

# **22c:145 Artificial Intelligence**

## **First-Order Logic**

**Readings: Chapter 8 of Russell &  
Norvig.**

# Einstein's Puzzle in Logic

- We used propositional variables to specify everything:  
 $x_1$  = “house #1 is red”;  
 $x_2$  = “house #2 is red”;  
...  
 $x_{26}$  = “Brit lives in house #1”;  
...
- Can we use one symbol with parameters for “the color of house # $x$  is  $y$ ”, i.e.,  $IsColor(x, y)$ ?
- Or simply,  $color(x) = y$ ?
- Yes, we can do this in First-Order Logic.

# First-Order Logic

- Rather powerful representation and reasoning system.
- Very well understood and extensively studied (a couple of thousand years!).
- Many fancy knowledge representation formalisms—semantic nets, frames, scripts—are basically sugar-coated variants of (part of) it.

# Ontological Commitments

1. The world is made of **objects**, *things* with individual identities and **properties** that distinguish them.
2. Various **relations** hold among objects. Some of these relations are **functional**.
3. Every fact involving objects and their relations is either true or false.

# The Language of First-Order Logic

## Symbols

- Variables
- Constant symbols
- Functional symbols (with *arities*)
- Relational symbols (with *arities*)
- Logical constants: As in Propositional Logic plus

$\forall$  (for all)     $\exists$  (there exists)     $=$  (equal)

# Language of FOL: Symbols

- Variables, constant symbols, and function symbols are used to build **terms**:

$x, Bill, 6, FatherOf(x), Height(FatherOf(Bill)), Log(3 + y), \dots$

- Relational symbols are applied to terms to build **predicates**:

$Even(x), Married(Bill, Hillary), Loves(x, MotherOf(x)), \dots$

- Predicates and logical constants are used to build **sentences**:

$Even(3), \exists x \neg Loves(x, MotherOf(x)), \forall x Even(x) \Rightarrow Odd(x + 1), \dots$

# Language of FOL: Terms

- A variable is a term.
- A constant symbol is a term.
- If  $f$  is an  $n$ -ary function symbol and  $t_1, \dots, t_n$  are  $n$  terms,  $f(t_1, \dots, t_n)$  is a term.
- Nothing else is a term.

# Language of FOL: Sentences

- True (False) is a sentence.
- If  $t_1, t_2$  are terms,  $t_1 = t_2$  is a sentence.
- If  $p$  is an  $n$ -ary relation symbol and  $t_1, \dots, t_n$  are  $n$  terms,  $p(t_1, \dots, t_n)$  is a sentence.
- If  $\varphi$  is a sentence,  $\neg\varphi$ ,  $\exists x \varphi$ ,  $\forall x \varphi$  are sentences.
- If  $\varphi_1, \varphi_2$  are sentences,  $\varphi_1 \wedge \varphi_2$ ,  $\varphi_1 \vee \varphi_2$ ,  $\varphi_1 \Rightarrow \varphi_2$ ,  $\varphi_1 \Leftrightarrow \varphi_2$  are sentences.
- Nothing else is a sentence.

# Language of FOL: Grammar

Sentence	::=	AtomicS   ComplexS
AtomicS	::=	<b>True</b>   <b>False</b>   RelationSymb(Term, ...)   Term = Term
ComplexS	::=	(Sentence)   Sentence Connective Sentence   $\neg$ Sentence   Quantifier Sentence
Term	::=	FunctionSymb(Term, ...)   ConstantSymb   Variable
Connective	::=	$\wedge$   $\vee$   $\Rightarrow$   $\Leftrightarrow$
Quantifier	::=	$\forall$ Variable   $\exists$ Variable
Variable	::=	$a$   $b$   ...   $x$   $y$   ...
ConstantSymb	::=	$A$   $B$   ...   <i>John</i>   0   1   ...   $\pi$   ...
FunctionSymb	::=	$F$   $G$   ...   <i>Cosine</i>   <i>Height</i>   <i>FatherOf</i>   +   ...
RelationSymb	::=	$P$   $Q$   ...   <i>Red</i>   <i>Brother</i>   <i>Apple</i>   >   ...

