Class 2: Formalize, Symbolize

PHILOSOPHY 201: INTRODUCTION TO LOGIC
WITH ZEE PERRY

First, some admin

I STILL have not been granted a Canvas site, so continue to use this website to access the book & syllabi:

- www.zrperry.com/logic201-spring2020

This site will *also* be populated with:

- PDF versions of the slides from previous classes (for review or as a supplemental source when studying)
- This week's homework assignment(!!)

First, some admin

I STILL have not been granted a Canvas site, so continue to use this website to access the book & syllabi:

- www.zrperry.com/logic201-spring2020

This site will *also* be populated with:

- PDF versions of the slides from previous classes (for review or as a supplemental source when studying)
- ***This week's homework assignment***

Review of Previous Class: Arguments

What is a **sentence**, in logic?

Sentences are *full declarative statements*

i.e. they can be true/false

What are **arguments**, in logic?

Arguments are collections of (any number of) sentences

- with ***one*** sentence designated the "**conclusion**"
- and ***all other*** sentences called the "**premises**"

What is **the purpose of an argument**, in logic?

Arguments are supposed to provide *rationally compelling* reason to accept their conclusion.

The premises are supposed to ***support*** the conclusion.

Review of Previous Class: *Good* Arguments (and Valid-ish Arguments)

An argument whose premises "support" its conclusion, in some way or another, is what we'll call "Good", "Strong", or "Valid-ish"

An argument, in general, is Good when its premises (if you assume they're true) give you **good reason** to accept the argument's conclusion.

- Good arguments **don't** always *guarantee* their conclusions. Some merely make it *very likely* that the conclusion is true.
  - (Such arguments are called "inductively strong" arguments, but I won't test you on that information. So feel free to forget it)

1. All fifteen people who ate some of that expired Halloween candy got sick.
2. You just ate some of that expired Halloween candy
3. You're about to get sick.

Review of Previous Class: *Deductively Valid* Arguments

The strongest way that an argument's premises can "support" its conclusion is for them to **logically imply** that conclusion. In logic, we call these arguments "**Deductively Valid**" or just "**Valid**".

An argument is valid when its premises are enough to ***guarantee*** the truth of the argument's conclusion, no matter what.

We don't need to *know* whether or not an argument's premises and conclusion are *actually* **true** or **false** to figure out if they're valid!

This argument is valid, even if you didn't just eat expired candy.

1. Every people who eats some of that expired Halloween candy will get sick.
2. You ate some of that expired Halloween candy.
3. You will get sick.

Valid arguments hold in “every case”

If an argument is **Valid**, its premises imply its conclusion no matter what.

- You can dream up any scenario you want, and, so long as the premises are true in the scenario, then the conclusion *must* be true (assuming the argument is valid)

Invalid arguments have Counterexamples.

- Counterexamples are: “cases” that show the argument is bad/invalid.
- Where a “case” is a possible hypothetical scenario (*waves hands*)

Invalid arguments have Counterexamples.

Valid arguments hold in every case.

Invalid arguments don’t hold in “every case”. They have **Counterexamples**

- Counterexamples are: “cases” that show the argument is bad/invalid.
- Where a “case” is a possible hypothetical scenario (*waves hands*)
- In the scenario, **the premises are all true**, but **the conclusion is false**.

Constructing Counterexamples

A **counterexample** to an invalid argument is a "case" (hypothetical scenario), where the argument's premises are true and the conclusion is false.

Suppose we want to prove *this* argument is invalid.

1. If it's raining, I'll get my hair wet after class. (want it True)
2. I got my hair wet after class. (want it True)
3. Therefore, It was raining. (want it False)

Make premises true: Premise 1 is true if, in our scenario, we make rain behave normally, and assume we don't have a hat or umbrella.

