1. Translate into the form "For all $n \geq b, P(n)$" for a fixed integer b.

2. Write out the words "Basis step." Then show that $P(b)$ is true.

   - This is often just a basic sub-in and proving equality.

3. Write out the words "Inductive step."

   (a) State and clearly identify the inductive hypothesis, in the form "Assume that $P(k)$ is true for an arbitrary (fixed) integer $k \geq b$."

   (b) State what needs to be proved under the assumption that the inductive hypothesis is true, i.e., write out what $P(k + 1)$ says.

   (c) Prove $P(k + 1)$ to be true using the assumption P(k) is true.

4. Finally, state the conclusion.

2.4.1 Examples

1. Prove: $1 + 2 + 3 + ... + n = \frac{n(n+1)}{2}$ for any positive integer n.

   Let $f(n)$ denote "$1 + 2 + 3 + ... + n$

   Basis Step $(n = 1)$: $1 = 1(1+1)/2$ ✓

   Inductive Step:

   > Assume that for any arbitrary positive integer k, $f(k) = \frac{k(k+1)}{2}$

   > We need to show $f(k+1) = \frac{(k+1)(k+2)}{2}$ ← *here we're plugging in $k+1$ in for k*

   > $f(k+1) = 1+2+3+...+k+(k+1) = f(k)+k(k+1) = \frac{k(k+1)}{2}+(k+1) = \frac{(k+1)(k+2)}{2}$

   > Thus, the statement is true for $k + 1$ as well.

2. Prove $n^2 > 3n$ for $n \geq 4$ for any positive integer n.

   Basis step $(n = 4)$: we have $4^2 > 3 * 4$ ✓

   Inductive Step:

   > Assume that for any arbitrary positive integer $k \geq 4$, $k^2 > 3k$

   > We need to show that $(k + 1)^2 > 3(k + 1)$

   > $(k + 1)^2 = k^2 + 2k + 1$
   $> 3k + 2k + 1$ (by the inductive hypothesis)
   $\geq 3k + 8 + 1$ $(k \geq 4)$
   $> 3k + 3$
   $= 3(3 + 1)$ ← *remember this is what we were trying to reduce the RHS to prove $(k+1)^2 > 3(k+1)$*

   this completes the proof

3. Prove $7^n - 2^n$ is divisible by 5 for <u>any positive integer n</u>

   Basis Step: $(n = 1)$: we have $7 - 2 = 5$, which is divisible by 5.

   We need to show that $7^{k+1} - 2^{k+1}$ is divisible by 5

   Induction Step:

   > Assume that for any arbitrary positive integer k, $7^k - 2^k$ is divisible by 5.

   > We need to show that $7^{k+1} - 2^{k+1}$ is divisible by 5.
   $$\begin{aligned} 7^{k+1} - 2^{k+1} &= 7 * 7^k - 2 * 2^k \\ &= 7 * 7^k - 7 * 2^k + 5 * 2^k \\ &= 7 * (7^k - 2^k) + 5 * 2^k \end{aligned}$$
   By the inductive hypothesis, $7 * (7^k - 2^k)$ is divisible by 5,
   thus $7^{k+1} - 2^{k+1}$ is divisible by 5.

   this completes the proof

2.5 Strong Induction

Let P be a predicate on positive integers. If...

1. $P(1)$ is true.

    ❶ This doesn't necessarily mean integer positive 1. This is wherever the set begins.

2. $\forall k(P(1) \wedge P(2) \wedge P(3) \wedge ... \wedge P(k) \implies P(k+1))$

Then $P(n)$ is true for all positive integers.

This method has more assumptions that can be utilized for the proof as it's assuming a bunch of elements *up to $P(k)$* are true.

The template is nearly identical to our proof by induction.

Induction vs Strong Induction

- Induction applies when information about "one position back" is enough for the inductive sterp.

- Strong induction applies when you cannot directly prove that $P(k+1)$ is true just because you know $P(k)$ is true.

    ❶ *Technically, you could use either type for a proof (as their technically equivalent). However, each is significantly easier for different types of proofs.*

2.5.1 Examples

₁ Prove: every integer $n > 1$ is prime or a product of primes.

Basis step $(n = 2)$: 2 is prime.

Inductive step.

> Assume that for any integer $k \geq 2$, the statement is true for any integer r, where $2 \leq r \leq k$

> We need to show that $k + 1$ is prime or a product of primes.

  ❶we can split it into two the separate conditions using proof by cases.

   * If $k + 1$ is prime, we're done.

   * If $k+1$ isn't prime, then $k+1 = ab$, where $1 < a < k+1$ and $1 < b < k+1$, or equivalently $2 \leq a \leq k$, $2 \leq b \leq k$. Based on this assumption, a and b are either prime or the product of primes. Thus, $k + 1 = ab$ is a product of primes.

This completes the proof

₂ Prove any amount of postage greater than or equal to 8 cents can be built using only 3-cent and 5-cent stamps.