# A Note on the Syntax of FOL

As defined, the syntax of FOL is ambiguous:

$$\forall x \neg P(x) \vee Q(x) \wedge R(x) \Rightarrow S(x) \quad (1)$$

It can be disambiguated by using parentheses or fixing a precedence ordering on the logical connectives. We will use the following ordering:

$$(\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow, \{\forall, \exists\})$$

With this ordering, (1) actually is

$$\forall x (((\neg P(x)) \vee (Q(x) \wedge R(x))) \Rightarrow S(x))$$

When in doubt, use parentheses!

# Semantics of First-Order Logic

## Interprets

- constant symbols and variables as *objects*;
- functional symbols as *functions* from objects to objects;
- relational symbols as *relations* over objects;
- = as equality (i.e., the *identity* relation).

# Constant Symbols and Variables

- Denote (stand for) objects/individuals.

Example:

- $Dog143$  is a constant symbol denoting a particular dog.
- $peach$  is a variable ranging over peaches.
- The symbol  $\longrightarrow$  object association is arbitrary!
- Under another interpretation  $Dog143$  could denote another dog, or maybe something else altogether. Similarly,  $peach$  might as well range over apples.

# Semantics of First-Order Logic

## Interprets

- the universal quantifier (essentially) as an *infinite conjunction*;

$$(\forall x \text{ Red}(x) \equiv \text{Red}(\text{Obj}_1) \wedge \text{Red}(\text{Obj}_2) \wedge \text{Red}(\text{Obj}_3) \wedge \text{Red}(\text{Obj}_4) \wedge \dots)$$

- the existential quantifier (essentially) as an *infinite disjunction*;

$$(\exists x \text{ Red}(x) \equiv \text{Red}(\text{Obj}_1) \vee \text{Red}(\text{Obj}_2) \vee \text{Red}(\text{Obj}_3) \vee \text{Red}(\text{Obj}_4) \vee \dots)$$

- **True, False,  $\wedge$ ,  $\vee$ ,  $\neg$ ,  $\Rightarrow$ ,  $\Leftrightarrow$**  as in propositional logic.

# Semantics of FOL: Examples

Sentence	Intuitive meaning
1. $Above(A, B)$	object A is above object B
2. $British(Elisabeth)$	an “object” called Elisabeth is British
3. $\forall x Apple(x) \Rightarrow Red(x)$	...
4. $\forall x Apple(x) \wedge Red(x)$	...
5. $\exists x Apple(x) \vee Red(x)$	...
6. $\forall x \exists y y > x$	...
6b. $\forall x Number(x) \Rightarrow \exists y y > x$	...
7. $(\forall x Apple(x)) \vee \exists y Pear(y)$	...
7b. $(\forall x Apple(x)) \vee \exists x Pear(x)$	...
8. $\forall x \exists y Loves(x, y)$	...
8b. $\exists y \forall x Loves(x, y)$	...
9. $\exists x \forall y Loves(x, y)$	...

# Semantics of FOL: Examples

	Sentence	Intuitive meaning
10.	$\forall x (Bird(x) \wedge \neg Ostrich(x)) \Rightarrow Flies(x)$	...
11.	$\forall x (Fish(x) \wedge \neg Ostrich(x)) \Rightarrow Flies(x)$	...
12.	$\forall x Flies(x) \Leftrightarrow (Bird(x) \vee Plane(x))$	...
13.	$\forall x Person(x) \Rightarrow$ $(W(x) \vee M(x)) \wedge \neg(W(x) \wedge M(x))$	...
14.	$\forall x Age(x) < Age(FatherOf(x))$	...