Premise 2 could be made true in our scenario in a number of different ways:

- We could have our hair get wet from the rain, but that would conflict with our *other* goal: **make the conclusion false**.
- We could have our hair get wet some *other way*. Would that be possible? Consider, e.g. if you were on the swim team (and had practice after class), or if you took a shower before going to bed.

Counterexamples and Validity

If an argument has a counterexample, then it must be **Invalid**.

If an argument has ***no*** counterexamples
*meaning it's **impossible** to conceive of a scenario where the premises are all true and the conclusion is false*
then it must be **Valid**.

So a **Valid** argument is one where it's **impossible** for all the premises to be true and the conclusion false.

Let's make this precise: Consequence

An argument is Valid if its conclusion is a **logical consequence** of its premises. We can define logical consequence in a general way:

Let's make this precise: Consequence

An argument is Valid if its conclusion is a **logical consequence** of its premises. We can define logical consequence in a general way:

A sentence, call it "C", is a **logical consequence** of a list of other sentences, call them "P_1", "P_2", "P_3" (and so on until "P_n"), if and only if there is **no case** where P_1, P_2, …. , P_n are all true and C is not true.

Let's make this precise: Consequence

An argument is Valid if its conclusion is a **logical consequence** of its premises. We can define logical consequence in a general way:

A sentence, call it "C", is a **logical consequence** of a list of other sentences, call them "P_1", "P_2", "P_3" (and so on until "P_n"), if and only if there is **no case** where P_1, P_2, ... , P_n are all true and C is not true.

When C is a logical consequence of P_1, P_2, ... , P_n, we sometimes just say C is "a **consequence** of" P_1, P_2, ... , P_n, or C "**follows from**" P_1, P_2, ... , P_n. We also say P_1, P_2, ... , P_n "**entail**" C.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

If you didn't know what "hair" was, could you know that this argument is valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

Validity (often) cares only about *Form*

What makes an argument valid?

If you didn't know what "hair" was, could you know that this argument is valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

Validity (often) cares only about *Form*

What makes an argument valid?

If you didn't know what "hair" was, could you know that this argument is valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If it's raining, then I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If **it's raining**, then I'll get my hair wet after class.
2. **It's raining**.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If __[same sentence]__, then I'll get my hair wet after class.
2. __[same sentence]__.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If dogs are mammals, then I'll get my hair wet after class.
2. dogs are mammals.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If [same sentence], then I'll get my hair wet after class.
2. [same sentence].
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If __dogs are reptiles__, then I'll get my hair wet after class.
2. __[same sentence]__.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If dogs are reptiles, then I'll get my hair wet after class.
2. It's sunny at midnight.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

If you didn't know what "hair" was, could you know that this argument is valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If __dogs are reptiles__, then I'll get my hair wet after class.
2. __dogs are reptiles__.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If ___[same sentence]___, then I'll get my hair wet after class.
2. ___[same sentence]___.
3. So, I'll get my hair wet after class.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If [same sentence], then **I'll get my hair wet after class**.
2. [same sentence].
3. So, **I'll get my hair wet after class**.

Validity (often) cares only about *Form*

What makes an argument valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

If you didn't know what "hair" was, could you know that this argument is valid?

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If [same sentence], then **I'll get my hair wet after class**.
2. [same sentence].
3. So, **I'll get my hair wet after class**.

Validity (often) cares only about *Form*

What makes an argument valid?

If you didn't know what "hair" was, could you know that this argument is valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If __[same sentence]__, then __[same sentence]__.
2. __[same sentence]__.
3. So, __[same sentence]__.

Validity (often) cares only about *Form*

What makes an argument valid?

If you didn't know what "hair" was, could you know that this argument is valid?