Basis Step ($n = 8$): $8 = 3 + 5$

Inductive Step

> Assume that given any positive integer $k \geq 8$, the statement is true for any integer r, where $8 \leq r \leq k$.

> Show that $k + 1$ can be built using only 3-cent and 5-cent stamps.

* Case 1: ($k + 1 = 9$): 9

* Case 2: ($k + 1 = 10$): $10 = 5 + 5$

* Case 3 ($k + 1 \geq 11$): $k + 1 = k - 2 + 3$. Since $k - 2 \geq 8$, $k - 2$ can be written as the sum of 3's and 5's. Thus $k + 1$ can be written as a sum of 3's and 5's.

This completes the proof.

Thats a lot of cases to go through in the induction steps! Let's do this example again but let's add more base cases in the basis step to avoid adding proof by cases within the inductive step.

₂.₁ Prove any amount of postage greater than or equal to 8 cents can be built using only 3-cent and 5-cent stamps.

Basis Step: ($n = 8, 9, 10$): $8 = 3 + 5, 9 = 3 + 3 + 3, 10 = 5 + 5$

Inductive Step:

> Assume that given any positive integer $k \geq 10$, the statement is true for any integer r, where $8 \leq r \leq k$

> Show that $k + 1$ can be built using only 3-cent and 5-cent stamps.

> $k + 1 = k - 2 + 3$. Since $k - 2 \geq 8$, $k - 2$ can be written as the sum of 3's and 5's.

> Thus, $k + 1$ can be written as the sum of 3's and 5's.

This completes the proof.

ⓘ this connects our inductive step to our basis step

TODO: add the tribonacci sequence exercise (its very different than the other examples covered)

3 Sets

📖 A <u>set</u> is an unordered collection of objects, which are called the members or elements of the set.

3.1 Set Basics

A Set is denoted by a capital letter with it's elements contained within curly brackets (comma seperated).

> C = {red, green, blue}

$y \in S$ means that y is a member or an element of S,
$y \notin S$ means that y is not a member or an element of S.

> red $\in$ C = {red, green, blue}, but yellow $\notin$ C.

Sets have no order!

> $\{x, y\}$ is the same as $\{y, x\}$

Each element only occurs once in a set.

> $\{x, x\} \rightarrow x$

You can have sets of sets...

> $\{\{a, b\}, \{\{b\}\}, \{\{e\}, f\}\}$

Standard Sets

$\mathbb{N}$ = Set of all nonnegative integers $\mathbb{N} = \{0, 1, 2, 3, \dots\}$
$\mathbb{Z}$ = Set of all integers $\mathbb{Z} = \{\dots, --3, -2, -1, 0, 1, 2, 3 \dots\}$
$\mathbb{Q}$ = Set of all rational numbers $\frac{1}{2}, \frac{-3}{5}, 16 \dots$
$\mathbb{R}$ = Set of all real numbers $\pi, e, -9, \sqrt{2} \dots$
$\varnothing$ = Empty set   A set with no elements.
❶Note that $\varnothing$ is not the same as $\{\varnothing\}$

> ❶ *A superscript of "+" restricts a set to its positive elements*

Describing a Set

List all the elements.

> Set of all positive even integers: S = $\{2, 4, 6, 8, \dots\}$

Characterize the property of the elements - $\{x | P(x)\}$

> Set of all positive even integers: S = $\{x | x > 0 \text{ and } x = 2k \text{ for } k \in \mathbb{N}\}$

ℹ this essentially means all x <u>such that</u> $P(x)$ is true

ℹ you can also use $\wedge$ instead of "and" here.

Size of a Set

The <u>size</u> or <u>cardinality</u> of set S is the number of elements in S denoted by $|S|$.

> A is the set of odd positive integers less than 10. $|A| = 5$.

> S is the set of letters in the English alphabet. $|S|=26$

> $|\varnothing| = 0$

> $|\mathbb{N}| =$ infinity.

Power Set

For a set S, the set consisting of **all** the subsets of S is called the <u>power set</u> of S, denoted by 2^S. $2^S = \{x \mid x \subseteq S\}$
The size of 2^S is $|2^S| = 2^{|S|}$

<u>Examples</u>

1 What is the power set of $\varnothing$?

$2^{\varnothing} = \{\varnothing\}$

$|2^{\varnothing}| = 2^0 = 1$

2 Let $S = \{a, b, c\}$. What is 2^S?

The subsets of S:

$\varnothing$

$\{a\}, \{b\}, \{c\}$

$\{a,b\}, \{a,c\}, \{b,c\}$

$\{a,b,c\}$

$2^S = \{\varnothing, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}, \{b,c\}, \{a,b,c\}\}$

3.1.1 Examples

₁ Describe the following sets by listing their elements:

1. $\{\, x \mid x \in \mathbb{N}$ and $3 < x \leq 7\,\}$

   > $\{4, 5, 6, 7\}$

2. $\{\, x \mid x$ is a month with exactly 30 days $\}$

   > $\{$ April, June, September, November $\}$

3. $\{\, x \mid x$ is the capital of the United States $\}$

   > $\{$ Washington DC $\}$

₂ Describe the following sets by giving a charactering property.