**Note:** The **scope** of a quantifier is the sentence it applies to:

$$\forall x \underbrace{P(x) \wedge Q(x)}_{\text{scope of } \forall x}$$

$$\forall x \underbrace{R(x, z) \wedge \underbrace{\exists y S(x, y)}_{\text{scope of } \exists y}}_{\text{scope of } \forall x}$$

# Free and Bound Variables

An occurrence of variable in a sentence is **free** if it is not in the scope of any quantifier with the same variable. A non-free variable occurrence is **bound**.

$$\forall x P(x) \wedge Q(x)$$

both occurrences of  $x$  bound by  $\forall x$

$$(\forall x P(x)) \wedge Q(x)$$

first occurrence of  $x$  bound by  $\forall x$ , second free

$$\forall x R(x, z) \wedge \exists y S(x, y)$$

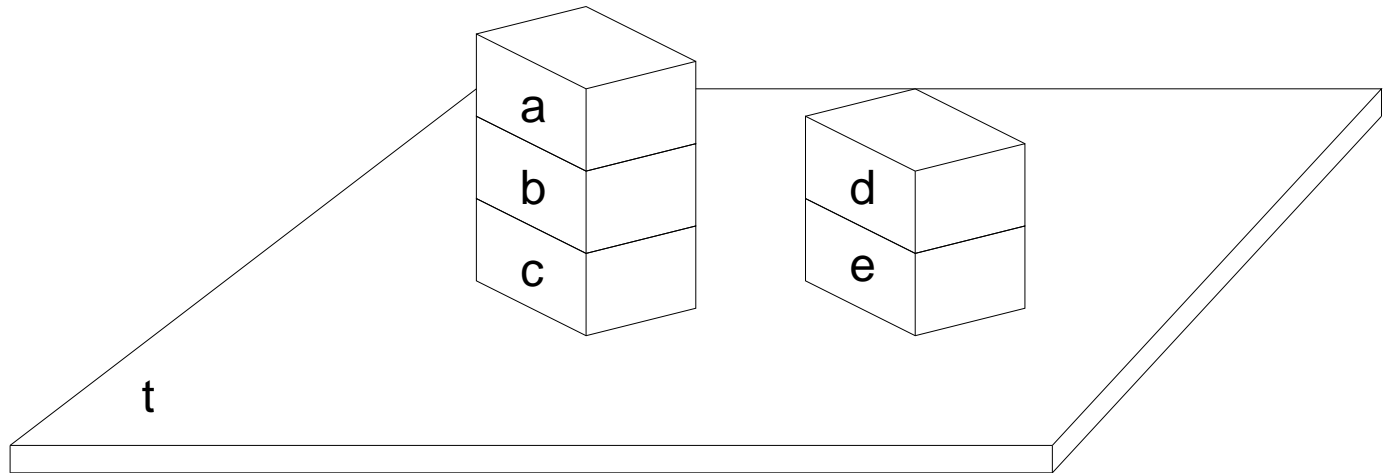
both occurrences of  $x$  bound by  $\forall x$ , occurrence of  $y$  bound by  $\exists y$ , occurrence of  $z$  free

A sentence is **open** if it contains free variables; it is **closed** otherwise.

# Formalizing Knowledge in FOL

First Step: Fix a **Universe of Discourse**, that is, the set of objects of interest.

- Example: The Blocks World

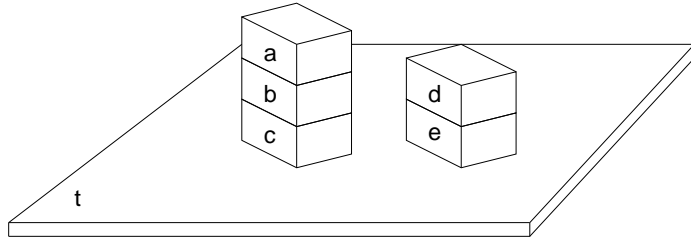


Here the universe is  $\{ \text{the table, block a, block b, } \dots, \text{block e} \}$ .

- Note: The universe may be infinite, even uncountable!  
(the integers, the reals, ...)