1. If it's raining, I'll get my hair wet after class.
2. It's raining.
3. So, I'll get my hair wet after class.

The rain argument is valid for ***the same reason*** that *this* one's valid:

1. If it's snowy, I'll get my fingers cold right now.
2. It's snowy.
3. So, I'll get my fingers cold right now.

1. If I'm tired, I'll have lots of energy in class.
2. I'm tired.
3. So, I'll have lots of energy in class.

These arguments are all valid because they have the same *form.*

1. If ___[same sentence]___, then ___[NEW same sentence]___.
2. ___[same sentence]___.
3. So, ___[NEW same sentence]___.

Let's simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ___[same sentence]___, then ___[NEW same sentence]___.
2. ___[same sentence]___.
3. So, ___[NEW same sentence]___.

Brief Aside:
Natural and Formal Languages

Natural Language is a language that is produced naturally, by whatever processes produce the usual languages, dialects, and vocabularies in human societies/communities.

- Natural Language sentences have a logical form, but these forms are usually complicated, and can involve many different grammatical forms, differing aspects and tones, or other linguistic wrinkles.

In logic, we want to understand **logical consequence** in a way that's *systematic* and *general*, which is really difficult if we're working with a natural language.

- To make our job easier, we construct a **Formal Language** that (while limited in what it can express) is much *simpler in form* and always behaves the same way.

Brief Aside:
Natural and Formal Languages

Natural Language is a language that is produced naturally, by whatever processes produce the usual languages, dialects, and vocabularies in human societies/communities.

- Natural Language sentences have a logical form, but these forms are usually complicated, and can involve many different grammatical forms, differing aspects and tones, or other linguistic wrinkles.

In logic, we want to understand **logical consequence** in a way that's *systematic* and *general*, which is really difficult if we're working with a natural language.

- To make our job easier, we construct a **Formal Language** that (while limited in what it can express) is much *simpler in form* and always behaves the same way.

Brief Aside:
Natural and Formal Languages

Natural Language is a language that is produced naturally, by whatever processes produce the usual languages, dialects, and vocabularies in human societies/communities.

- Natural Language sentences have a logical form, but these forms are usually complicated, and can involve many different grammatical forms, differing aspects and tones, or other linguistic wrinkles.

In logic, we want to understand **logical consequence** in a way that's *systematic* and *general*, which is really difficult if we're working with a natural language.

- To make our job easier, we construct a **Formal Language** that (while limited in what it can express) is much *simpler in form* and always behaves the same way.

Brief Aside:
Natural and Formal Languages

Natural Language is a language that is produced naturally, by whatever processes produce the usual languages, dialects, and vocabularies in human societies/communities.

- Natural Language sentences have a logical form, but these forms are usually complicated, and can involve many different grammatical forms, differing aspects and tones, or other linguistic wrinkles.

In logic, we want to understand **logical consequence** in a way that's *systematic* and *general*, which is really difficult if we're working with a natural language.

- To make our job easier, we construct a **Formal Language** that (while limited in what it can express) is much *simpler in form* and always behaves the same way.

Let's simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ___[same sentence]___, then ___________[NEW same sentence]___________.
2. ___[same sentence]___.
3. So, ___________[NEW same sentence]___________.

Let's simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If __[same sentence]__, then __[NEW same sentence]__.
2. __[same sentence]__.
3. So, __[NEW same sentence]__.

Let's introduce a ***Formal Language*** to let us express what all these arguments have *in common* (by ignoring what makes them different)

Let's simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ___[same sentence]___, then ___[NEW same sentence]___.
2. ___[same sentence]___.
3. So, ___[NEW same sentence]___.

Let's introduce a ***Formal Language*** to let us express what all these arguments have *in common* (by ignoring what makes them different)

We'll use **Italicized Capital Letters** to stand-in for **Sentences**.

- The *same letter* will always refer to the *same sentence* (in any given argument)

Let’s simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ___ A ___, then ___ [NEW same sentence] ___.
2. ___ [same sentence] ___.
3. So, ___ [NEW same sentence] ___.

Let’s introduce a ***Formal Language*** to let us express what all these arguments have *in common* (by ignoring what makes them different)

We’ll use **Italicized Capital Letters** to stand-in for **Sentences**.

