INTRODUCTION TO LOGIC

Lecture 4

The Syntax of Predicate Logic

Dr. James Studd

I counsel you, dear friend, in sum,
That first you take collegium logicum.
Your spirit's then well broken in for you,
In Spanish boots laced tightly to,
That you henceforth may more deliberately keep
The path of thought and straight along it creep,
And not perchance criss-cross may go,
A- will-o'-wisping to and fro.
Then you'll be taught full many a day
What at one stroke you've done alway,
Like eating and like drinking free,
It now must go like: One! Two! Three!
Goethe, Faust I

Outline

1. Introduction to $\mathcal{L}_2$
2. Straightforward predicate formalisation
3. The syntax of $\mathcal{L}_2$

Recall argument 2 from week 1.

Argument 2

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Propositional Formalisation

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Propositional Formalisation

(1) T

Dictionary: T: Zeno is a tortoise.

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Propositional Formalisation

(1) T

(2) A

Dictionary: T: Zeno is a tortoise.
A: All tortoises are toothless.

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Propositional Formalisation

(1) T

(2) A

(C) L

Dictionary: T: Zeno is a tortoise.
A: All tortoises are toothless. L: Zeno is toothless.

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Propositional Formalisation

(1) T

(2) A

(C) L

Dictionary: T: Zeno is a tortoise.
A: All tortoises are toothless. L: Zeno is toothless.

But $T, A \not\models L$.

Recall argument 2 from week 1.

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Is the argument propositionally valid?

Propositional Formalisation **Not Valid**

(1) T

(2) A

(C) L

Dictionary: T: Zeno is a tortoise.
A: All tortoises are toothless. L: Zeno is toothless.

But $T, A \not\models L$.

Subject-Predicate form.

To capture the validity of Argument 2 we need a richer formal language:

Subject-Predicate form.

To capture the validity of Argument 2 we need a richer formal language:
One that can capture some of the internal structure of sentences like ‘Zeno is a tortoise’ and ‘All tortoises are toothless’.

Subject-Predicate form.

To capture the validity of Argument 2 we need a richer formal language:
One that can capture some of the internal structure of sentences like ‘Zeno is a tortoise’ and ‘All tortoises are toothless’.

$$\underbrace{\text{Zeno}}_{\text{Designator}}\ \underbrace{\text{is a tortoise}}_{\text{Predicate}}$$

Subject-Predicate form.

To capture the validity of Argument 2 we need a richer formal language:
One that can capture some of the internal structure of sentences like ‘Zeno is a tortoise’ and ‘All tortoises are toothless’.

$$\underbrace{\text{Zeno}}_{\text{Designator}}\ \underbrace{\text{is a tortoise}}_{\text{Predicate}}$$

The language we use is the language of predicate logic: $\mathcal{L}_2$

Designators/Constants

In English, designators are singular noun phrases that (purport to) refer to a single thing.

Designators/Constants

In English, designators are singular noun phrases that (purport to) refer to a single thing.

English designators

- Proper names: ‘Zeno’, ‘London’, ‘Elizabeth II’

Designators/Constants

In English, designators are singular noun phrases that (purport to) refer to a single thing.

English designators

- Proper names: ‘Zeno’, ‘London’, ‘Elizabeth II’
- Definite descriptions: ‘The capital of France’,‘The Queen’, ‘England’s capital’, ‘Volker’s tortoise’.

Designators/Constants

In English, designators are singular noun phrases that (purport to) refer to a single thing.

English designators

- Proper names: ‘Zeno’, ‘London’, ‘Elizabeth II’
- Definite descriptions: ‘The capital of France’,‘The Queen’, ‘England’s capital’, ‘Volker’s tortoise’.

In $\mathcal{L}_2$, the equivalent of designators are constants

Designators/Constants

In English, designators are singular noun phrases that (purport to) refer to a single thing.

English designators

- Proper names: ‘Zeno’, ‘London’, ‘Elizabeth II’
- Definite descriptions: ‘The capital of France’,‘The Queen’, ‘England’s capital’, ‘Volker’s tortoise’.

In $\mathcal{L}_2$, the equivalent of designators are constants

$\mathcal{L}_2$ constants

- a, b, c, etc.

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- Zeno is a tortoise

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- ____ is a tortoise

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise
- London is bigger than the capital of France

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise
- is bigger than the capital of France

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise
- is bigger than

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise
- is bigger than
- John opened the bottle with his keyring

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- ______ is a tortoise
- ______ is bigger than ______
- ______ opened the bottle with his keyring

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- ______ is a tortoise
- ______ is bigger than ______
- ______ opened ______ with his keyring

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise
- is bigger than
- opened with

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise
- is bigger than
- opened with

Arity

A predicate's arity is the number of occurrences of designators it takes to make a whole sentence again.

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise Arity: 1
- is bigger than
- opened with

Arity

A predicate's arity is the number of occurrences of designators it takes to make a whole sentence again.

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise Arity: 1
- is bigger than Arity: 2
- opened with

Arity

A predicate's arity is the number of occurrences of designators it takes to make a whole sentence again.

Predicates

In English, a predicate is the result of deleting one or more designators from a sentence.