1. $\{\, 1, 4, 9, 16 \,\}$

   > $\{\, x \mid 1 \leq x \leq 16$ and $x = k^2$ for $k \in \mathbb{N} \,\}$

2. $\{\, 2, 3, 5, 7, 11, 13, 17, \ldots \,\}$

   > $\{\, x \mid x$ is prime $\}$

₃ Simplify the descriptions of the following sets:

1. $A = \{\, x \mid \exists\, y\ (y \in \{0,1,2\}$ and $x = y^3)\}$

   - $A = \{0,1,8\,\}$

2. $B = \{\, x \mid x \in \mathbb{N}$ and $\exists\, y\ (y \in \mathbb{N}$ and $x \leq y)\}$

   - $B = \mathbb{N}$

3. $C = \{\, x \mid x \in \mathbb{N}$ and $\forall\, y\ (y \in \mathbb{N} \implies x \leq y)\}$

   - $C = \{\, 0 \,\}$

3.2   Relationships Between Sets

There are three main set relationships important to this class;

$$\text{Subset} \quad \text{Equal} \quad \text{Proper Subset}$$

Relationship	Definition	Details
Subset	A is a <u>subset</u> of B if every element of A is also in element B	Denoted by: $A \subseteq B$ Logical Representation: $\forall x (x \in A \implies x \in B)$
Equal	Two sets are <u>equal</u> if and only if they contain the same elements	Denoted by A = B Logical Representation: $\forall x\ (x \in A \iff x \in B)$ or $\forall x\ (x \in A \implies x \in B) \wedge \forall x\ (x \in B \implies x \in A)$ A = B if and only if A $\subseteq$ B $\wedge$ B $\subseteq$ A
Proper Subset	If A $\subseteq$ B but A $\neq$ B, then A is a <u>proper subset</u> of B	Denoted By A $\subset$ B Logical Representation: $\forall x\ (x \in A \implies x \in B) \wedge \exists x\ (x \in B \wedge x \notin A)$

Example

₁ Let A = { 1, 7, 9, 15}, B = {7, 9}, C = {7, 9, 15, 20}
Then the following statements are all true;
  B $\subseteq$ C   {7,9} $\subseteq$ B   B $\subset$ A   {7} $\subset$ A   B $\subseteq$ A $\varnothing \subseteq$ C   15 $\in$ C

"$\varnothing \subseteq$ C" is easier to understand if you look at the logical representation:
$\forall x (x \in \varnothing \implies x \in C) \rightsquigarrow \forall x(\text{F} \implies x \in C)$
Since $\varnothing$ is an empty set, the first part of that implication is false (x cant be in nothing). Meaning the entire statement is true.

3.3 Proving the relationships

Prove that $a \in A = \{x \mid P(x)\}$

> Show that $P(a)$ is true.

Prove that $A \subseteq B = \{x \mid P(x)\}$

> Pick an arbitrary $x \in A$ and show that $P(x)$ holds.

Prove that $A \subset B = x \mid P(x)$

> Prove that $A \subseteq B$ but there exists $a\ in$ B such that $a \notin A$.

Prove that $A = B$

> $A = B$ means $\forall x(x \in A \implies x \in B) \wedge \forall x(x \in B \implies x \in A)$

> Equivalently, $A = B$ if and only if $A \subseteq B \wedge B \subseteq A$.

> Prove that $A \subseteq B$ and $B \subseteq A$.

3.3.1 Examples

[1] Let $A = \{\, x \mid x$ is a multiple of 8$\}$, and $B = \{\, x \mid x$ is a multiple of 4$\}$.
Prove that $A \subseteq B$.

ⓘ bruh

Proof:

Let $x \in$ be any arbitrary element.

We need to show that x is a multiple of 4.

Since x is a multiple of 8, $x = 8k$ for some $k \in \mathbb{Z}$

$x = 8k = 4*2k = 4m$, where $m = 2k$.

Thus x is a multiple of 4 and $x \in B$. So $A \subseteq B$.

(Some numbers, like 12, are multiples of 4 but not 8. So $A \subset B$)

₂ Let A = { x | x ∈ ℝ and x^2-4x+3 = 0}, and B = { x | x ∈ ℕ and 1 ≤ x ≤ 4}.
Prove that A ⊂ B.
Proof:

A can be written as A = 1, 3 (just solving for when x satisfies that eq)

Let x ∈ A. We need to show that x ∈ ℕ and 1 ≤ x ≤ 4. Thus x ∈ B.

If x = 3, then x ∈ ℕ and 1 ≤ x ≤ 4. Thus x ∈ B.

> Same idea for 1.

So A ⊆ B.