- The *same letter* will always refer to the *same sentence* (in any given argument)

Let’s simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ___A___, then ___[NEW same sentence]___.
2. ___A___.
3. So, ___[NEW same sentence]___.

Let’s introduce a ***Formal Language*** to let us express what all these arguments have *in common* (by ignoring what makes them different)

We’ll use **Italicized Capital Letters** to stand-in for **Sentences**.

- The *same letter* will always refer to the *same sentence* (in any given argument)

Let's simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ____A____, then __________B__________.
2. ____A____.
3. So, __________B__________.

Let's introduce a ***Formal Language*** to let us express what all these arguments have *in common* (by ignoring what makes them different)

We'll use **Italicized Capital Letters** to stand-in for **Sentences**.

- The *same letter* will always refer to the *same sentence* (in any given argument)

Let’s simplify by symbolizing!

Any argument with the following *form* will be valid, but we need a less complicated and less unclear way to **describe that form.**

1. If ____A____, then ____________B____________.
2. ____A____.
3. So, ____________B____________.

Let’s introduce a ***Formal Language*** to let us express what all these arguments have *in common* (by ignoring what makes them different)

We’ll use **Italicized Capital Letters** to stand-in for **Sentences**.

- The *same letter* will always refer to the *same sentence* (in any given argument)

What’s left of our language?

In this new formal language, entire sentences are replaced by single letters. What this means is that statements that seem to be saying ***very different*** things in English will get translated into the *same* thing.

Quizlet Q1: How do you translate these sentences?

- If it’s snowy, then I’ll get my fingers cold right now.
- If you build a house, then there’s a red pipe in this room.
- If you think about it, then you’ll realize all cats are just holograms.

What's left of our language?

In this new formal language, entire sentences are replaced by single letters. What this means is that statements that seem to be saying ***very different*** things in English will get translated into the *same* thing.

Quizlet Q1: How do you translate these sentences?

- If it's snowy, then I'll get my fingers cold right now.
- If you build a house, then there's a red pipe in this room.
- If you think about it, then you'll realize all cats are just holograms.

What's left of our language?

In this new formal language, entire sentences are replaced by single letters. What this means is that statements that seem to be saying ***very different*** things in English will get translated into the *same* thing.

Quizlet Q1: How do you translate these sentences?

- If it's snowy, then I'll get my fingers cold right now.
- If you build a house, then there's a red pipe in this room.
- If you think about it, then you'll realize all cats are just holograms.

- if A, then B
- if B, then A
- if X, then G
- A
- Q

What's left of our language?

In this new formal language, entire sentences are replaced by single letters. What this means is that statements that seem to be saying ***very different*** things in English will get translated into the *same* thing.

Quizlet Q1: How do you translate these sentences?

- If it's snowy, then I'll get my fingers cold right now.
- If you build a house, then there's a red pipe in this room.
- If you think about it, then you'll realize all cats are just holograms.

- if A, then B
- if B, then A
- if X, then G
- A
- Q

So all that's left are **sentences** (in the logician's sense)
and the things that **connect** sentences (like if/then, and, but, or, *etc*.)

Truth-Functional Logic (TFL)

This formal language is named ***Truth-Functional Logic***, or, as it's sometimes called, ***Sentential Logic*** (aka the logic of *sentences*).

Truth-Functional Logic (TFL)

This formal language is named ***Truth-Functional Logic***, or, as it's sometimes called, ***Sentential Logic*** (aka the logic of *sentences*).

Every sentence in this language is made using these building blocks:

-

Truth-Functional Logic (TFL)

This formal language is named ***Truth-Functional Logic***, or, as it's sometimes called, ***Sentential Logic*** (aka the logic of *sentences*).

Every sentence in this language is made using these building blocks:

- **Sentence-Letters**, which stand-in for "atomic sentences" (like A, B, Q, X, etc.)
-

Truth-Functional Logic (TFL)

This formal language is named ***Truth-Functional Logic***, or, as it's sometimes called, ***Sentential Logic*** (aka the logic of *sentences*).