English predicates

- is a tortoise — Arity: 1
- is bigger than — Arity: 2
- opened with — Arity: 3

Arity

A predicate's arity is the number of occurrences of designators it takes to make a whole sentence again.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc
- P^2, P^2_1, R^2_{43}, etc.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc
- P^2, P^2_1, R^2_{43}, etc.
- P^3, P^3_1, R^3_{43}, etc

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc
- P^2, P^2_1, R^2_{43}, etc.
- P^3, P^3_1, R^3_{43}, etc

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc — Arity: 1
- P^2, P^2_1, R^2_{43}, etc.
- P^3, P^3_1, R^3_{43}, etc

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc.
- P^3, P^3_1, R^3_{43}, etc

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. Arity: 2
- P^3, P^3_1, R^3_{43}, etc

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc (3-place, ternary) Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc (3-place, ternary) Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

Combining n occurrences of constants with an n-ary predicate makes an $\mathcal{L}_2$-sentence.

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc (3-place, ternary) Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

Combining n occurrences of constants with an n-ary predicate makes an $\mathcal{L}_2$-sentence.

Some $\mathcal{L}_2$-sentences

Examples: P^1a

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc (3-place, ternary) Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

Combining n occurrences of constants with an n-ary predicate makes an $\mathcal{L}_2$-sentence.

Some $\mathcal{L}_2$-sentences

Examples: P^1a, R^2ab

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc (3-place, ternary) Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

Combining n occurrences of constants with an n-ary predicate makes an $\mathcal{L}_2$-sentence.

Some $\mathcal{L}_2$-sentences

Examples: P^1a, R^2ab, R^2aa

In $\mathcal{L}_2$, we simply stipulate what the predicates are.

$\mathcal{L}_2$-predicates

- P^1, Q^1, R^1, P^1_1, Q^1_{36}, etc (1-place, unary) Arity: 1
- P^2, P^2_1, R^2_{43}, etc. (2-place, binary) Arity: 2
- P^3, P^3_1, R^3_{43}, etc (3-place, ternary) Arity: 3

Arity

The superscript (or 'arity index') tells us what the arity is.

Combining n occurrences of constants with an n-ary predicate makes an $\mathcal{L}_2$-sentence.

Some $\mathcal{L}_2$-sentences

Examples: P^1a, R^2ab, R^2aa, $P^4_{23}abca$

Predicate formalisation

We can now formalise the first premise

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: P^1a

Dictionary: a: Zeno. P^1: $\ldots$is a tortoise.

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: P^1a

Dictionary: a: Zeno. P^1: ...is a tortoise.

More examples **Formalisation**

(1) Alice paid Beatrice.

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: ...is a tortoise.

More examples **Formalisation**

(1) Alice paid Beatrice.

Dictionary: a: Alice. b: Beatrice. P^2: ...paid ....

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: ...is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	$P^2 ab$

Dictionary: a: Alice. b: Beatrice. P^2: ...paid ....

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	$P^2 ab$
(2) Beatrice paid Alice.	

Dictionary: a: Alice. b: Beatrice. P^2: …paid ….

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: P^1a

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	P^2ab
(2) Beatrice paid Alice.	P^2ba

Dictionary: a: Alice. b: Beatrice. P^2: …paid ….

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: P^1a

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	P^2ab
(2) Beatrice paid Alice.	P^2ba
(3) Alice was paid by Beatrice.	

Dictionary: a: Alice. b: Beatrice. P^2: …paid ….

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	$P^2 ab$
(2) Beatrice paid Alice.	$P^2 ba$
(3) Alice was paid by Beatrice. Paraphrase: Beatrice paid Alice.	

Dictionary: a: Alice. b: Beatrice. P^2: …paid ….

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	$P^2 ab$
(2) Beatrice paid Alice.	$P^2 ba$
(3) Alice was paid by Beatrice. Paraphrase: Beatrice paid Alice.	$P^2 ba$

Dictionary: a: Alice. b: Beatrice. P^2: …paid … .

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: $\ldots$is a tortoise.

More examples	**Formalisation**
(1) Alice paid Beatrice.	$P^2 ab$
(2) Beatrice paid Alice.	$P^2 ba$
(3) Alice was paid by Beatrice. Paraphrase: Beatrice paid Alice.	$P^2 ba$
(4) Alice paid Beatrice or was paid by Beatrice	

Dictionary: a: Alice. b: Beatrice. P^2: $\ldots$paid $\ldots$.

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: $P^1 a$

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	$P^2 ab$
(2) Beatrice paid Alice.	$P^2 ba$
(3) Alice was paid by Beatrice. Paraphrase: Beatrice paid Alice.	$P^2 ba$
(4) Alice paid Beatrice or was paid by Beatrice Paraphrase: Alice paid Beatrice or Beatrice paid Alice	

Dictionary: a: Alice. b: Beatrice. P^2: …paid ….

Predicate formalisation

We can now formalise the first premise

Formalise: Zeno is a tortoise

Formalisation: P^1a

Dictionary: a: Zeno. P^1: …is a tortoise.

More examples	Formalisation
(1) Alice paid Beatrice.	P^2ab
(2) Beatrice paid Alice.	P^2ba
(3) Alice was paid by Beatrice. Paraphrase: Beatrice paid Alice.	P^2ba
(4) Alice paid Beatrice or was paid by Beatrice Paraphrase: Alice paid Beatrice or Beatrice paid Alice	$P^2ab \vee P^2ba$

Dictionary: a: Alice. b: Beatrice. P^2: …paid ….