# Relations

- Let  $U$  be the universe of discourse.
- A relation of arity  $n$  is a subset of the Cartesian product  $\underbrace{U \times \dots \times U}_{n \text{ times}}$ .
- Example: Some relations in the Blocks World



$On := \{\langle a, b \rangle, \langle b, c \rangle, \langle c, t \rangle, \langle d, e \rangle, \langle e, t \rangle\}$

$Clear := \{\langle a \rangle, \langle d \rangle\}$

- Example: “Less Than” over the naturals

$< := \{\langle x, y \rangle \mid x, y \in \mathbb{N}, \exists z \neq 0 \in \mathbb{N} \text{ s.t. } x + z = y\}$

$:= \{\langle 0, 1 \rangle, \langle 0, 2 \rangle, \dots, \langle 1, 2 \rangle, \langle 1, 3 \rangle, \dots\}$

# Functions

- A function in one variable is simply a *functional* binary relation.
- A relation  $f$  over  $U \times U$  is functional if for all  $x \in U$  there is *one and only one*  $y \in U$  such that  $\langle x, y \rangle \in f$ .
- If  $f$  is a function

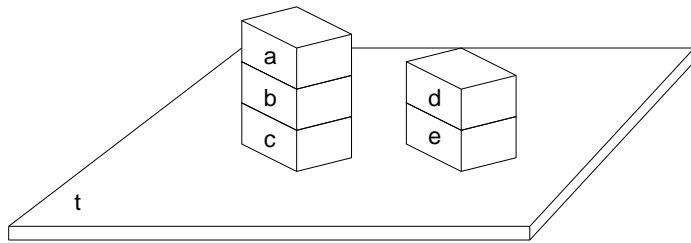
$$(\langle x, y \rangle \in f) \wedge (\langle x, z \rangle \in f) \quad \text{implies} \quad y = z$$

Hence, instead of writing  $\langle x, y \rangle \in f$  we write  $y = f(x)$ .

- Functions in  $n > 1$  variables are defined similarly.

# Examples of Functions

- Most mathematical functions  $+$ ,  $\times$ ,  $e^x$ ,  $sine$ , ...
- Some relations in the Blocks World



$Color := \{\langle a, blue \rangle, \langle b, red \rangle, \langle c, blue \rangle, \dots\}$

$Support := \{\langle a, b \rangle, \langle b, c \rangle, \langle c, t \rangle, \langle d, e \rangle, \langle e, t \rangle\}$

- We may define function  $color(x)$  such that  $color(a) = blue$ ,  $color(b) = red$ , ...
- **Note:** Functions in FOL are always **total**, i.e., defined over the entire universe of discourse.

# Relations and Predicates

- The notation  $Mother(Jane, Bill)$  is intended to mean that

$$\langle Jane, Bill \rangle \in Mother$$

- A relation implicitly defines (or provides the meaning of) a **predicate**, i.e., a function ranging over  $\{True, False\}$ .
- *Predicate* is a syntactic way of representing a relation (in the style of a function).

# Functions and Names

- The functional notation can be used to create several names for the *same* object.
- Examples:

$3$ ,  $\sqrt{9}$ ,  $1 + 2$ ,  $age(Baby)$

$George\_Bush$ ,  $PresidentOf(US)$ ,  $SonOf(HusbandOf(Barbara$

# Semantics of First-Order Logic

(A little) more formally:

An **interpretation** is a pair  $(\mathcal{D}, \sigma)$  where

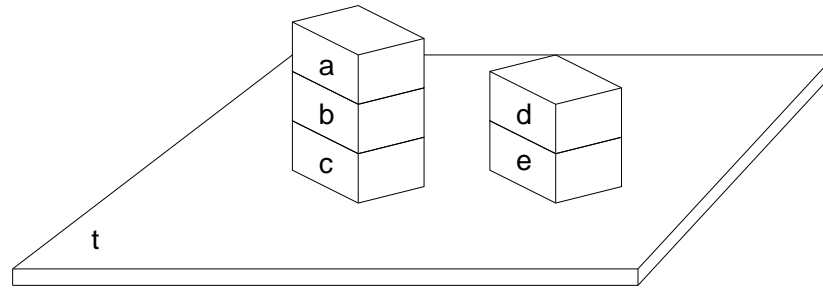
- $\mathcal{D}$  is a set of objects, the universe (or *domain*);
- $\sigma$  is mapping from variables to objects in  $\mathcal{D}$ ;
- $C^{\mathcal{D}}$  is an object in  $\mathcal{D}$  for every constant symbol  $C$ ;
- $F^{\mathcal{D}}$  is a function from  $\mathcal{D}^n$  to  $\mathcal{D}$  for every function symbol  $F$  of arity  $n$ ;
- $R^{\mathcal{D}}$  is a relation over  $\mathcal{D}^n$  for every relation symbol  $R$  of arity  $n$ ;

