Predicate Logic

Week 2

Slides and materials based on the courses by
Noortje Venhuizen and Mareike Hartmann

The most certain principle in semantics

"For two sentences A and B, if in some possible situation A is true and B is false, A and B must have different meanings."

(M. Cresswell, 1975)

- Knowing the meaning of a (declarative) sentence requires knowing what the world would have to be like for the sentence to be true:

Meaning = Truth Conditions

- Applied to logical representations:

For sentence A and formula α: If there is a possible situation in which A is true and α is not, or vice versa, then α is ***not*** *an appropriate meaning representation for A.*

A central notion: Entailment

- *“Tina is tall and thin”* ⇒ *“Tina is tall”*
- *“Tina is tall, and Ms. Turner is not tall”* ⇒ *“Tina is not Ms.Turner”*
- *“a dog entered the room”* ⇒ *“an animal entered the room”*
- *“Tweety is a bird”* ⇏ *“Tweety can fly”*

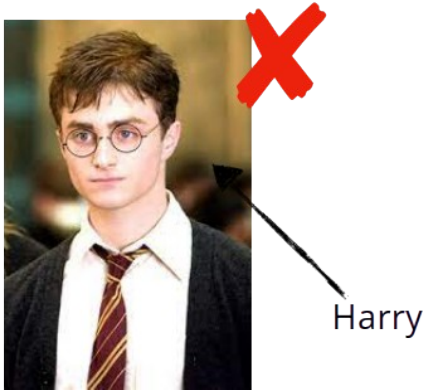
- **Entailment**: speakers intuitively judge S_2 to be true whenever S_1 is true
  - S_1 *entails* S_2 ($S_1 \Rightarrow S_2$)

Truth-conditional formal semantics

- The meaning representation of a sentence must be true in exactly the same situations as the sentence itself

"Harry is a prince" ⟷ prince(harry)

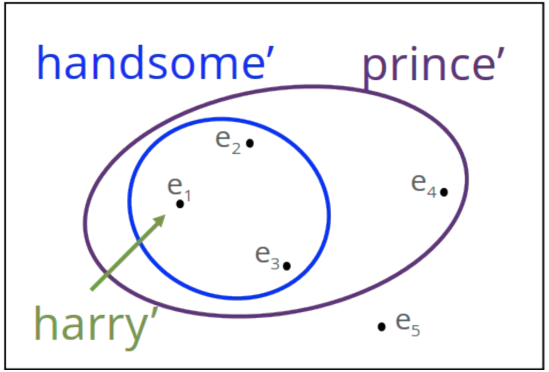
language *logic*



Model structures and formulas

- A **model structure** is a formal representation of a single possible situation
- A **formula** is a statement about model structures in a formal language
  - Formulas (formulae) obtain a **truth value** (true/false) with respect to model structure *M*.

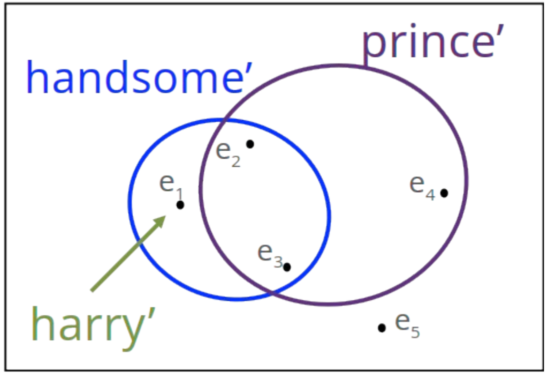
M_1:



✓ handsome′(harry′) ✓

✓ prince′(harry′) ✗

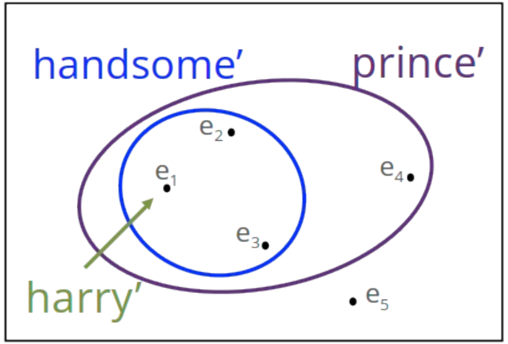
M_2:



Model structures: Definition

- Formally, a model structure *M* can be defined as a tuple *M* = (*U*, *V*), where:
  - *U* is a set of individual **entities**, called the **universe** (sometimes called domain *D*);
  - *V* is an **interpretation function** (sometimes denoted by *I*) that maps formula expressions onto (sets of) these entities.