2 ∈ B but ∉ A. So A ⊂ B.

> ❶ Since there's only two elements, we can just prove each directly.

₃ Let A = { x | x ∈ ℕ⁺ and x^2 < 15}, and B = { x | x ∈ ℕ⁺ and 2x < 7}.
Prove that A = B.
Proof:

A ⊆ B

> Let x ∈ A. Then x can be 1, 2 or 3.

> 2x < 7 for any of these three integers.

> Thus x ∈ B and A ⊆ B.

B ⊆ A

> Let x ∈ B. Then x can be 1, 2 or 3.

> x^2 < 15 for any of these three integers.

> Thus x ∈ A and B ⊆ A.

Therefore, we proved that A = B.

> ❶ We're essentially just showing that the items of each of these sets are the same.

3.4   Cartesian Product

The <u>Cartesian product</u> of A and B, denoted by A $\times$ B, is defined as A $\times$ B $= \{(x,y) \mid x \in$ A $\land$ y $\in$ B $\}$.
This essentially means all the possible ordered pairs you can make with the items in A and B.

A $\times \varnothing = \varnothing$, $\varnothing \times$ A $= \varnothing$

A $\times$ B $=$ B $\times$ A if

> A $=$ B or

> A or B is $\varnothing$ <small>as this would satisfy that first bullet point idea.</small>

$|A \times B| = |A| \times |B|$

$A^n = A \times A \times ... \times A$. (n times)

> $A^2 = A \times A$

> $A^3 = A \times A$ *times* A.
<small>Note that these multi-products creates an n-tuple set $(\_, \_, \_, ...)$</small>

In general, $(A \times B) \times C \neq A \times (B \times C) \neq A \times B \times C$.
$$((\_, \_), \_) \qquad (\_, (\_, \_)) \qquad (\_, \_, \_)$$
<small>These would only be equal when A, B, or C = $\varnothing$</small>

Examples

[1] A $= \{1,2\}$, B $= \{2,4\}$

A $\times$ B $= \{(1,2), (1,4), (2,2), (2,4)\}$

B $\times$ A $= \{(2,1), (4,1), (2,2), (4,2)\}$

[2] A $= \{1,2\}$

$A^2 = \{(1,1), (1,2), (2,1), (2,2)\}$

$A^3 = \{(1,1,1), (1,1,2), (1,2,1), (1,2,2), (2,1,2), (2,2,1)\ (2,2,2)\}$

3.5   Set Operations

Union   Intersection   Set Difference   Complement

Before each of these operations is introduced, let's look at the <u>Venn Diagram</u> as it is an important tool to demonstrate what these operations are "pulling" from the sets.
A set S is called the <u>universal set</u> if it contains all objects <u>under consideration</u>

A Venn diagram is used to visualize all possible relations of sets. All within this universal set S (depicted as a box around the Venn Diagram). Reference the image next to every set operations description for a quick visualization of the set relation.

> ❶ If we're considering all real numbers then $S = \mathbb{R}$. If we're considering students at Mines, then the set is all students at Mines.

> ❶ *Note that while Venn diagrams are helpful for illustrating an operation,*
> *<u>they should not be used for a proof</u>*

Union

The union of A and B, denoted by A $\cup$ B is $\{x | x \in A \vee x \in B \}$
All the elements in A combined with all the elements in B

<u>Example</u>
  Let A = {1, 3, 5, 7, 9} and B = {3, 5, 6, 10, 11}
  Then A $\cup$ B = {1, 3, 5, 6, 7, 9, 10, 11}
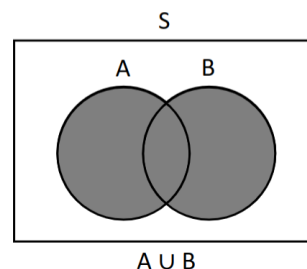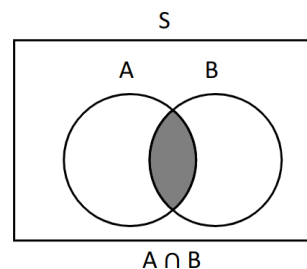

Intersection

The intersection of A and B, denoted by A $\cap$ B is $\{x | x \in A \wedge x \in B \}$
The elements that are in both A and B

<u>Example</u>
  Let A = {1, 3, 5, 7, 9} and B = {3, 5, 6, 10, 11}.
  Then A $\cap$ B = {3,5}

Set Difference

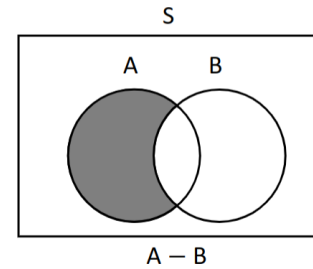The set difference of A and B, denoted by A - B, is $\{x | x \in A \wedge x \notin B\}$

<small>The elements that are in A but not in B</small>

Note: A - B = A - (A ∩ B)

