$\forall x$ $\exists x$

The Universal Quantifier

The Existential Quantifier

PHIL 50 - Introduction to Logic

Marcello Di Bello, Stanford University, Spring 2014

Week 7 — Wednesday Class - Syntax and Semantics of Predicate Logic

Summary from Monday (1): Interpretation of Constants and Predicate Symbols

Linguistic ingredient	*Interpretation function **I***	*Value of the interpretation function*
constant symbol ***c***	***I(c)***	one object
1-place predicate symbol ***P***	***I(P)***	set of objects
2-place predicate symbol ***R***	***I(R)***	set of ordered pairs of objects

Illustration: Interpretation of Constants and Predicate Symbols

$I(a) =$  $I(b) =$  $I(c) =$ 

$I(A) = \{$  ,  $\}$ $I(B) = \{$  ,  $\}$ $I(C) = \{$  ,  $\}$

$I(Eat) = \{ \langle$  ,  $\rangle , \langle$  ,  $\rangle \}$

This is just one way to define *I*. There are many other possibilities, of course.

Summary from Monday (2): Truth Conditions for Formulas with Constant and Predicate Symbols

$M \vDash \phi$ *iff* ϕ is true in (relative to) model M.

A model M is a tuple $\langle D, I, g\rangle$ where

- D is the **domain**, i.e. D is a non-empty set of objects
- I is an **interpretation function** where
  - I assigns to every constant symbol an object in D
  - I assigns to a *1*-place predicate symbol a set of objects
  - I assigns to a **2**-place predicate symbol a set of ordered pairs
- g is the assignment function for variables [**to discuss today**]

$\langle D, I, g\rangle \vDash \mathbf{P}(c)$ *iff* $\mathbf{I(c)} \in \mathbf{I(P)}$

$\langle D, I, g\rangle \vDash \mathbf{R}(c_1, c_2)$ *iff* $\langle \mathbf{I(c_1)}, \mathbf{I(c_2)}\rangle \in \mathbf{I(R)}$

Definition of truth conditions for simple formulas

Illustration: Truth Conditions for Formulas with Constant and Predicate Symbols

$I(a) =$  $I(b) =$  $I(c) =$ 

$I(A) = \{$  ,  $\}$ $I(B) = \{$  ,  $\}$ $I(C) = \{$  ,  $\}$

$I(Eat) = \{ \langle$  ,  $\rangle, \langle$  ,  $\rangle \}$

$\mathbf{M} \models \mathbf{A(b)}$ *because* $\mathbf{I(b)} \in I(A)$

$\mathbf{M} \models \mathbf{Eat}(c, a)$ *because* $\langle \mathbf{I(c), I(a)} \rangle \in \mathbf{I(Eat)}$

$\mathbf{M} \not\models \mathbf{Eat}(a, b)$ *because* $\langle \mathbf{I(a), I(b)} \rangle \notin \mathbf{I(Eat)}$

Three Equivalent Notations — Do Not Get Confused!

$M \vDash \phi$	*iff*	ϕ is true in model M
$\langle D, I, g \rangle \vDash \phi$	*iff*	ϕ is true in model $\langle D, I, g \rangle$

The two above are exactly the same, because model **M** is given by $\langle D, I, g \rangle$

$M \vDash_g \phi$	*iff*	ϕ is true in model M and assignment g

This third variant is used in the book and it highlights the variable assignment g.

Let’s now Look at

Existential Quantification

What’s so special about **$\exists xPx$**?

You’ll see!

The Variable Assignment Function g

A model M is a tuple $\langle D, I, g \rangle$ where

- *D* is the **domain**, i.e. *D* is a non-empty set of objects
- *I* is an **interpretation function** where

  I assigns to every constant symbol an object in *D*

  I assigns to *1*-place predicate symbol a set of objects

  I assigns to **2**-place predicate symbol a set of ordered pairs
- *g* assigns to every variable an element of **D**

The assignment function **g** does not do anything different from the interpretation function for constant symbols. While *I* assigns to every constant symbol an object in *D*, the assignment function *g* assigns to every variable symbol an object in *D*.

Illustration: Interpretation for Constants and Assignment Function for Variables Symbols

$I(a) =$  $I(b) =$  $I(c) =$ 

$g(x) =$  $g(y) =$  $g(z) =$ 

This is just one way to define g. There are many other possibilities, of course.