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

This cannot be formalised as a designator-predicate sentence.

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

This cannot be formalised as a designator-predicate sentence. To formalise this sentence we additionally need to deal with quantifiers and pronouns.

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

This cannot be formalised as a designator-predicate sentence. To formalise this sentence we additionally need to deal with quantifiers and pronouns.

$\underbrace{\text{Everything is such that}}_{\text{Quantifier}}\ \underbrace{\text{it}}_{\text{Pronoun}}\ \underbrace{\text{has mass}}_{\text{Predicate}}$

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

This cannot be formalised as a designator-predicate sentence. To formalise this sentence we additionally need to deal with quantifiers and pronouns.

$$\underbrace{\text{Everything is such that}}_{\text{Quantifier}}\ \underbrace{\text{it}}_{\text{Pronoun}}\ \underbrace{\text{has mass}}_{\text{Predicate}}$$

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

This cannot be formalised as a designator-predicate sentence. To formalise this sentence we additionally need to deal with quantifiers and pronouns.

$$\underbrace{\text{Everything is such that}}_{\text{Quantifier}}\ \underbrace{\text{it}}_{\text{Pronoun}}\ \underbrace{\text{has mass}}_{\text{Predicate}}$$

Argument 2 revisited, again

What about the second premiss?

All tortoises are toothless

This cannot be formalised as a designator-predicate sentence. To formalise this sentence we additionally need to deal with quantifiers and pronouns.

$$\underbrace{\text{Everything is such that}}_{\text{Quantifier}}\ \underbrace{\text{it}}_{\text{Pronoun}}\ \underbrace{\text{has mass}}_{\text{Predicate}}$$

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

‘she’, ‘him’, ‘it’

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and she likes him.

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and she likes him.

The pronouns each refer to a particular person

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and she likes him.

The pronouns each refer to a particular person

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes him.

The pronouns each refer to a particular person

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Quantificational use

If a girl likes Tom, Tom likes her.

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Quantificational use

If a girl likes Tom, Tom likes her.

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Quantificational use

If a girl likes Tom, Tom likes her.

The pronoun cannot be replaced by the noun to which it refers back (without changing the meaning).

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Quantificational use

If a girl likes Tom, Tom likes a girl.

The pronoun cannot be replaced by the noun to which it refers back (without changing the meaning).

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Quantificational use

If a girl likes Tom, Tom likes her.

The pronoun cannot be replaced by the noun to which it refers back (without changing the meaning).

Pronouns/Variables

In English, pronouns are expressions used instead of nouns

English pronouns

'she', 'him', 'it'

Distinguish two uses of pronouns: lazy and quantificational.

Lazy use

Tom likes Miranda Fitzwilliam-Carter and Miranda Fitzwilliam-Carter likes Tom.

The pronouns each refer to a particular person

Quantificational use

If a girl likes Tom, Tom likes her.

The pronoun cannot be replaced by the noun to which it refers back (without changing the meaning).
The pronoun 'her' does not refer to a single person: it is used to make a general claim.

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

‘Part’ expresses a transitive relation

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

‘Part’ expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

‘Part’ expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

We could instead express this as:

If an object x_1 is part of another object x_2 and x_2 is part of yet another object x_3, then x_1 is part of x_3.

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

‘Part’ expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

We could instead express this as:

If an object x_1 is part of another object x_2 and x_2 is part of yet another object x_3, then x_1 is part of x_3.

NB: not idiomatic English.

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

‘Part’ expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

We could instead express this as:

If an object x_1 is part of another object x_2 and x_2 is part of yet another object x_3, then x_1 is part of x_3.

NB: not idiomatic English.
In $\mathcal{L}_2$, the equivalent of pronouns are variables.

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

'Part' expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

We could instead express this as:

If an object x_1 is part of another object x_2 and x_2 is part of yet another object x_3, then x_1 is part of x_3.

NB: not idiomatic English.
In $\mathcal{L}_2$, the equivalent of pronouns are variables.

$\mathcal{L}_2$ variables

- x, y, z,

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

'Part' expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

We could instead express this as:

If an object x_1 is part of another object x_2 and x_2 is part of yet another object x_3, then x_1 is part of x_3.

NB: not idiomatic English.
In $\mathcal{L}_2$, the equivalent of pronouns are variables.

$\mathcal{L}_2$ variables

- x, y, z, x_1, y_1, z_1,

In English, we rely on contextual clues to keep track of what pronouns and other expressions are referring back to.

‘Part’ expresses a transitive relation

If an object is part of another object and this object is part of yet another object, then it is part of that object.

We could instead express this as:

If an object x_1 is part of another object x_2 and x_2 is part of yet another object x_3, then x_1 is part of x_3.

NB: not idiomatic English.
In $\mathcal{L}_2$, the equivalent of pronouns are variables.