M_1:



$M_1 = (U_1, V_1)$, where:

$U_1 = \{e_1, e_2, e_3, e_4, e_5\}$

$V_1(\text{handsome'}) = \{e_1, e_2, e_3\}$
$V_1(\text{prince'}) = \{e_1, e_2, e_3, e_4\}$
$V_1(\text{harry'}) = e_1$

Formulas: Logical languages

- A logical language is a mathematical device that defines under what conditions a model makes a formula true.

- **Propositional logic**: Propositions as basic atoms
  - Syntax: propositions (*p*, *q*,..), logical connectives ($\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$)
  - Semantics: truth tables — truth conditions, entailment
  - Limitation: propositions with internal structure

p	q	$p \& q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

Formulas: Logical languages

- A logical language is a mathematical device that defines under what conditions a model makes a formula true.

- First-order predicate logic (FOL): Predicates and arguments
  - Syntax: predicates, constants and variables (*love*(*j*, *m*), *mortal*(x), …), quantifiers (∀,∃), logical connectives (∧ , ∨ , ¬, → , ↔)
  - Semantics: model structures and variable assignments
  - Limitations: none*

First-order predicate logic: Vocabulary

- **Non-logical expressions**:
  - Individual constants: *CON*
  - *n*-place relation constants: $PRED^n$, for all $n \geq 0$
- Infinite set of individual variables: *VAR*
- Logical connectives: $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$
- Brackets: (,)

harry' $\in$ *CON*
prince' $\in$ $PRED^1$
x, y, z, ... $\in$ VAR

First-order predicate logic: Syntax

- **Terms**: $TERM = VAR \cup CON$
- **Atomic formulas**:
  - $R(t_1, \ldots, t_n)$ for $R \in PRED^n$ and $t_1, \ldots, t_n \in TERM$
  - $t_1 = t_2$ for $t_1, t_2 \in TERM$

- **Well-formed formulas** (WFF):
  - All atomic formulas are WFFs
  - If φ and ψ are WFFs, then $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$ are WFFs
  - If $x \in$ VAR, and φ is a WFF, then $\forall x\varphi$ and $\exists x\varphi$ are WFFs
  - Nothing else is a WFF

Well-Formed FOL Formulas?

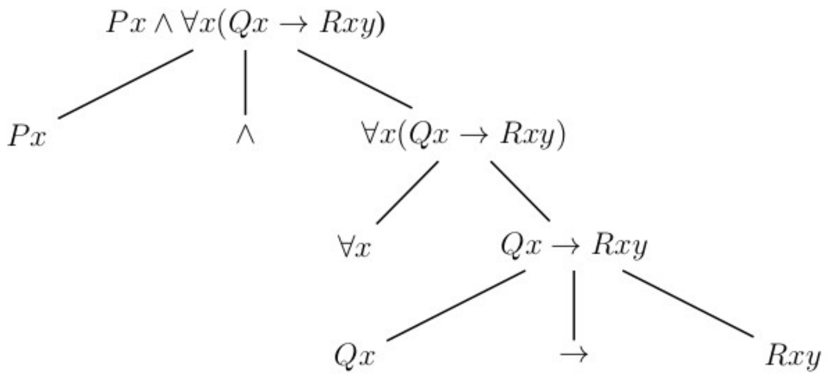
- *prince*
- *prince*(x)
- *prince*(*harry* ∧ *william*)
- ¬*prince*(*harry*)
- *rain* → *happy*(*kate*)
- ∀x(*rain*)
- ∃x(∀x(*happy*(x)))
- ∀x(*prince*(x)) → *handsome*(x)

Well-Formed FOL Formulas?