Example

Let A = {1, 3, 5, 7, 9} and B = {3, 5, 6, 10, 11}

Then A - B = {1, 7, 9}

Complement

Let A ⊆ S. The complement of A, denoted by A′, is $\{x | x \in S \wedge x \notin A\}$

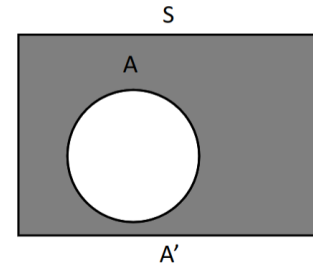<small>The set of everything <u>not</u> in A</small>

Example

S = ℕ

$A = \{x | x \in \mathbb{N} \wedge x = 2k \text{ for } k \in \mathbb{N}\}$

$A' = \{x | x \in \mathbb{N} \wedge x = 2k + 1 \text{ for } k \in \mathbb{N}\}$

Set Difference and Intersection

Two sets A and B are <u>disjoint</u> if A ∩ B = ∅.

<small>Meaning they have no elements in common</small>
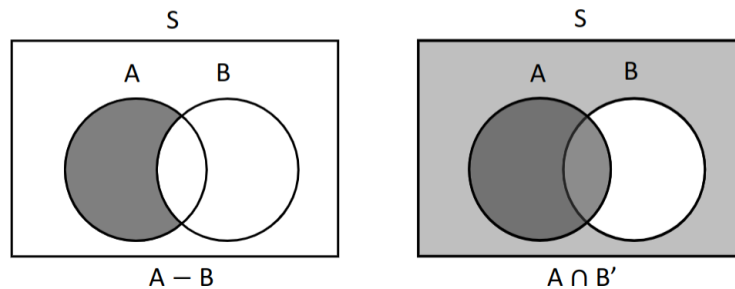
If A and B are disjoint, then A - B = A and B - A = B

For any A ⊆ S, B ⊆ S, A - B and B - A are disjoint.
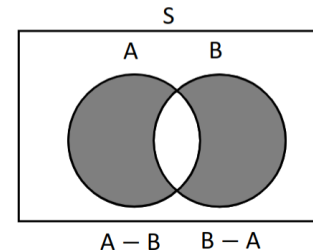
Set Difference and Complement

The set difference can be rewritten as $\{x | x \in A \wedge x \in B'\}$. Therefore, A - B = A ∩ B′

The darkest shaded part is what is actually included by the operation A ∩ B′, the other shades are the "layers" of each operation

3.5.1 Proving Relations Involving Set Operations

(Examples)

In general, its good to start by looking at the Venn Diagram for visually understanding why a statement is true, and deriving the formal proof from there.

₁ Prove that $A \cup B \subseteq A$

We need to show that for some arbitrary element in A, that element is also in B.

$\rightsquigarrow \quad x \in A \implies x \in B$

Proof:

Let x be an arbitrary member of $A \cap B$.

$x \in A \cap B \quad \implies \quad x \in A \land x \in B$ ₌ₛᵢₙ𝓰 ₜₕₑ 𝒹ₑ𝒻ᵢₙᵢₜᵢₒₙ ₒ𝒻 ᵢₙₜₑᵣₛₑ𝒸ₜᵢₒₙ ₕₑᵣₑ

$\qquad\qquad\qquad \implies \quad x \in A$ ₜᵣₑₐₜ ₓ ∈ ₐ ∧ ₓ ∈ ᵦ ₐₛ ₚ ∧ ◵ ₐₙ𝒹 ₌ₛₑ ₛᵢₘₚₗᵢ𝒻ᵢ𝒸ₐₜᵢₒₙ ᵢₙ𝒻' ₗₐₓ

Therefore, $A \cap B \subseteq A$. ᵦy ₜₕₑ 𝒹ₑ𝒻ᵢₙᵢₜᵢₒₙ ₒ𝒻 ₛₗᵦₛₑₜ ᵣₑₗₐₜᵢₒₙ

Or intuitively we've shown that any x in A ∩ B is also in just A

₂ Prove that $C \subseteq A$ and $C \subseteq B$ if and only if $C \subseteq A \cap B$.

Remember we can break $P \iff Q$ into $(P \implies Q) \land (Q \implies P)$ where P is "C ⊆ A and C ⊆ B" and Q is "C ⊆ A ∩ B"

Proof:

First we need to prove the "only if" ₍ₚ ⟹ ◵₎ part:

Let x be an arbitrary member of C.

$x \in C \quad \implies \quad x \in A \land x \in B$ ₛᵢₙ𝒸ₑ 𝒸 ⊆ ₐ ₐₙ𝒹 𝒸 ⊆ ᵦ

$\qquad\qquad \implies \quad x \in A \cap B$ ᵣₑ𝒻ₑᵣₑₙ𝒸ₑ ₜₕₑ ᵢₙₜₑᵣₛₑ𝒸ₜᵢₒₙ 𝒹ₑ𝒻ᵢₙᵢₜᵢₒₙ.