But What's the Difference Between "I" for Constant Symbols and "g" for Variables Symbols? Don't They Do Exactly the Same?

What makes the variable assignment function "g" special is the possibility to modify it

Modifying the Variable Assignment **g** into $\mathbf{g}_{[x:=d]}$

Let **g** be a variable assignment. Let **x** and **y** be variable symbols. Let ***d*** be an object in the domain **D**.

We define $g_{[x:=d]}$ as follows:

$$g_{[x:=d]}(y) = g(y)$$

$$g_{[x:=d]}(x) = d$$

In other words, $g_{[x:=d]}$ assigns the object d to the variable symbol x. With respect to any other variable, g and $g_{[x:=d]}$ are the same, and that's why $g_{[x:=d]}(\mathbf{y}) = g(\mathbf{y})$. To put it another way, the only (possible) difference between g and $g_{[x:=d]}$ is relative to the object they assign to x.

Updated Summary Table

Linguistic ingredient	*Function **I** or **g***	*Value of **I** or **g***
constant symbol ***c***	$\boldsymbol{I(c)}$	one object
1-place predicate symbol ***P***	$\boldsymbol{I(P)}$	set of objects
2-place predicate symbol ***R***	$\boldsymbol{I(R)}$	set of ordered pairs of objects

variable ***x***	$\boldsymbol{g(x)}$	one object
variable ***x***	$\mathbf{g}_{[x:=d]}(\mathbf{x})$	object ***d***

Let's Now See How the Truth of an Existentially Quantified Formula is Assessed

Truth Conditions for formulas with Constant and Predicate Symbols, and Existential Quantifier

$M \vDash \phi$ *iff* ϕ is true in (relative to) model $M = \langle D, I, g \rangle$

Truth conditions for simple formulas

$\langle D, I, g \rangle \vDash \mathbf{P}(c)$ *iff* $\mathbf{I(c)} \in \mathbf{I(P)}$

$\langle D, I, g \rangle \vDash \mathbf{R}(c_1, c_2)$ *iff* $\langle \mathbf{I(c_1)}, \mathbf{I(c_2)} \rangle \in \mathbf{I(R)}$

Truth condition for existentially quantified formulas

$\langle D, I, g \rangle \vDash \exists x\phi$ *iff* there is a d [$d \in D$ and $\langle D, I, g_{[x:=d]} \rangle \vDash \phi$]

First Illustration of

$\langle \boldsymbol{D, I, g} \rangle \vDash \mathbf{\exists x \phi}$ *iff* there is a $\boldsymbol{d}$ $[\boldsymbol{d} \in \boldsymbol{D}$ and $\langle \boldsymbol{D, I, g_{[x:=d]}} \rangle \vDash \boldsymbol{\phi}]$

$\boldsymbol{I(A)}=\{$  ,  $\}$ $\boldsymbol{I(B)}=\{$  ,  $\}$ $\boldsymbol{I(C)}=\{$  ,  $\}$

$\mathbf{M} \vDash \mathbf{\exists x(A(x))}$

iff there is a $\boldsymbol{d}$ $[\boldsymbol{d} \in \boldsymbol{D}$ and $\langle \boldsymbol{D, I, g_{[x:=d]}} \rangle \vDash \mathbf{A(x)}]$
iff there is a $\boldsymbol{d}$ $[\boldsymbol{d} \in \boldsymbol{D}$ and $\boldsymbol{g_{[x:=d]}(x)} \in \boldsymbol{I(A)}]$
iff there is a $\boldsymbol{d}$ $[\boldsymbol{d} \in \boldsymbol{D}$ and $\boldsymbol{d} \in \boldsymbol{I(A)}]$ $[b/c\ \boldsymbol{g_{[x:=d]}(x)=d}]$

Since it is the case that there is a $\boldsymbol{d}$ such that $\boldsymbol{d} \in \boldsymbol{D}$ and $\boldsymbol{d} \in \boldsymbol{I(A)}$, we can say that $\mathbf{M} \vDash \mathbf{\exists x(A(x))}$.