$\mathcal{L}_2$ variables

- $x, y, z, x_1, y_1, z_1, x_2, \ldots$

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that'

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) ‘Everything is such that’, ‘Something is such that’

(2) ‘Every philosopher’

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) ‘Everything is such that’, ‘Something is such that’

(2) ‘Every philosopher’, ‘All tortoises’

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

(2) 'Every philosopher', 'All tortoises', 'No people'

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

(2) 'Every philosopher', 'All tortoises', 'No people'

(3) 'Most people'

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) ‘Everything is such that’, ‘Something is such that’

(2) ‘Every philosopher’, ‘All tortoises’, ‘No people’

(3) ‘Most people’, ‘Over half of all tortoises’

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

(2) 'Every philosopher', 'All tortoises', 'No people'

(3) 'Most people', 'Over half of all tortoises'

The $\mathcal{L}_2$-quantifiers correspond to the quantifiers in (1).

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) ‘Everything is such that’, ‘Something is such that’

(2) ‘Every philosopher’, ‘All tortoises’, ‘No people’

(3) ‘Most people’, ‘Over half of all tortoises’

The $\mathcal{L}_2$-quantifiers correspond to the quantifiers in (1).

$\mathcal{L}_2$-quantifiers

(1) $\forall x$

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

(2) 'Every philosopher', 'All tortoises', 'No people'

(3) 'Most people', 'Over half of all tortoises'

The $\mathcal{L}_2$-quantifiers correspond to the quantifiers in (1).

$\mathcal{L}_2$-quantifiers

(1) $\forall x$, $\exists x$

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) ‘Everything is such that’, ‘Something is such that’

(2) ‘Every philosopher’, ‘All tortoises’, ‘No people’

(3) ‘Most people’, ‘Over half of all tortoises’

The $\mathcal{L}_2$-quantifiers correspond to the quantifiers in (1).

$\mathcal{L}_2$-quantifiers

(1) $\forall x$, $\exists x$; also: $\exists y$, $\exists z_{34}$, etc.

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

(2) 'Every philosopher', 'All tortoises', 'No people'

(3) 'Most people', 'Over half of all tortoises'

The $\mathcal{L}_2$-quantifiers correspond to the quantifiers in (1).

$\mathcal{L}_2$-quantifiers

(1) $\forall x$, $\exists x$; also: $\exists y$, $\exists z_{34}$, etc.

$\mathcal{L}_2$ can formalise the quantifiers in (2) via paraphrase.

Quantifiers

In English, quantifiers are noun phrases typically used to make general claims

English quantifiers

(1) 'Everything is such that', 'Something is such that'

(2) 'Every philosopher', 'All tortoises', 'No people'

(3) 'Most people', 'Over half of all tortoises'

The $\mathcal{L}_2$-quantifiers correspond to the quantifiers in (1).

$\mathcal{L}_2$-quantifiers

(1) $\forall x$, $\exists x$; also: $\exists y$, $\exists z_{34}$, etc.

$\mathcal{L}_2$ can formalise the quantifiers in (2) via paraphrase.
$\mathcal{L}_2$ cannot formalise the quantifiers in (3).

Formalising general sentences

Formalise

Everything is such that it has mass

Formalising general sentences

Formalise

Everything is such that it has mass

Formalising general sentences

Formalise

Everything is such that it has mass

$\forall x$ corresponds to 'Everything is such that'

Formalising general sentences

Formalise

Everything is such that it has mass

$\forall x$ corresponds to ‘Everything is such that’

x corresponds to ‘it’

Formalising general sentences

Formalise

Everything is such that it has mass

$\forall x$ corresponds to ‘Everything is such that’

x corresponds to ‘it’

Let M correspond to ‘has mass’

Formalising general sentences

Formalise

Everything is such that it has mass

$\forall x$ corresponds to ‘Everything is such that’

x corresponds to ‘it’

Let M correspond to ‘has mass’

Then Mx corresponds to ‘it has mass’

Formalising general sentences

Formalise

$$\underbrace{\text{Everything is such that}}_{\forall x}\ \underbrace{\text{it}}_{x}\ \underbrace{\text{has mass}}_{M}$$

$\forall x$ corresponds to 'Everything is such that'
x corresponds to 'it'
Let M correspond to 'has mass'
Then Mx corresponds to 'it has mass'

Formalisation: $\forall x Mx$

Formalising general sentences

Formalise

$\underbrace{\text{Everything is such that}}_{\forall x}\ \underbrace{\text{it}}_{x}\ \underbrace{\text{has mass}}_{M}$

$\forall x$ corresponds to ‘Everything is such that’
x corresponds to ‘it’
Let M correspond to ‘has mass’
Then Mx corresponds to ‘it has mass’

Formalisation: $\forall x Mx$
Dictionary: M: …has mass.

Formalising general sentences

Formalise

$\underbrace{\text{Everything is such that}}_{\forall x}\ \underbrace{\text{it}}_{x}\ \underbrace{\text{has mass}}_{M}$

$\forall x$ corresponds to 'Everything is such that'
x corresponds to 'it'
Let M correspond to 'has mass'
Then Mx corresponds to 'it has mass'

Formalisation: $\forall x Mx$
Dictionary: M: …has mass.

Note: it's fine to omit the arity index when formalising.

More examples

(1) Everything has mass.

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

More examples

(1) Everything has mass.

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

Paraphrases.

(1) Everything is such that: it has mass.

More examples

(1) Everything has mass.