Every sentence in this language is made using these building blocks:

- **Sentence-Letters**, which stand-in for "atomic sentences"
  (like A, B, Q, X, etc.)
- **Sentential Connectives**, which connect to one or more existing sentence(s) to build a new, more complex sentence!
  (*like* **not**, **and**, **or**, **if.. then..**, *and* **if-and-only-if**)

Truth-Functional Logic (TFL)

This formal language is named ***Truth-Functional Logic***, or, as it's sometimes called, ***Sentential Logic*** (aka the logic of *sentences*).

Every sentence in this language is made using these building blocks:

- **Sentence-Letters**, which stand-in for "atomic sentences"
  (like A, B, Q, X, etc.)
- **Sentential Connectives**, which connect to one or more existing sentence(s) to build a new, more complex sentence!
  (*like* **not**, **and**, **or**, **if.. then..**, *and* **if-and-only-if**)

- We'll use symbols to refer to *these* things too, like $\neg$ $\wedge$, $\rightarrow$,and $\leftrightarrow$

Sentential Connectives: Not

The first connective we'll look at is "not..", which we'll write as '¬'

"Not" or "¬" is a sentential **connective** because it connects to a sentence and makes a new sentence (the original's "negation").

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation:

Sentential Connectives: Not

The first connective we'll look at is "not..", which we'll write as '¬'

"Not" or "¬" is a sentential **connective** because it connects to a sentence and makes a new sentence (the original's "negation").

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation:

Sentential Connectives: Not

The first connective we'll look at is "not..", which we'll write as '¬'

"Not" or "¬" is a sentential **connective** because it connects to a sentence and makes a new sentence (the original's "negation").

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation: **It's not the case that** it was raining this morning

It was **not** raining this morning

It isn't true that it was raining this morning

The negation leaves the original sentence's internal parts unchanged. It just *denies* whatever the original sentence *asserted*!

Sentential Connectives: Not

“Not” or “¬” is a sentential **connective** because it connects to a sentence and makes a new sentence (the original’s “negation”).

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation: **It’s not the case that** it was raining this morning

It was **not** raining this morning

It isn’t true that it was raining this morning

How would you **translate** the original sentence and its negation into TFL?

Sentential Connectives: Not

"Not" or "¬" is a sentential **connective** because it connects to a sentence and makes a new sentence (the original's "negation").

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation: **It's not the case that** it was raining this morning

It was **not** raining this morning

It isn't true that it was raining this morning

How would you **translate** the original sentence and its negation into TFL?

Sentence (TFL): It was raining this morning.

Negation (TFL): **It's not the case that** it was raining this morning.

Sentential Connectives: Not

“Not” or “¬” is a sentential **connective** because it connects to a sentence and makes a new sentence (the original’s “negation”).

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation: **It’s not the case that** it was raining this morning

It was **not** raining this morning

It isn’t true that it was raining this morning

How would you **translate** the original sentence and its negation into TFL?

Sentence (TFL): R

Negation (TFL): **It’s not the case that** it was raining this morning.

Sentential Connectives: Not

"Not" or "¬" is a sentential **connective** because it connects to a sentence and makes a new sentence (the original's "negation").

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation: **It's not the case that** it was raining this morning

It was **not** raining this morning

It isn't true that it was raining this morning

How would you **translate** the original sentence and its negation into TFL?

Sentence (TFL): R

Negation (TFL): **It's not the case that** R

Sentential Connectives: Not

"Not" or "¬" is a sentential **connective** because it connects to a sentence and makes a new sentence (the original's "negation").

Quizlet Q2: What would be the **negation** of this sentence?

Sentence: It was raining this morning.

Negation: **It's not the case that** it was raining this morning

It was **not** raining this morning

It isn't true that it was raining this morning

How would you **translate** the original sentence and its negation into TFL?