Second Illustration of

$\langle D, I, g\rangle \vDash \exists x\phi$ *iff* there is a ***d*** $[d\in D$ and $\langle D, I, g_{[x:=d]}\rangle \vDash \phi]$

$I(a) =$  $I(b) =$  $I(c) =$ 

$I(Eat) = \{\langle$  ,  $\rangle, \langle$  ,  $\rangle\}$

M $\vDash$ **∃x(Eat(x, b))**

iff there is a ***d*** $[d\in D$ and $\langle D, I, g_{[x:=d]}\rangle \vDash$ **Eat(x, b)**]

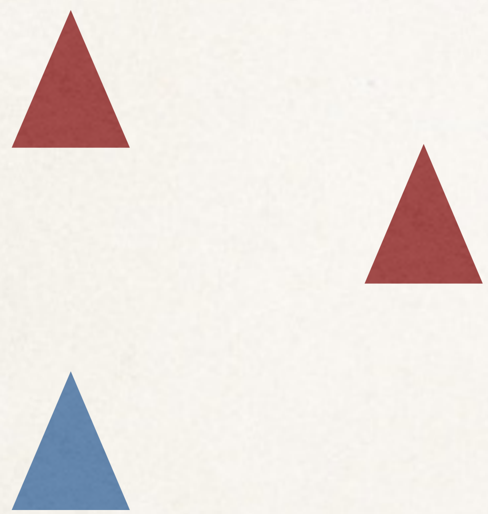
iff there is a ***d*** $[d\in D$ *and* $\langle g_{[x:=d]}(x), I(b)\rangle \in I(Eat)]$

iff there is a ***d*** $[d\in D$ and $\langle d, I(b)\rangle \in I(Eat)]$

Since it is the case that there is a ***d*** such that $d\in D$ and $\langle d, I(b)\rangle \in I(Eat)$, we can say that **M** $\vDash$ **∃x(Eat(x, b)).**

Is the Existential Quantifier Really Necessary? Can we Do Away With it?

The Existential Quantifier Is a Hidden Disjunction



Natural Language: Something is red

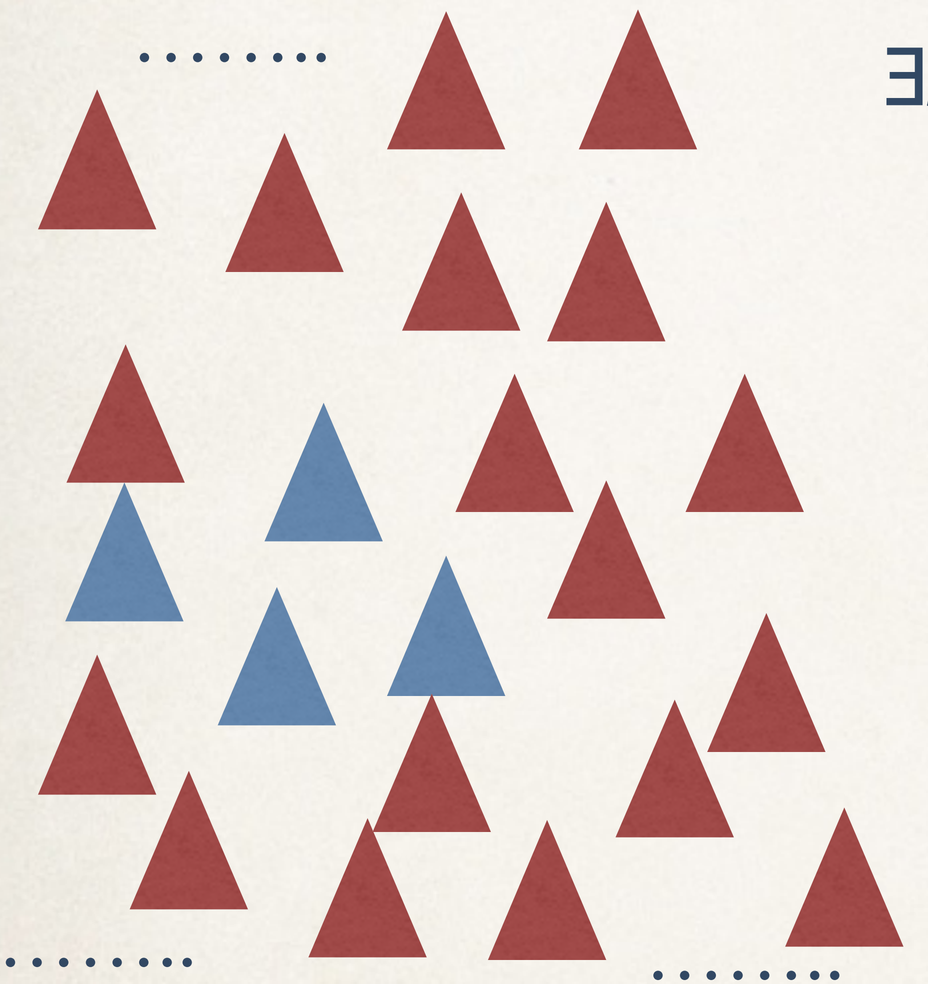
Predicate logic: $\boldsymbol{\exists x(Red(x))}$