- ~~*prince*~~ assuming prince $\in$ $PRED^1$
- *prince*(x)
- ~~*prince*(*harry* ∧ *william*)~~ correct: prince(harry) ∧ prince(william)
- ¬*prince*(*harry*) only if interpreted as: ¬(prince(harry))
- *rain* → *happy*(*kate*) only if: rain $\in$ $PRED^0$ (~*"it is raining"*)
- ∀x(*rain*)
- ∃x(∀x(*happy*(x)))
- ∀x(*prince*(x)) → *handsome*(x)

Variable binding

- Given a quantified formula ∀xφ (or ∃xφ), we say that φ (and every part of φ) is in the **scope** of the quantifier
- In a formula ∀xφ (or ∃xφ), the quantifier **binds** all occurrences of x in φ that are not bound by any quantifier occurrence ∀x or ∃x inside φ
- If a variable is not bound in a formula φ, it occurs **free** in φ



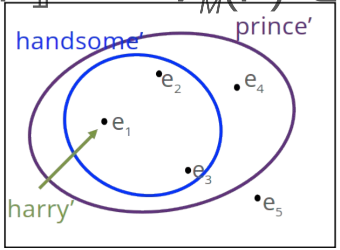
FOL Formulas

- $prince$
- $prince(x)$
- $prince(harry \wedge william)$
- $\neg prince(harry)$
- $rain \rightarrow happy(kate)$
- $\forall x(rain)$
- $\exists x(\forall x(happy(x)))$
- $\forall x(prince(x)) \rightarrow handsome(x)$

First-order predicate logic: Semantics

- FOL formulas obtain a truth value with respect to a model structure *M* and an **assignment function** *g*: $[\![\varphi]\!]^{M,g} \in \{0, 1\}$ (OR $\{\perp, \top\}$)
- First-order model structures are formally defined as tuples $M = (U_M, V_M)$, where U_M is a non-empty set (the **universe**) and V_M is an **interpretation function**:
  - $[\![c]\!]^{M,g} = V_M(c) \in U_M$ if c is an individual constant
  - $[\![P]\!]^{M,g} = V_M(P) \subseteq (U_M)^n$ if P is an *n*-place predicate symbol
  - $[\![P]\!]^{M,g} = V_M(P) \in \{0, 1\}$ if P is an 0-place predicate

M_1:



M1 = ⟨U_1, V_1⟩, where:

U1 = $\{e_1, e_2, e_3, e_4, e_5\}$

V1(prince') = $\{e_1, e_2, e_3, e_4\}$

V1(handsome') = $\{e_1, e_2, e_3\}$

V1(harry') = e_1

$[\![\text{harry'}]\!]^{M1,g} = V_M(\text{harry'}) = e_1$

$[\![\text{handsome'}]\!]^{M1,g} = V_M(\text{handsome'})$

$= \{e_1, e_2, e_3\}$

Assignment function

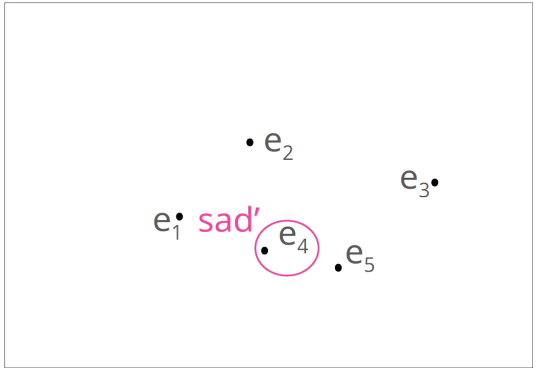
- An assignment function *g* assigns values to all variables—g: $VAR \rightarrow U_M$
  - $[\![x]\!]^{M,g} = g(x) \in U_M$ if x is a variable
  - Write $g[x/d]$ for the assignment function g' that assigns $d \in U_M$ to x and assigns the same values as g to all other variables

	x	y	z	u	...
g	e_1	e_2	e_3	e_4	...
$g[y/e_1]$	e_1	e_1	e_3	e_4	...
$g[x/e_1]$	e_1	e_2	e_3	e_4	...
$g[y/g(z)]$	e_1	e_3	e_3	e_4	...
$g[y/e_1][u/e_1]$	e_1	e_1	e_3	e_1	...
$g[y/e_1][y/e_2]$	e_1	e_2	e_3	e_4	...

