**PH126 Logic I** Fast Lecture 6
Lecturer: s.butterfill@warwick.ac.uk

## Quantifier equivalences

P→Q ⫤⊨ ¬Q→¬P
∀x ( Square(x) → Broken(x) )
⫤⊨ ∀x (¬Broken(x) → ¬Square(x) )

P→Q ⫤⊨ ¬P∨Q
∀x ( Square(x) → Broken(x) )
⫤⊨ ∀x (¬Square(x) ∨ Broken(x) )

¬P∨Q ⫤⊨ ¬(P∧ ¬Q)
∃x ¬F(x) ⫤⊨ ¬∀x F(x)
∀x (¬Square(x) ∨ Broken(x) )
⫤⊨ ∃¬x ( Square(x) ∧ ¬Broken(x) )

∃x ¬Created(x) ⫤⊨ ¬∀x Created(x)

### *Hybrid example*

| | |
|---|---|
| | Every object is either non-square or broken<br>**∀x (¬Square(x) ∨ Broken(x) )** |
| ¬¬P ⫤⊨ P | ⫤⊨ ¬¬∀x (¬Square(x) ∨ Broken(x) ) |
| ¬∀x F(x) ⫤⊨ ∃x¬F(x) | ⫤⊨ ¬∃x ¬(¬Square(x) ∨ Broken(x) ) |
| ¬(P∨Q) ⫤⊨ ¬P∧¬Q | ⫤⊨ ¬∃x (¬¬Square(x) ∧ ¬Broken(x) ) |
| ¬¬P ⫤⊨ P | ⫤⊨ Nothing is square and non-broken<br>**¬∃x ( Square(x) ∧ ¬Broken(x) )** |

## Truth-functional completeness (LPL §7.4)

How many truth-functions can be constructed using 2 sentence letters?

| P | Q | P % Q |
|---|---|---|
| T | T | ? |
| T | F | ? |
| F | T | ? |
| F | F | ? |

Each of these can be T or F

Df. 'A set of truth-functors is said to be *expressively adequate* (or sometimes *functionally complete*) iff, for every truth-function whatever, there is a formula containing only those truth-functors which express that truth-function, i.e. which has as its truth-table the truth-table specifying that function.' (Bostock, *Intermediate Logic* p. 45)

Illustration of the proof that {¬, ∧, ∨} is truth-functionally complete:

| P | Q | P → Q | |
|---|---|---|---|
| T | T | T | [P∧Q] ∨ |
| T | F | F | |
| F | T | T | [¬P∧Q] ∨ |
| F | F | T | [¬P∧ ¬Q] |

[P∧Q] ∨ [¬P∧Q] ∨ [¬P∧ ¬Q]

*Exercise* assuming {¬,∨,∧} is truth-functionally complete, show that {¬,∨} is.

## Variables

Names : a, b, c, …

Variables : x, y, z, w, …

Variables are for saying several things about one thing even without specifying which thing it is

Notation:
**A** ⊢ B means there is a proof of B using premises **A**
⊢ B means there is a proof of B using no premises
**A** ⊢ means the same as A ⊢ ⊥

**A** ⊨ B means B is a logical consequence of A
⊨ B means B is a tautology
**A** ⊨ means the same as **A** ⊨ ⊥
**A** $\models_{TT}$ B means B is a logical consequence of A *just in virtue of the meanings of truth-functions* (the textbook LPL calls this 'tautological consequence')

Soundness: If **A** ⊢ B then **A** ⊨ B
I.e. if you can prove it in Fitch, it's valid

Completeness: If **A** $\models_{TT}$ B then **A** ⊢ B
I.e. if it's valid just in virtue of the meanings of truth functions, you can prove it in Fitch

**Proofs about proofs**

1. If A ├ B then ├ A→B

*Proof* Given a proof for A ├ B …

| | | |
|---|---|---|
| 1 | A | |
| | … | |
| n | B | |

… we can turn it into a proof for ├ A→B:

| | | | |
|---|---|---|---|
| 1 | | A | |
| | | … | |
| n | | B | |
| | A→B | | →Intro: 1-n |

2. If ├ A→B then A ├ B

3. If A ├ B then A ├ ¬¬B

*Proof*:

This is the original proof of A ├ B

Adding this turns it into a proof of A ├ ¬¬B

| | | | |
|---|---|---|---|
| 1 | A | | |
| 2 | … | | |
| n | B | | |
| p | | ¬B | |
| q | | ⊥ | ⊥Intro:n,p |
| | ¬¬B | | ¬Intro:p-q |

4. If A ├ C then A ├ B→C

5. If A ├ B and A ├ ¬C then A ├ ¬(B→C)

**There Is Exactly One**

There is one creator:

$\exists x\ Creator(x)$

Brian is the one and only creator:

$Creator(b) \land \forall x(\ Creator(x) \rightarrow x{=}b\ )$

There is one and only one creator:

$\exists y(\ Creator(y)$
$\land \forall x(\ Creator(x) \rightarrow x{=}y\ )\ )$

or:

$\exists y(\ \forall x(\ Creator(x) \leftrightarrow x{=}y\ )\ )$

Ex: There is one and only one female creator.

Ex: There is one and only one creator and she is female.

Someone is loved by everyone
**$\exists x \forall y$ Loves(y,x)**

Someone loves everyone
**$\exists x \forall y$ Loves(x,y)**

**"Every time I go to the dentist, somebody dies."**

$\forall t\ ((Time(t) \land ToDentist(a,t)\ ) \rightarrow \exists x\ (\ Person(x) \land DiesAt(x,t)\ )\ )$

**Exercises 07**

*For your seventh seminar*

*Only for fast groups*

11.10 (nb. especially no. 20, which requires you to write a sentence that says there's exactly one large tetrahedron) [Translation, incl. uniqueness]

11.13 (translation to English)

13.43–45 (proofs like ¬∃x to ∀x¬)

13.49–50 (hard quantifier proofs)

14.2–3 (Trans. with numerical quantifiers)

14.10–11 (Proofs about uniqueness)

*Despite what book says, DON'T even THINK about using TAUT CON.*