Without quantifier: $\boldsymbol{Red(a) \vee Red(b) \vee Red(c)}$

Formulas with the existential quantifier are abbreviations of potentially very long disjunctions.

To turn the existential quantifier into a disjunction, we need to suppose that we have constant symbols **a, b, c** for every object.

But Suppose You Had an Infinite Number of Objects



$\exists x(Red(x))$

To turn the existential quantifier into a disjunction, we would need an infinite number of constant symbols **a, b, c, … ,** each for every object.

Absent the existential quantifier, we would need a **formula with an infinite number of disjuncts**!

Equivalent formula without quantifier:

$\boldsymbol{Red(a) \lor Red(b) \lor Red(c) \lor \ldots\ldots \lor \ldots\ldots}$

The existential quantifier is an example of how logic makes the infinite finite.

The Power of Existential Quantification

Given an infinite domain, an existentially quantified formula is equivalent to a formula consisting of an <u>infinite number of disjuncts</u>.

Formulas of predicate logic can <u>only contain a finite number of disjuncts</u>. So, existentially quantified formulas are a device to go beyond this limitation without introducing infinite disjunctions.

Why Should We Care About the Possibility of an Infinite Domain of Objects?

Think of arithmetic and the natural numbers, which are infinite. If predicate logic wants to be a tool to formalize mathematical reasoning, predicate logic should be able to talk about an infinite domain of objects.

Let's now Look at

Universal Quantification

Truth Conditions for Formulas with Constant and Predicate Symbols, and Quantifiers

$\boldsymbol{M} \vDash \phi$ *iff* ϕ is true in (relative to) model $\boldsymbol{M} = \langle \boldsymbol{D}, \boldsymbol{I}, \boldsymbol{g} \rangle$

Truth condition for existentially quantified formulas

$\langle \boldsymbol{D}, \boldsymbol{I}, \boldsymbol{g} \rangle \vDash \exists x \phi$ *iff* there is $\boldsymbol{d} \in \boldsymbol{D}$ and $\langle \boldsymbol{D}, \boldsymbol{I}, \boldsymbol{g}_{[x:=d]} \rangle \vDash \phi$

Truth condition for universally quantified formulas

$\langle \boldsymbol{D}, \boldsymbol{I}, \boldsymbol{g} \rangle \vDash \forall x \phi$ *iff* for all **d**, *if* $\boldsymbol{d} \in \boldsymbol{D}$, *then* $\langle \boldsymbol{D}, \boldsymbol{I}, \boldsymbol{g}_{[x:=d]} \rangle \vDash \phi$

Illustration of

$\langle D, I, g\rangle \vDash \forall \mathbf{x}\phi$ *iff* for all **d**, *if* $d \in D$, *then* $\langle D, I, g_{[x:=d]}\rangle \vDash \phi$

D ={  ,  ,  }

$I(A)$={  ,  }

$I(B)$= {  ,  }

$I(C)$={  ,  }

$\mathbf{M} \vDash \forall \mathbf{x}(\mathbf{A}(\mathbf{x}))$

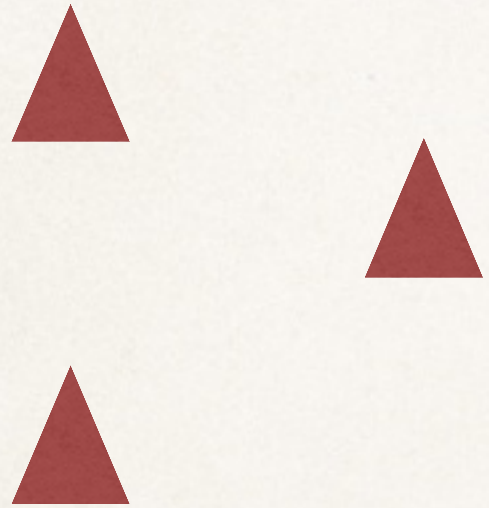
iff for all **d**, if $d \in D$, then $\langle D, I, g_{[x:=d]}\rangle \vDash \mathbf{A}(\mathbf{x})$

iff for all **d**, if $d \in D$, then $g_{[x:=d]}(x) \in$ **I(A)**

iff for all **d**, if $d \in D$, then $d \in$ **I(A)**

Since it is **NOT** the case that for all **d**, if $d \in D$, then $d \in$ **I(A)** ,
it follows that $\mathbf{M} \nvDash \forall \mathbf{x}(\mathbf{A}(\mathbf{x}))$.