Assignment function: Interpretation of quantifiers

- How to interpret the following sentence in model *M*:

  "someone is sad" -> $\exists x(sad'(x))$



- Intuition:
  - find an entity in the universe for which the statement $x \in V_M(sad')$ holds: e_4
  - replace x by e_4 in order to make $\exists x(sad'(x))$ true
- More formally:
  - Interpret the sentence relative to assignment function *g*: i.e., $[\![\exists x(sad'(x))]\!]^{M,g}$, such that $g(x) = e_4$; this can be generalised to any *g'* as follows: $g'[x/e_4](x) = e_4$

Recap: Semantics of first-order predicate logic

- FOL formulas obtain a truth value with respect to a model structure *M* and an **assignment function** *g*: $[\![\varphi]\!]^{M,g} \in \{0, 1\}$ (OR $\{\perp, \top\}$)
- First-order model structures are formally defined as tuples $M = (U_M, V_M)$, where U_M is a non-empty set (the universe) and V_M is an interpretation function:
  - $[\![c]\!]^{M,g} = V_M(c) \in U_M$ if c is an individual constant
  - $[\![P]\!]^{M,g} = V_M(P) \subseteq (U_M)^n$ if P is an *n*-place predicate symbol
  - $[\![P]\!]^{M,g} = V_M(P) \in \{0, 1\}$ if P is an 0-place predicate
- The assignment function *g* assigns values to all variables—$g: VAR \rightarrow U_M$
  - $[\![x]\!]^{M,g} = g(x) \in U_M$ if x is a variable

First-order predicate logic: (more) semantics

- $[\![R(t_1, ..., t_n)]\!]^{M,g} = 1$ iff $([\![t_1]\!]^{M,g}, \ldots, [\![t_n]\!]^{M,g}) \in V_M(R)$
- $[\![t_1 = t_2]\!]^{M,g} = 1$ iff $[\![t_1]\!]^{M,g} = [\![t_2]\!]^{M,g}$
- $[\![\neg\varphi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 0$
- $[\![\varphi \wedge \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 1$ and $[\![\psi]\!]^{M,g} = 1$
- $[\![\varphi \vee \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 1$ or $[\![\psi]\!]^{M,g} = 1$
- $[\![\varphi \rightarrow \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 0$ or $[\![\psi]\!]^{M,g} = 1$
- $[\![\varphi \leftrightarrow \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = [\![\psi]\!]^{M,g}$
- $[\![\exists x\varphi]\!]^{M,g} = 1$ iff there is a $d \in U_M$ such that $[\![\varphi]\!]^{M,g[x/d]} = 1$
- $[\![\forall x\varphi]\!]^{M,g} = 1$ iff for all $d \in U_M$, $[\![\varphi]\!]^{M,g[x/d]} = 1$

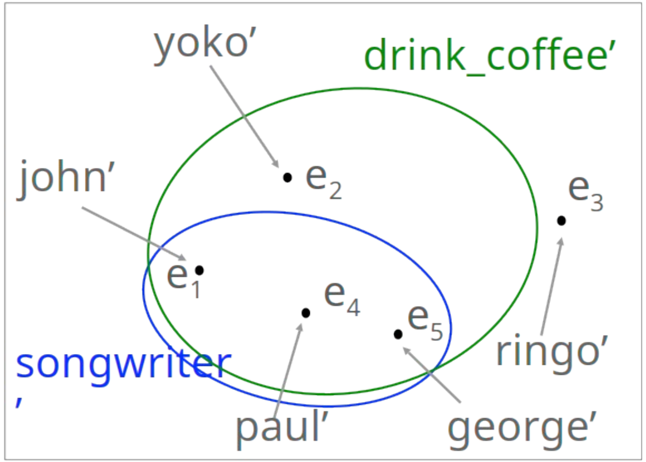
First-order predicate logic: (more) semantics