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

More examples — Formalisation

(1) Everything has mass. $\forall x Mx$

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

More examples — Formalisation

(1) Everything has mass. $\forall x Mx$

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

Dictionary: M:... has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

More examples — Formalisation

(1) Everything has mass. $\forall x Mx$

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

Dictionary: M:…has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

More examples — Formalisation

(1) Everything has mass. $\forall x Mx$

(2) Something has mass.

(3) Some person has mass.

(4) Every person has mass.

Dictionary: M:... has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

More examples — Formalisation

(1)	Everything has mass.	$\forall x M x$
(2)	Something has mass.	$\exists x M x$
(3)	Some person has mass.	
(4)	Every person has mass.	

Dictionary: M:. . . has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

More examples — Formalisation

(1) Everything has mass. $\forall x Mx$

(2) Something has mass. $\exists x Mx$

(3) Some person has mass.

(4) Every person has mass.

Dictionary: M:...has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	
(4)	Every person has mass.	

Dictionary: M:...has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	
(4)	Every person has mass.	

Dictionary: M:…has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	
(4)	Every person has mass.	

Dictionary: M:... has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	$\exists x(Px \wedge Mx)$
(4)	Every person has mass.	

Dictionary: M:. . . has mass.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	$\exists x(Px \land Mx)$
(4)	Every person has mass.	

Dictionary: M:…has mass. P …is a person.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	$\exists x(Px \wedge Mx)$
(4)	Every person has mass.	

Dictionary: M:. . . has mass. P . . . is a person.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	$\exists x(Px \land Mx)$
(4)	Every person has mass.	

Dictionary: M:…has mass. P …is a person.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

(4) Everything that is a person has mass.

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	$\exists x (Px \wedge Mx)$
(4)	Every person has mass.	

Dictionary: M:...has mass. P ...is a person.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

(4) Everything that is a person has mass.
Every x is such that (if x is a person, then x has mass).

More examples — Formalisation

(1)	Everything has mass.	$\forall x Mx$
(2)	Something has mass.	$\exists x Mx$
(3)	Some person has mass.	$\exists x(Px \wedge Mx)$
(4)	Every person has mass.	$\forall x(Px \to Mx)$

Dictionary: M:...has mass. P ...is a person.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

(4) Everything that is a person has mass.
Every x is such that (if x is a person, then x has mass).

More examples	Formalisation
(1) Everything has mass.	$\forall x Mx$
(2) Something has mass.	$\exists x Mx$
(3) Some person has mass.	$\exists x(Px \wedge Mx)$
(4) Every person has mass.	$\forall x(Px \rightarrow Mx)$

Dictionary: M:. . . has mass. P . . . is a person.

Paraphrases.

(1) Everything is such that: it has mass.
Every x is such that: x has mass.

(2) Some x is such that: x has mass.

(3) Something that is a person has mass.
Something is such that (it is a person and it has mass).
Some x is such that (x is a person and x has mass).

(4) Everything that is a person has mass.
Every x is such that (if x is a person, then x has mass).

Common variants

$\mathcal{L}_2$	standard connective	some other formulations
$\forall x$	everything	every, all, any, a
$\exists x$	something	some, at least one, any, a

Common variants

$\mathcal{L}_2$	standard connective	some other formulations
$\forall x$	everything	every, all, any, a
$\exists x$	something	some, at least one, any, a

Common variants

$\mathcal{L}_2$	standard connective	some other formulations
$\forall x$	everything	every, all, any, a
$\exists x$	something	some, at least one, any, a

Common variants

$\mathcal{L}_2$	standard connective	some other formulations
$\forall x$	everything	every, all, any, a
$\exists x$	something	some, at least one, any, a

Examples

(1) A dog barked.

(2) A dog barks.

Dictionary. D: ...is a dog. B_1: ...barked. B_2: ...barks

Examples

(1) A dog barked.

(2) A dog barks.

Dictionary. D: …is a dog. B_1: …barked. B_2: …barks

Paraphrase (of the most natural readings)

(1) Some dog barked

Examples	Formalisations
(1) A dog barked.	$\exists x(Dx \wedge B_1 x)$
(2) A dog barks.	

Dictionary. D: ...is a dog. B_1: ...barked. B_2: ...barks

Paraphrase (of the most natural readings)

(1) Some dog barked
Some x is such that (x is a dog and x barked).

Examples | **Formalisations**

(1) A dog barked. $\exists x(Dx \wedge B_1 x)$

(2) A dog barks.

Dictionary. D: …is a dog. B_1: …barked. B_2: …barks

Paraphrase (of the most natural readings)

(1) Some dog barked
Some x is such that (x is a dog and x barked).

(2) Every dog barks

Examples	Formalisations
(1) A dog barked.	$\exists x(Dx \wedge B_1 x)$
(2) A dog barks.	$\forall x(Dx \rightarrow B_2 x)$

Dictionary. D: …is a dog. B_1: …barked. B_2: …barks

Paraphrase (of the most natural readings)

(1) Some dog barked
Some x is such that (x is a dog and x barked).

(2) Every dog barks
Every x is such that (if x is dog, then x barks).

Quantifiers at the end of sentences

Examples

(1) Everyone loves Zuleika

(2) Zuleika loves everyone

Quantifiers at the end of sentences

Examples

(1) Everyone loves Zuleika

(2) Zuleika loves everyone

Dictionary: a: Zuleika. P: …is a person. L: …loves ….