Now we prove the "If" ₍◵ ⟹ ₚ₎ part:

Let x be an arbitrary member of C.

$x \in C \quad \implies \quad x \in A \cap B$ 𝒻ᵣₒₘ ₕyₚₒₜₕₑₛᵢₛ "𝒸 ⊆ ₐ ∩ ᵦ"

$\qquad\qquad \implies \quad x \in A \land x \in B$ 𝒻ᵣₒₘ ₜₕₑ ᵢₙₜₑᵣₛₑ𝒸ₜᵢₒₙ 𝒹ₑ𝒻ᵢₙᵢₜᵢₒₙ ₐ𝓰ₐᵢₙ.

Therefore, $C \subseteq A \land C \subseteq B$ ᵣₑ𝒻ₑᵣₑₙ𝒸ₑ ₜₕₑ ₗₒ𝓰ᵢ𝒸ₐₗ ᵣₑₚ' ₒ𝒻 ₐ ₛₗᵦₛₑₜ ₍𝒻ᵣₒₘ ₜₕₑ ₜₐᵦₗₑ₎

₃ Prove that $2^A \cap 2^B = 2^{A \cap B}$

You can prove two sets to be equal by proving Left $\subseteq$ Right and Right $\subseteq$ Left

⇝ *Prove $2^A \cap 2^B \subseteq 2^{A \cap B}$ and $2^{A \cap B} \subseteq 2^A \cap 2^B$*

Proof:

$2^A \cap 2^B \subseteq 2^{A \cap B}$

  Let x be an arbitrary member of $2^A \cap 2^B$.

  $x \in 2^A \cap 2^B \implies x \in 2^A \wedge x \in 2^B$

  $\implies x \subseteq A \wedge x \subseteq B$ Ref Power Set Definition, If x is a member of a power set, its also a member of the main set

  $\implies x \subseteq A \cap B$ Intersection Def-kinda ref examples above

  $\implies x \in 2^{A \cap B}$ Reverse use of the same Power Set Def, if x is a member of this set, its also a member of the power set

  Therefore $2^A \cap 2^B \subseteq 2^{A \cap B}$

$2^{A \cap B} \subseteq 2^A \cap 2^B$

  Let x be an arbitrary member of $2^{A \cap B}$

  $x \in 2^{A \cap B} \implies x \subseteq A \cap B$

  $\implies x \subseteq A \wedge x \subseteq B$

  $\implies x \in 2^A \wedge x \in 2^B$

  Therefore, $2^{A \cap B} \subseteq 2^A \cap 2^B$.

₄ Prove that if $A \neq \varnothing$ and $A \cap B = A - B$, then $B = \varnothing$

Both a direct proof and proof by contraposition lead to you trying to prove a set is empty, which is very difficult. So the best option is proof by contradiction

Proof:

Assume to the contrary that $A \neq \varnothing$, $A \cap B = A - B$ and $B \neq \varnothing$

Case 1: $A \cap B = \varnothing$ ($A \cap B$ is empty)

    $A \cap B = A - B$ (from second assumption)

    $A - B = A$ (ref "set difference and intersection" notes.)

    All combining to $A \cap B = \varnothing = A - B = A \neq \varnothing$

    "$\varnothing \neq \varnothing$", We have a contradiction.

Case 2: $A \cap B \neq \varnothing$

    Let x be any element in A ∩ B.

    $x \in A \cap B \implies x \in A - B$ (from $A \cap B = A - B$)

    $\implies x \notin B$ (defn of a set difference)

    This is a contradiction, since $x \in A \cap B$ implies that $x \in B$

3.6   Set Identities

📖 Set identities are a group of laws that can be used to prove equality between sets. They are all equalities, involving union, intersection, difference, and complementation, that are <u>always true</u> for all subsets of a given set S.

Set Identities	
Identity Laws	$A \cup \varnothing = A$ $A \cap S = A$
Domination Laws	$A \cup S = S$ $A \cap \varnothing = \varnothing$
Idempotent Laws	$A \cup A = A$ $A \cap A = A$
Complementation Law	$(A')' = A$
Commutative Laws	$A \cup B = B \cup A$ $A \cap B = B \cap A$
Associative Laws	$(A \cup B) \cup C = A \cup (B \cup C)$ $(A \cap B) \cap C = A \cap (B \cap C)$
Distributive Laws	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
DeMorgan's Laws	$(A \cap B)' = A' \cup B'$ $(A \cup B)' = A' \cap B'$
Absorption Laws	$A \cup (A \cap B) = A$ $A \cap (A \cup B) = A$
Complement Laws	$A \cup A' = S$ $A \cap A' = \varnothing$

TODO: Add venn diagram images (pictures including the rule is fine)

3.6.1　Proving Identities
(Examples)

₁ Prove that A ∪ (B ∩ C) = (A ∪ B) ∩ (A ∪ C)