- $[\![R(t_1, \ldots, t_n)]\!]^{M,g} = 1$ iff $([\![t_1]\!]^{M,g}, \ldots, [\![t_n]\!]^{M,g}) \in V_M(R)$
- $[\![t_1 = t_2]\!]^{M,g} = 1$ iff $[\![t_1]\!]^{M,g} = [\![t_2]\!]^{M,g}$
- $[\![\neg\varphi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 0$
- $[\![\varphi \wedge \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 1$ and $[\![\psi]\!]^{M,g} = 1$
- $[\![\varphi \vee \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 1$ or $[\![\psi]\!]^{M,g} = 1$
- $[\![\varphi \rightarrow \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 0$ or $[\![\psi]\!]^{M,g} = 1$
- $[\![\varphi \leftrightarrow \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = [\![\psi]\!]^{M,g}$
- $[\![\exists x\varphi]\!]^{M,g} = 1$ iff there is a $d \in U_M$ such that $[\![\varphi]\!]^{M,g[x/d]} = 1$
- $[\![\forall x\varphi]\!]^{M,g} = 1$ iff for all $d \in U_M$, $[\![\varphi]\!]^{M,g[x/d]} = 1$

Interpretation of formulas

- *“every songwriter drinks coffee”*
  $\mapsto$ $\forall x(\mathit{songwriter}(x) \rightarrow \mathit{drink\text{-}coffee}(x))$

- **Truth conditions** w.r.t. M_1:
  $[\![\forall x(\mathit{songwriter}(x) \rightarrow \mathit{drink\text{-}coffee}(x))]\!]^{M_1,g} = 1$ iff for all $e \in U_1$:
  - $[\![\mathrm{songwriter}(x) \rightarrow \mathrm{drink\_coffee}(x)]\!]^{M_1,g[x/e]} = 1$
  - $[\![\mathit{songwriter}(x)]\!]^{M_1,g[x/e]} = 0$ or
    $[\![\mathit{drink\text{-}coffee}(x)]\!]^{M_1,g[x/e]} = 1$
  - $[\![x]\!]^{M_1,g[x/e]} \notin V_1(\mathit{songwriter})$ or
    $[\![x]\!]^{M_1,g[x/e]} \in V_1(\mathit{drink\text{-}coffee})$
  - $g[x/e](x) \notin V_1(\mathit{songwriter})$ or
    $g[x/e](x) \in V_1(\mathit{drink\text{-}coffee})$
  - $e \notin V_1(\mathit{songwriter})$ or $e \in V_1(\mathit{drink\text{-}coffee})$



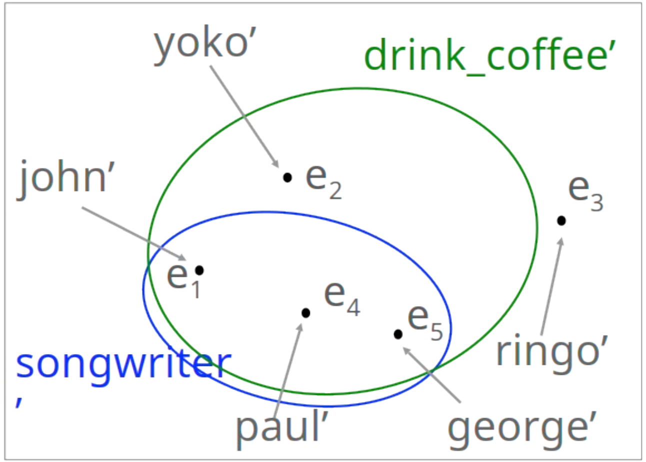
$M_1 = \langle U_1, V_1 \rangle$, where:

- $U_1 = \{e_1, e_2, e_3, e_4, e_5\}$
- $V_1(\text{john'}) = e_1$; $V_1(\text{yoko'}) = e_2$; $V_1(\text{ringo'}) = e_3$; $V_1(\text{paul'}) = e_4$; $V_1(\text{george'}) = e_5$
- $V_1(\text{song-writer'}) = \{e_1, e_4, e_5\}$
  $V_1(\text{drink\_coffee'}) = \{e_1, e_2, e_4, e_5\}$

Interpretation of formulas

- *"every songwriter drinks coffee"*
  $\mapsto \forall x(\textit{songwriter}(x) \rightarrow \textit{drink-coffee}(x))$

- **Truth value** in M_1:
  let $\varphi = \textit{songwriter}(x) \rightarrow \textit{drink-coffee}(x)$
  - For $e \in \{e_1, e_2, e_4, e_5\}$: $[\![\varphi]\!]^{M_1,g[x/e]} = 1$ since $e \in V_1(\textit{drink-coffee})$
  - For $e = e_3$: $[\![\varphi]\!]^{M_1,g[x/e]} = 1$ since $e \notin V_1(\textit{songwriter})$
  - Therefore:
    $[\![\forall x(\textit{songwriter}(x) \rightarrow \textit{drink-coffee}(x))]\!]^{M_1,g} = 1$ for any g



$M_1 = \langle U_1, V_1 \rangle$, where:

- $U_1 = \{e_1, e_2, e_3, e_4, e_5\}$
- $V_1(\text{john'}) = e_1$; $V_1(\text{yoko'}) = e_2$; $V_1(\text{ringo'}) = e_3$; $V_1(\text{paul'}) = e_4$; $V_1(\text{george'}) = e_5$
- $V_1(\text{song-writer'}) = \{e_1, e_4, e_5\}$
  $V_1(\text{drink\_coffee'}) = \{e_1, e_2, e_4, e_5\}$

Exercise: Formalizing Natural Language

1. *Bill loves Mary.*
2. *Bill reads an interesting book.*
3. *Every student reads a book.*
4. *Bill passed every exam.*
5. *Not every student answered every question.*
6. *Only Mary answered every question.*
7. *Mary is annoyed when someone is noisy.*
8. *Although nobody makes noise, Mary is annoyed.*

Truth, validity, and entailment

- A formula φ is **true** in a model *M* iff:
  - $[\![\varphi]\!]^{M,g} = 1$ for every variable assignment *g*

- A formula φ is **valid** ($\vDash \varphi$) iff:
  - φ is true in all models

- A formula φ is **satisfiable** iff:
  - there is *at least one* model *M* such that φ is true in *M*

- A set of formulas Γ *entails* formula φ ($\Gamma \vDash \varphi$) iff:
  - φ is true in every model in which all formulas in Γ are true
  - the elements of Γ are called the **premises** or **hypotheses**
    - φ is called the **conclusion**

Logical Equivalence

- Formula φ is **logically equivalent** to formula ψ ($\varphi \Leftrightarrow \psi$), iff:
  - $[\![\varphi]\!]^{M,g} = [\![\psi]\!]^{M,g}$ for all models *M* and variable assignments *g*

- For all *closed* formulas φ and ψ, the following assertions are *equivalent*:
  - $\varphi \Leftrightarrow \psi$ (logical equivalence)
  - $\varphi \vDash \psi$ and $\psi \vDash \varphi$ (mutual entailment)
  - $\vDash \varphi \leftrightarrow \psi$ (validity of “material equivalence”)