Quantifiers at the end of sentences

Examples

(1) Everyone loves Zuleika

(2) Zuleika loves everyone

Dictionary: a: Zuleika. P: ...is a person. L: ...loves ....

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

Quantifiers at the end of sentences

Examples	Formalisation
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	

Dictionary: a: Zuleika. P: …is a person. L: …loves ….

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

Quantifiers at the end of sentences

Examples	**Formalisation**
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	

Dictionary: a: Zuleika. P: ... is a person. L: ... loves ....

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

Quantifiers at the end of sentences

Examples	**Formalisation**
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	

Dictionary: a: Zuleika. P: …is a person. L: …loves ….

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika

Quantifiers at the end of sentences

Examples | **Formalisation**

(1) Everyone loves Zuleika $\forall x(Px \to Lxa)$

(2) Zuleika loves everyone

Dictionary: a: Zuleika. P: ...is a person. L: ...loves ....

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika
Every x is such that
(if x is a person, then x is loved by Zuleika)

Quantifiers at the end of sentences

Examples	Formalisation
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	

Dictionary: a: Zuleika. P: ... is a person. L: ... loves ....

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika
Every x is such that
(if x is a person, then x is loved by Zuleika)

Quantifiers at the end of sentences

Examples	Formalisation
(1) Everyone loves Zuleika	$\forall x(Px \rightarrow Lxa)$
(2) Zuleika loves everyone	

Dictionary: a: Zuleika. P: . . . is a person. L: . . . loves . . . .

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika
Every x is such that
(if x is a person, then x is loved by Zuleika)
Every x is such that
(if x is a person, then Zuleika loves x)

Quantifiers at the end of sentences

Examples	**Formalisation**
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	

Dictionary: a: Zuleika. P: …is a person. L: …loves ….

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika
Every x is such that
(if x is a person, then x is loved by Zuleika)
Every x is such that
(if x is a person, then Zuleika loves x)

Quantifiers at the end of sentences

Examples	Formalisation
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	$\forall x(Px \to Lax)$

Dictionary: a: Zuleika. P: ... is a person. L: ... loves ....

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika
Every x is such that
(if x is a person, then x is loved by Zuleika)
Every x is such that
(if x is a person, then Zuleika loves x)

Quantifiers at the end of sentences

Examples	**Formalisation**
(1) Everyone loves Zuleika	$\forall x(Px \to Lxa)$
(2) Zuleika loves everyone	$\forall x(Px \to Lax)$

Dictionary: a: Zuleika. P: ...is a person. L: ...loves ....

Paraphrases

(1) Every x is such that
(if x is a person, then x loves Zuleika)

(2) Everyone is loved by Zuleika
Every x is such that
(if x is a person, then x is loved by Zuleika)
Every x is such that
(if x is a person, then Zuleika loves x)

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

Dictionary: P: …is a person. L: …loves ….

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

Dictionary: P: ... is a person. L: ... loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

Dictionary: P: ...is a person. L: ...loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

Dictionary: P: ...is a person. L: ...loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)

Some x is such that (x is a person and everyone is loved by x)

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

Dictionary: P: ...is a person. L: ...loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)

Some x is such that (x is a person and everyone is loved by x)

Some x is such that

(x is a person and everything is such that:

(if it is a person, then it is loved by x))

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone
Dictionary: P: ... is a person. L: ... loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)
Some x is such that (x is a person and everyone is loved by x)
Some x is such that
(x is a person and everything is such that:
(if it is a person, then it is loved by x))
Some x is such that
(x is a person and every y is such that:
(if y is a person, then y is loved by x))

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone
Dictionary: P: ...is a person. L: ...loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)
Some x is such that (x is a person and everyone is loved by x)
Some x is such that
(x is a person and everything is such that:
(if it is a person, then it is loved by x))
Some x is such that
(x is a person and every y is such that:
(if y is a person, then y is loved by x))

We use the same trick to deal with multiple quantifiers.

Example

Someone loves everyone

Dictionary: P: ... is a person. L: ... loves ... .

Paraphrases

Some x is such that (x is a person and x loves everyone)

Some x is such that (x is a person and everyone is loved by x)

Some x is such that
    (x is a person and everything is such that:
        (if it is a person, then it is loved by x))

Some x is such that
    (x is a person and every y is such that:
        (if y is a person, then y is loved by x))

Some x is such that
    (x is a person and every y is such that:
        (if y is a person, then x loves y))

We use the same trick to deal with multiple quantifiers.

Example **Formalisation**

Someone loves everyone $\exists x(Px \wedge \forall y(Py \to Lxy))$

Dictionary: P: ...is a person. L: ...loves ....