Proof:

We need to prove:

A ∪ (B ∩ C) ⊆ (A ∪ B) ∩ (A ∪ C)

(A ∪ B) ∩ (A ∪ C) ⊆ A ∪ (B ∩ C)

Prove that A ∪ (B ∩ C) ⊆ (A ∪ B) ∩ (A ∪ C):

Let x be an arbitrary member of A ∪ (B ∩ C)

$$
\begin{aligned}
x \in A \cup (B \cap C) \quad &\implies \quad x \in A \vee x \in (B \cap C) \\
&\implies \quad x \in A \vee (x \in B \wedge x \in C) \\
&\implies \quad (x \in A \vee x \in B) \wedge (x \in A \vee x \in C) \\
&\implies \quad (x \in A \cup B) \wedge (x \in A \cup C) \\
&\implies \quad x \in (A \cup B) \cap (A \cup C)
\end{aligned}
$$

Prove that (A ∪ B) ∩ (A ∪ C) ⊆ A ∪ (B ∩ C)

Let x be an arbitrary member of (A ∪ B) ∩ (A ∪ C).

$$
\begin{aligned}
x \in (A \cup B) \cap (A \cup C) \quad &\implies \quad (x \in A \cup B) \wedge (x \in A \cup C) \\
&\implies \quad (x \in A \vee x \in B) \wedge (x \in A \vee x \in C) \\
&\implies \quad x \in A \vee (x \in B \wedge x \in C) \\
&\implies \quad x \in A \vee x \in (B \cup C) \\
&\implies \quad x \in A \cup (B \cap C)
\end{aligned}
$$

TODO: Add exercises if needed

4 Recursion

📖 A <u>recursive definition</u> is a definition in which the object being defined appears as part of the definition

---

There are two parts of a recursive definition:

Basis Step: some simple cases of the object being defined are explicitly given (end of the recursion).

Recursive step: New cases of the object being defined are given in terms of "previous" or "simpler" cases.

The relation in the recursive step is called the <u>recurrence relation</u>.

Examples

1. Give a recursive definition for the ancestors of James.
   Recursive Definition:

   - Basis Step: James' parentsd are ancestors of James.

   - Recursive Step: The parents of James' ancestors are ancestors of James.

2. Give a recursive defintion of the exponentiation operation a^n on a nonzero real number a, where n is a nonnegative integer.
   Recursive Definition:

   - Basis Step: $a^0 = 1$

   - Recursive Step: $a^n = (a^n - 1)a$ for $n \geq 1$

4.1 Sequences

A <u>sequence</u> is an ordered list of elements, denoted by,

$$S_1, S_2, S_3, ...$$

Where S_k denotes the kth element in the sequence.

Examples:

1,2,3,5,8 is a sequence of 5 elements.

1,3,9,27,81,...,3^n,... is an infinite sequence.

Summations

Given the sequence $a_1, a_2, a_3, ...a_n$ we use the notation

$$\sum_{i=m}^{n} a_i$$

to denote the sum of the terms $a_m, a_{m+1}, ..., a_n$. Read as "the sum from $i = m$ to $i = n$ of a$_i$"

Examples:

$1 + 2 + ... + n = \sum_{i=1}^{n} i$

$\sum_{j=1}^{5} = 1^2 + 2^2 + 3^2 + 4^2 + 5^2$

Recursively Defined Sequences

A sequence might be defined recursively by explicitly naming the first value (or the first few values) and then defining later values n terms of previous values.

Examples

1 Sequence F_n

Basis Step: $F_1 = 1, F_2 = 1$

Recursive step: $F_n = F_{n-1} + F_{n-2}$ for $n > 2$

The sequence is 1,1,2,3,5,8,13,21,... Otherwise known as the Fibonacci sequence.

2 Give a recursive definition of the sequence $a_1, a_2, a_3, ...$, where $a_n = 4n - 2$ for $n \geq 1$

Basis Step: $a_1 = 2$

Recursive step: $a_n = a_{n-1} + 4$ for $n \geq 2$

TODO (if needed): Add Fibbonacci Proof.

Recursively Defined Sets

A set might be defined recursively by explicitly describing one or more <u>simple elements</u> and then defining other elements <u>in terms of existing simpler elements</u> in the set.

1 The set of positive integers, denoted by $\mathbb{N}^+$ has the recursive definition:

1 is $\mathbb{N}^+$

If n is in $\mathbb{N}^+$, then n+1 is in $\mathbb{N}^+$

2 Give a recursive definition of the set S of positive integers that are multiples of 5.

Basis Step: $5 \in S$

Recursive step: If $x \in S$, then $x + 5 \in S$ (or $x \in S \implies x + 5 \in S$)

3 Give a recursive definition of the set of strings, that, is $\sum^* =$"", "a", "amigien", "words", "school", "ziemg"...
Notations:

λ is the empty string "".

Σ is the set of all letters a,b,c...