Sentence (TFL): R

Negation (TFL): $\neg R$

Sentences are Truth-Functions

The connective "not" or '¬' doesn't care about what's going on *inside* of a sentence or sentence-letter. It just is used to say the *opposite*, or *negation*, of whatever the original sentence says.

So, for **any** sentence, $\mathcal{S}$, you can make its negation: $\neg \mathcal{S}$

If "$\mathcal{S}$," is a true statement, then we know that "$\neg \mathcal{S}$," must be false. So putting a '¬' in front of any sentence will *switch* it from true to false or from false to true.

Quizlet Q3: Translate the following three sentences into TFL:

"Zee is happy" "Zee isn't happy" "Zee is unhappy"

Sentences are Truth-Functions

The connective "not" or '¬' doesn't care about what's going on *inside* of a sentence or sentence-letter. It just is used to say the *opposite*, or *negation*, of whatever the original sentence says.

So, for **any** sentence, S, you can make its negation: $\neg S$

If "S," is a true statement, then we know that "$\neg S$," must be false. So putting a '¬' in front of any sentence will *switch* it from true to false or from false to true.

Quizlet Q3: Translate the following three sentences into TFL:

"Zee is happy"	"Zee isn't happy"	"Zee is unhappy"
H	$\neg H$	U

Sentential Connectives: And

The next connective we'll consider is "and", represented as '$\wedge$'.

'$\wedge$' is a ***two-place*** **connective** meaning that it links *two* sentences together to make a bigger sentence, the "**conjunction**" of the two.

Consider these two sentences:

"I'm feeling kind of queasy"   "There's still a mile left in this marathon"

and their **conjunction:**

I'm feeling kind of queasy, **and** there's still a mile left in this marathon

I'm feeling kind of queasy, **but** there's still a mile left in this marathon

I'm feeling kind of queasy; **however**, there's still a mile left in this marathon

Sentences in TFL: Q, M Their Conjunction (in TFL): $Q \wedge M$

Sentences as Truth-Functions:

Again, the connective `∧' doesn't care about what's *inside* the sentences. It's just making a new sentence that says "Yeah, whatever *both* these sentences says is true!"

So whenever "P" and "Q" are **both true**, then "$P \wedge Q$" is true.

But if one or both of them is **false**! Then "$P \wedge Q$" will also be false.

So, when would "I'm hungry ∧ There's no food in this room" be true?
When would it be false?

Sentential Connectives: Or

The next connective we'll consider is "or", represented as '$\vee$'.

'$\vee$' is a ***two-place*** **connective** meaning that it links *two* sentences together to make a bigger sentence, the "**disjunction**" of the two.

Consider these two sentences:

"I'm afraid of geese" "I really can't stand goose honk sounds"

and their **disjunction:**

I'm afraid of geese, **or** I really can't stand goose honk sounds

either I'm afraid of geese, **or** I really can't stand goose honk sounds

I'm afraid of geese, **and/or** I really can't stand goose honk sounds

Sentences in TFL: G, H Their Disjunction (in TFL): $G \vee H$

Sentences as Truth-Functions:

The connective `V' doesn't act quite like you'd expect, given the way that "or" works in natural language.

In logic, the connective 'V' takes two sentences and makes a new one that says "Either *one* or *both* of these sentences is true!". It's what logicians call an "inclusive-'or'

It's like if somebody asked "do you want onions or peppers on your pizza?" and ***you wanted both***, you wouldn't say "Nope!", you'd say "Yes please!".

So whenever either "P", or "Q", or *both* are **true**, then "$P \vee Q$" is true.

But if they're both **false**, then "$P \vee Q$" will also be false.

HOMEWORK:

This will be listed on the website as well:

Chapter-Questions:

1-2 and 1-4, 2-A and 2-B, 3-C and 3-E,

5-B-(just the even questions) and 5-E

These questions are due, before 11:59pm, this coming MONDAY, Feb 3rd