Paraphrases

Some x is such that (x is a person and x loves everyone)

Some x is such that (x is a person and everyone is loved by x)

Some x is such that
(x is a person and everything is such that:
(if it is a person, then it is loved by x))

Some x is such that
(x is a person and every y is such that:
(if y is a person, then y is loved by x))

Some x is such that
(x is a person and every y is such that:
(if y is a person, then x loves y))

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$ 40

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$ 40

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', …

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

$$P^1 \quad Q^1 \quad R^1 \quad P^1_1 \quad Q^1_1 \quad R^1_1 \quad P^1_2 \quad \ldots$$

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P_n^k, Q_n^k, or R_n^k are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

P^1 Q^1 R^1 P_1^1 Q_1^1 R_1^1 P_2^1 $\ldots$ Unary predicates

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', …

P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	…	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	…	

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	...	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	...	Binary predicates

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

P	Q	R	P_1	Q_1	R_1	P_2	$\dots$	Sentence letters
P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	$\dots$	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	$\dots$	Binary predicates
$\vdots$								

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', …

P	Q	R	P_1	Q_1	R_1	P_2	$\ldots$	Sentence letters
P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	$\ldots$	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	$\ldots$	Binary predicates
$\vdots$								

Definition

The following are variables: x, y, z

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$ 40

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

$$\begin{array}{llllllll l}
P & Q & R & P_1 & Q_1 & R_1 & P_2 & \dots & \text{Sentence letters} \\
P^1 & Q^1 & R^1 & P^1_1 & Q^1_1 & R^1_1 & P^1_2 & \dots & \text{Unary predicates} \\
P^2 & Q^2 & R^2 & P^2_1 & Q^2_1 & R^2_1 & P^2_2 & \dots & \text{Binary predicates} \\
\vdots & & & & & & & &
\end{array}$$

Definition

The following are variables: x, y, z, x_1, y_1, z_1

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

P	Q	R	P_1	Q_1	R_1	P_2	$\dots$	Sentence letters
P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	$\dots$	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	$\dots$	Binary predicates
$\vdots$								

Definition

The following are variables: x, y, z, x_1, y_1, z_1, x_2, ...

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', …

P	Q	R	P_1	Q_1	R_1	P_2	…	Sentence letters
P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	…	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	…	Binary predicates
$\vdots$								

Definition

The following are variables: x, y, z, x_1, y_1, z_1, x_2, …
The following are constants: a, b, c

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

P	Q	R	P_1	Q_1	R_1	P_2	$\ldots$	Sentence letters
P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	$\ldots$	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	$\ldots$	Binary predicates
$\vdots$								

Definition

The following are variables: x, y, z, x_1, y_1, z_1, x_2, ...

The following are constants: a, b, c, a_1, b_1, c_1

The syntax of $\mathcal{L}_2$

Here's the official syntax of $\mathcal{L}_2$

Definition (Predicate letters)

P^k_n, Q^k_n, or R^k_n are predicate letters, where k and n are either missing (no symbol) or a numeral '1', '2', '3', ...

P	Q	R	P_1	Q_1	R_1	P_2	$\ldots$	Sentence letters
P^1	Q^1	R^1	P^1_1	Q^1_1	R^1_1	P^1_2	$\ldots$	Unary predicates
P^2	Q^2	R^2	P^2_1	Q^2_1	R^2_1	P^2_2	$\ldots$	Binary predicates
$\vdots$								

Definition

The following are variables: x, y, z, x_1, y_1, z_1, x_2, ...

The following are constants: a, b, c, a_1, b_1, c_1, a_2, ...

Definition (atomic formulae of $\mathcal{L}_2$)

If Z is a predicate letter of arity n and each of $t_1, \ldots, t_n$ is a variable or a constant, then $Zt_1 \ldots t_n$ is an atomic formula of $\mathcal{L}_2$.

Definition (atomic formulae of $\mathcal{L}_2$)

If Z is a predicate letter of arity n and each of $t_1, \ldots, t_n$ is a variable or a constant, then $Zt_1 \ldots t_n$ is an atomic formula of $\mathcal{L}_2$.

Definition (formulae of $\mathcal{L}_2$)

1. All atomic formulae of $\mathcal{L}_2$ are formulae of $\mathcal{L}_2$.

Definition (atomic formulae of $\mathcal{L}_2$)

If Z is a predicate letter of arity n and each of $t_1, \dots, t_n$ is a variable or a constant, then $Zt_1 \dots t_n$ is an atomic formula of $\mathcal{L}_2$.

Definition (formulae of $\mathcal{L}_2$)

1. All atomic formulae of $\mathcal{L}_2$ are formulae of $\mathcal{L}_2$.
2. If ϕ and ψ are formulae of $\mathcal{L}_2$, then $\neg\phi$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \to \psi)$ and $(\phi \leftrightarrow \psi)$ are formulae of $\mathcal{L}_2$.

Definition (atomic formulae of $\mathcal{L}_2$)

If Z is a predicate letter of arity n and each of $t_1, \ldots, t_n$ is a variable or a constant, then $Zt_1 \ldots t_n$ is an atomic formula of $\mathcal{L}_2$.

Definition (formulae of $\mathcal{L}_2$)

1. All atomic formulae of $\mathcal{L}_2$ are formulae of $\mathcal{L}_2$.
2. If ϕ and ψ are formulae of $\mathcal{L}_2$, then $\neg\phi$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$ and $(\phi \leftrightarrow \psi)$ are formulae of $\mathcal{L}_2$.
3. If v is a variable and ϕ is a formula, then $\forall v\, \phi$ and $\exists v\, \phi$ are formulae of $\mathcal{L}_2$.