Let $x \in \Sigma^*$ and $y \in \Sigma^*$. xy is the concatenation of x and y.

Recursive Definition:

Basis step: $\lambda \in \Sigma^*, x \in \Sigma^*$ where $x \in \Sigma$

Recursive step: $x \in \Sigma^* \wedge y \in \Sigma^* \implies xy \in \Sigma^*$

> There's a ton of exercises and further descriptions of the examples in the videos and slides, add those if needed!

Recursively Defined Sequence

4.2   Closed Form Solution

Intro/Motivation

Consider the following recursively defined sequence;

Basis step: $S_1 = 2$

Recursive Step: $S_n = 2S_{n-1}$ for $n \geq 2$

S would be the sequence 2,4,8,16,32 $_{notice \ a \ pattern \ here?}$

If you were to represent this sequence in code you'd get something like one of these;

```
int S_prev = 2;                    int CompSn(int n)
for(int i = 2; i <= n; ++i)        {
{                                      if(n == 1)
    S = 2*S_prev;                          return 2;
    S_prev = S;                        return 2*CompSn(n-1);
}                                  }
```

Note that both of these would be O(n), quite a gross time complexity for something with a clear trend.

Looking back at our sequence S_n is just 2^n! In other words, our two sets of code could be replaced with an $O(\log n)$ `int Sn = pow(2,n)`.

ⓘ This time complexity comes from the implementation of pow

Definitions

What we did above is an example of finding/solving the **closed-form solution**.
An equation, where we can substitute values and get the output value back directly, is called a closed-form solution.
Another example would be $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ as the closed-form solution to $ax^2 + bx + c = 0$.
Note that an algorithm is <u>not</u> a closed-form solution because you cannot get an output value directly.
Not all kinds of recurrence relations can be solved.

A recurrence relation for a sequence $S_1, S_2, S_3, ...$ is a <u>linear recurrence relation</u> if S_n is a linear function of the previous elements.

> The <u>general linear recurrence relation</u> has the form:
> $S_n = f_1(n)S_{n_1} + f_2 S_{n-2} + ...f_k S_{n-k} + g(n)$
> where $1 \leq k \leq n - 1$, the f_i's and g <u>can</u> be expressions involving n.

> The recurrence relation has <u>constant coefficients</u> if the f_i's are all constants.

A recurrence relation is <u>first-order</u> if the nth element depends only on the $(n - 1)$th element.

	$S_n = 2S_{n-1}$	Linear, first-order.
<u>examples:</u>	$F_n = F + n - 1 + F_{n-2}$	Linear but not first order.
	$S_n = S_{n-1}S_{n-2}$	Not Linear.

> The general form solution with a constant coefficient $S_n = cS_{n-1} + g(n)$ for $n \geq 2$ can be expanded k times to look like;
> $c^k S_{n-k} + c^{k-1}g(n - (k - 1)) + ... + cg(n - 1) + g(n)$
> This series of expansions will stop when $n - k = 1$ So we can say that $k = n - 1$. Giving us a better solution of $c^{n-1}S_1 + c^{n-2}g(2) + ... + cg(n-1) + c^0 g(n)$. Finally we can simplify this further by using summation notation;
> $S_n = c^{n-1}S_1 + \sum_{i=2}^{n} c^{n-i}g(i)$
> So to solve for linear first-order, constant coeff recurrence relations you must identify c and $g(n)$ and plug into the form above. You can identify c and $g(n)$ by arranging the recurrence relation into the $S_n = cS_{n-1} + g(n)$ form.

Examples

1 Solve $S_n = 2S_{n-1} + 3$ for $n \geq 2$ with $S_1 = 4$

Match and identify c and g(n)
$c = 2 \quad g(n) = 3$

Substitute in $S_n = c^{n-1}S_1 + \sum_{i=2}^{n} c^{n-i}g(i)$
$$
\begin{aligned}
S_n &= 2^{n-1} * 4 + \sum_{i=2}^{n} 2^{n-1} * 3 \\
&= 2^n + 1 + 3(2^{n-1} - 1) \quad \text{\footnotesize We're Simplifying the summation here by using the identity: } 1 + 2 + 2^2 + ... + 2^m = 2^{m+1} - 1 \text{ where } m = n - 2
\end{aligned}
$$

2 Solve $T_n = T_n - 1 + (n+1)$ for $n \geq 2$ with $T_1 = 2$

Match and identify c and g(n)
$c = 1 \quad g(n) = n + 1$

Substitute
$$
\begin{aligned}
T_n &= 1^{n-1} * 2 + \sum_{i=2}^{n} 1^{n-i}(i+1) \\
&= 2 + (3 + 4 + ... + (n+1)) \\
&= \frac{(n+1)(n+2)}{2} - 1 \quad \text{\footnotesize We're simplifying using the identity: } 1 + 2 + 3 + ... + m = \frac{m(m+1)}{2}
\end{aligned}
$$