Logical Equivalence Theorems: Propositional Logic

1. $\neg\neg\varphi \Leftrightarrow \varphi$ (double negation)
2. $\varphi \wedge \psi \Leftrightarrow \psi \wedge \varphi$ (commutativity)
3. $\varphi \vee \psi \Leftrightarrow \psi \vee \varphi$
4. $\varphi \wedge (\psi \vee \chi) \Leftrightarrow (\varphi \wedge \psi) \vee (\varphi \wedge \chi)$ (distributivity)
5. $\varphi \vee (\psi \wedge \chi) \Leftrightarrow (\varphi \vee \psi) \wedge (\varphi \vee \chi)$
6. $\neg(\varphi \wedge \psi) \Leftrightarrow \neg\varphi \vee \neg\psi$ (de Morgan’s Laws)
7. $\neg(\varphi \vee \psi) \Leftrightarrow \neg\varphi \wedge \neg\psi$
8. $\varphi \rightarrow \neg\psi \Leftrightarrow \psi \rightarrow \neg\varphi$ (Law of Contraposition)
9. $\varphi \rightarrow \psi \Leftrightarrow \neg\varphi \vee \psi$
10. $\neg(\varphi \rightarrow \psi) \Leftrightarrow \varphi \wedge \neg\psi$

Logical Equivalence Theorems: Quantifiers

1. $\neg\forall x\varphi \Leftrightarrow \exists x\neg\varphi$ (quantifier negation)
2. $\neg\exists x\varphi \Leftrightarrow \forall x\neg\varphi$
3. $\forall x(\varphi \land \Psi) \Leftrightarrow \forall x\varphi \land \forall x\Psi$ (quantifier distribution)
4. $\exists x(\varphi \lor \Psi) \Leftrightarrow \exists x\varphi \lor \exists x\Psi$
5. $\forall x\forall y\varphi \Leftrightarrow \forall y\forall x\varphi$ (quantifier swap)
6. $\exists x\exists y\varphi \Leftrightarrow \exists y\exists x\varphi$
7. $\exists x\forall y\varphi \Rightarrow \forall y\exists x\varphi$ (but <u>not</u> vice-versa)

Logical Equivalence Theorems: Quantifiers and Variables

The following equivalences are valid theorems of FOL, *provided that x does not occur free in φ*:

(φ[x/y] is the result of replacing all free occurrences of y in φ with x)

1. $\exists y\varphi \Leftrightarrow \exists x\varphi[x/y]$
2. $\forall y\varphi \Leftrightarrow \forall x\varphi[x/y]$
3. $\varphi \land \forall x\Psi \Leftrightarrow \forall x(\varphi \land \Psi)$
4. $\varphi \land \exists x\Psi \Leftrightarrow \exists x(\varphi \land \Psi)$
5. $\varphi \lor \forall x\Psi \Leftrightarrow \forall x(\varphi \lor \Psi)$
6. $\varphi \lor \exists x\Psi \Leftrightarrow \exists x(\varphi \lor \Psi)$
7. $\varphi \rightarrow \forall x\Psi \Leftrightarrow \forall x(\varphi \rightarrow \Psi)$
8. $\varphi \rightarrow \exists x\Psi \Leftrightarrow \exists x(\varphi \rightarrow \Psi)$
9. $\exists x\Psi \rightarrow \varphi \Leftrightarrow \forall x(\Psi \rightarrow \varphi)$
10. $\forall x\Psi \rightarrow \varphi \Leftrightarrow \exists x(\Psi \rightarrow \varphi)$

Equivalence Transformations: Example

1. *"nobody masters every problem"* $\mapsto \neg\exists x\forall y(Py \to Rxy)$
2. *"everybody fails to master some problem"* $\mapsto \forall x\exists y(Py \land \neg Rxy)$

- We show the equivalence of 1. and 2. as follows:

$\neg\exists x\forall y(Py \to Rxy)$

$\Leftrightarrow \forall x\neg\forall y(Py \to Rxy)$ $\quad(\neg\exists x\varphi \Leftrightarrow \forall x\neg\varphi)$

$\Leftrightarrow \forall x\exists y\neg(Py \to Rxy)$ $\quad(\neg\forall x\varphi \Leftrightarrow \exists x\neg\varphi)$

$\Leftrightarrow \forall x\exists y(Py \land \neg Rxy)$ $\quad(\neg(\varphi \to \psi) \Leftrightarrow \varphi \land \neg\psi)$