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa

(ii) Q^3axy

(iii) P

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy

(iii) P

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy

(iii) P

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy Yes

(iii) P

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy Yes

(iii) P

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy Yes

(iii) P Yes

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy Yes

(iii) P Yes

(iv) $\forall v\phi$

(No bracketing conventions, etc. have been applied.)

Worked example: which of these are $\mathcal{L}_2$-formulae?

(i) Pa No

(ii) Q^3axy Yes

(iii) P Yes

(iv) $\forall v\phi$ No

(No bracketing conventions, etc. have been applied.)

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- $P^1 x$ is not a sentence.

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- P^1x is not a sentence. No quantifier binds the variable.

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- P^1x is not a sentence. No quantifier binds the variable.
- $\forall x P^1 x$ is a sentence:

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- $P^1 x$ is not a sentence. No quantifier binds the variable.
- $\forall x P^1 x$ is a sentence: x is bound by $\forall x$.

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- P^1x is not a sentence. No quantifier binds the variable.
- $\forall x P^1 x$ is a sentence: x is bound by $\forall x$.
- $\forall y P^1 x$ is not a sentence:

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- P^1x is not a sentence. No quantifier binds the variable.
- $\forall x P^1 x$ is a sentence: x is bound by $\forall x$.
- $\forall y P^1 x$ is not a sentence: x is not bound by $\forall y$

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- P^1x is not a sentence. No quantifier binds the variable.
- $\forall x P^1 x$ is a sentence: x is bound by $\forall x$.
- $\forall y P^1 x$ is not a sentence: x is not bound by $\forall y$
- $(\forall x P^1 x \vee P^1 x)$ is not a sentence:

Formulae vs Sentences

An $\mathcal{L}_2$ formula ϕ is a sentence if every occurrence of each variable v in ϕ is 'bound' by a quantifier $\exists v$ or $\forall v$.

Examples

- $P^1 x$ is not a sentence. No quantifier binds the variable.
- $\forall x P^1 x$ is a sentence: x is bound by $\forall x$.
- $\forall y P^1 x$ is not a sentence: x is not bound by $\forall y$
- $(\forall x P^1 x \vee P^1 x)$ is not a sentence: only the first occurrence is bound by $\forall x$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 xy)$

Bound variables

- An occurrence of a variable v in ϕ is bound if it occurs within the scope of $\forall v$ or $\exists v$; otherwise it is free.
- The scope of an occurrence of a quantifier in ϕ is the smallest subformula of ϕ that contains this occurrence.

Underline the free variables in the following:

$\forall x(\forall y(P^1 x \to P^1 y) \to R^2 x \underline{y})$

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.
(We've been applying them already.)

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.
(We've been applying them already.)

- The bracketing conventions from $\mathcal{L}_1$ may be applied to drop brackets.

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.
(We've been applying them already.)

- The bracketing conventions from $\mathcal{L}_1$ may be applied to drop brackets.
- Arity indices may be dropped.

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.
(We've been applying them already.)

- The bracketing conventions from $\mathcal{L}_1$ may be applied to drop brackets.
- Arity indices may be dropped.
  e.g Pxy is used to abbreviate P^2xy

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.
(We've been applying them already.)

- The bracketing conventions from $\mathcal{L}_1$ may be applied to drop brackets.
- Arity indices may be dropped.
  e.g Pxy is used to abbreviate P^2xy
- Letters other than the official P, Q, R and a, b, c may be used in formalisations.

Notational conventions

These aren't part of the official syntax of $\mathcal{L}_2$.
But it is permitted to apply the following conventions to make it easier to read $\mathcal{L}_2$-formulae.
(We've been applying them already.)

- The bracketing conventions from $\mathcal{L}_1$ may be applied to drop brackets.
- Arity indices may be dropped.
  e.g Pxy is used to abbreviate P^2xy
- Letters other than the official P, Q, R and a, b, c may be used in formalisations.
- But you should stick to letters early in the alphabet for constants and letters late in the alphabet (start with: x, y, z) for variables.

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$
(C) La

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$
(C) La

Dictionary: a: Zeno. T:...is a tortoise. L: ...is toothless

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$
(C) La

Dictionary: a: Zeno. T:. . . is a tortoise. L: . . . is toothless

What is it for this $\mathcal{L}_2$-argument to be valid?

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$
(C) La

Dictionary: a: Zeno. T:. . . is a tortoise. L: . . . is toothless

What is it for this $\mathcal{L}_2$-argument to be valid? (semantics: week 5)

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$
(C) La

Dictionary: a: Zeno. T:. . . is a tortoise. L: . . . is toothless

What is it for this $\mathcal{L}_2$-argument to be valid? (semantics: week 5)
How can we show that it is valid?

What of argument 2?

Argument 2 **Valid**

(1) Zeno is a tortoise.
(2) All tortoises are toothless.
Therefore, (C) Zeno is toothless.

Formalisation

(1) Ta
(2) $\forall x(Tx \to Lx)$
(C) La

Dictionary: a: Zeno. T:...is a tortoise. L: ...is toothless

What is it for this $\mathcal{L}_2$-argument to be valid? (semantics: week 5)
How can we show that it is valid? (natural deduction: week 6)

http://logicmanual.philosophy.ox.ac.uk