The Universal Quantifier as a Hidden Conjunction



Natural Language: Everything is red

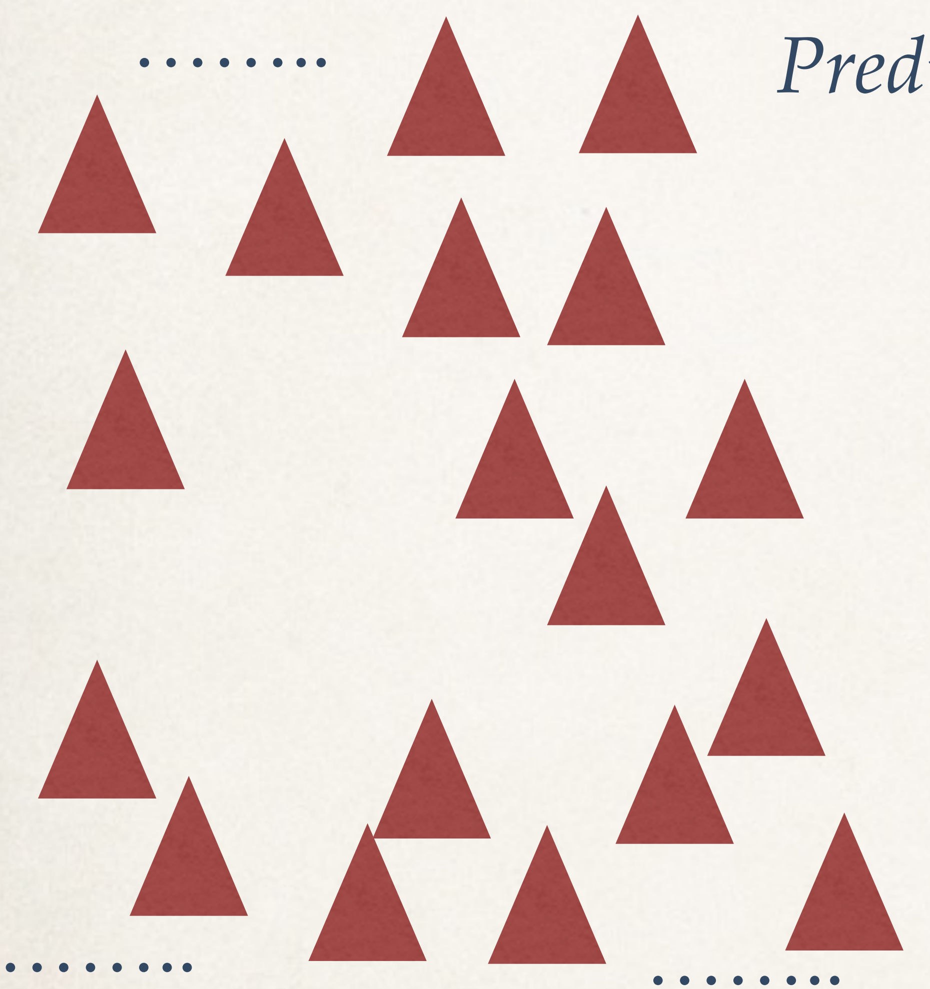
Predicate logic: $\forall x(Red(x))$

Without quantifier: $Red(a) \wedge Red(b) \wedge Red(c)$

Formulas with the universal quantifier are abbreviations of potentially very long conjunctions.

To turn the universal quantifier into a conjunction, we need to suppose that we have constant symbols **a, b, c** for every object.

But Suppose You Had an Infinite Number of Objects



Predicate logic: $\forall x(Red(x))$

To turn the universal quantifier into a conjunction, we would need an infinite number of constant symbols **a, b, c, … ,** each for every object.

Equivalent formula without quantifier:
$Red(a) \wedge Red(b) \wedge Red(c) \wedge \ldots\ldots \wedge \ldots$

Absent the universal quantifier, we would need a **formula with an infinite number of conjuncts**!

The Power of Universal Quantification

The universal quantifier is an example of how logic makes the infinite finite.

Given an infinite domain, a universally quantified formula is equivalent to a formula consisting of an <u>infinite number of conjuncts</u>.

Formulas of predicate logic can <u>only contain a finite number of conjuncts</u>. So, universally quantified formulas are a device to go beyond this limitation without introducing infinite conjunctions.

Let's now Look at

Propositional Connectives

Truth Conditions for Formulas Containing Connectives

The connectives in predicate logic do not behave any differently from propositional logic. However, the way in which we shall write their truth conditions slightly different from what we did in the case of propositional logic.

$\mathbf{M} \vDash \neg \phi$	*iff*	*it is not the case that* $\mathbf{M} \vDash \phi$, *i.e.* $\mathbf{M} \nvDash \phi$
$\mathbf{M} \vDash \phi \wedge \psi$	*iff*	$\mathbf{M} \vDash \phi$ *and* $\mathbf{M} \vDash \psi$
$\mathbf{M} \vDash \phi \vee \psi$	*iff*	$\mathbf{M} \vDash \phi$ *or* $\mathbf{M} \vDash \psi$
$\mathbf{M} \vDash \phi \rightarrow \psi$	*iff*	$\mathbf{M} \vDash \phi$ *implies* $\mathbf{M} \vDash \psi$

Assessing the Truth of Formulas with Constants, Predicate Symbols, and Connectives

$I(a) =$  $I(b) =$  $I(c) =$ 

$I(A) = \{$  ,  $\}$ $I(B) = \{$  ,  $\}$ $I(C) = \{$  ,  $\}$

$I(Eat) = \{ \langle$  ,  $\rangle , \langle$  ,  $\rangle \}$

$\mathbf{M} \models \neg\, \mathbf{A(c)}$

$\mathbf{M} \models \mathbf{Eat}(c, a) \wedge \mathbf{Eat}(c, b)$

$\mathbf{M} \models \mathbf{Eat}(a, c) \rightarrow \mathbf{Eat}(c, b)$

b/c $\mathbf{I(c)} \notin I(A)$

b/c $\langle \mathbf{I(c)}, \mathbf{I(a)} \rangle \in \mathbf{I(Eat)}$ *and* $\langle \mathbf{I(c)}, \mathbf{I(b)} \rangle \in \mathbf{I(Eat)}$

b/c $\langle \mathbf{I(a)}, \mathbf{I(c)} \rangle \in \mathbf{I(Eat)}$ *implies* $\langle \mathbf{I(c)}, \mathbf{I(b)} \rangle \in \mathbf{I(Eat)}$

[vacuously b/c antecedent is false]

Summary: Truth Conditions for Formulas in Predicate Logic so far

$\langle D, I, g\rangle \vDash P(c)$ *iff* $\mathbf{I(c) \in I(P)}$

$\langle D, I, g\rangle \vDash R(c_1, c_2)$ *iff* $\langle \mathbf{I(c_1), (I(c_2)} \rangle \in \mathbf{I(R)}$

$\langle D, I, g\rangle \vDash \neg \phi$ *iff* $\langle D, I, g\rangle \nvDash \phi$

$\langle D, I, g\rangle \vDash \phi \wedge \psi$ *iff* $\langle D, I, g\rangle \vDash \phi$ ***and*** $\langle D, I, g\rangle \vDash \psi$

$\langle D, I, g\rangle \vDash \phi \vee \psi$ *iff* $\langle D, I, g\rangle \vDash \phi$ ***or*** $\langle D, I, g\rangle \vDash \psi$

$\langle D, I, g\rangle \vDash \phi \rightarrow \psi$ *iff* $\langle D, I, g\rangle \vDash \phi$ ***implies*** $\langle D, I, g\rangle \vDash \psi$

$\langle D, I, g\rangle \vDash \exists x\phi$ *iff* there is a d [$d \in D$ and $\langle D, I, g_{[x:=d]}\rangle \vDash \phi$]

$\langle D, I, g\rangle \vDash \forall x\phi$ *iff* for all d [*if* $d \in D$, *then* $\langle D, I, g_{[x:=d]}\rangle \vDash \phi$]