# An Interpretation $\mathcal{A}$ in the Blocks World

Constant Symbols:  $A, B, C, D, E, T$

Function Symbols:  $Support$

Relation Symbols:  $On, Above, Clear$



$$A^{\mathcal{A}} = a, B^{\mathcal{A}} = b, C^{\mathcal{A}} = c, D^{\mathcal{A}} = d, E^{\mathcal{A}} = e, F^{\mathcal{A}} = f, T^{\mathcal{A}} = t$$

$$Support^{\mathcal{A}} = \{\langle a, b \rangle, \langle b, c \rangle, \langle c, t \rangle, \langle d, e \rangle, \langle e, t \rangle, \langle t, t \rangle\}$$

$$On^{\mathcal{A}} = \{\langle a, b \rangle, \langle b, c \rangle, \langle d, e \rangle\}$$

$$Above^{\mathcal{A}} = \{\langle a, b \rangle, \langle a, c \rangle, \langle b, c \rangle, \langle d, e \rangle\}$$

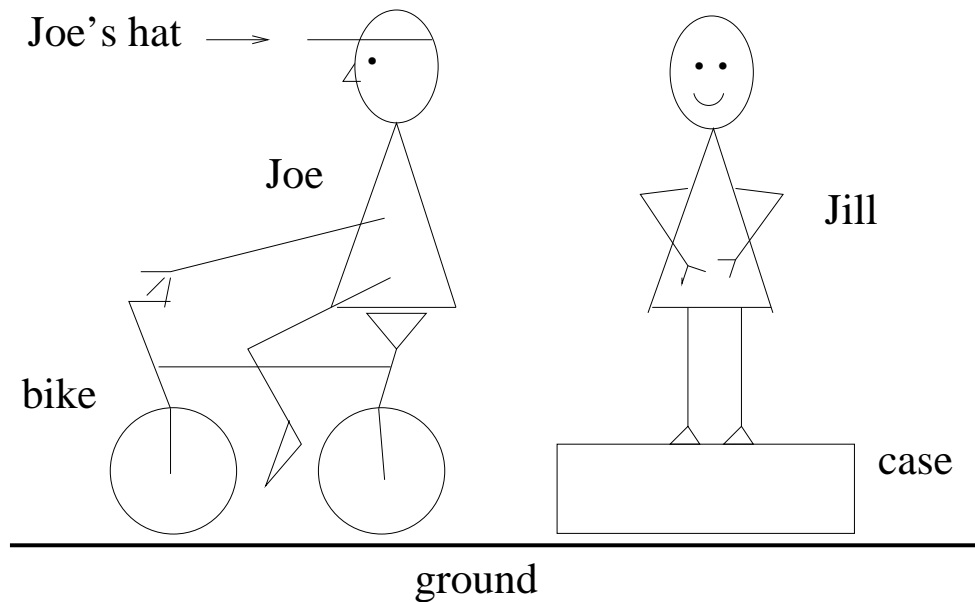
$$Clear^{\mathcal{A}} = \{\langle a \rangle, \langle d \rangle\}$$

# A different interpretation: $\mathcal{B}$

Constant Symbols:  $A, B, C, D, E, T$

Function Symbols:  $Support$

Relation Symbols:  $On, Above, Clear$



$A^{\mathcal{B}} = Joe'sHat, B^{\mathcal{B}} = Joe, C^{\mathcal{B}} = bike, D^{\mathcal{B}} = Jill, E^{\mathcal{B}} = case, T^{\mathcal{B}} = ground$

$Support^{\mathcal{B}} = \{\langle Joe's\ hat, Joe \rangle, \langle Joe, bike \rangle, \langle bike, ground \rangle, \langle Jill, case \rangle, \langle case, ground \rangle, \langle ground, ground \rangle\}$

$On^{\mathcal{B}} = \{\langle Joe's\ hat, Joe \rangle, \langle Joe, bike \rangle, \langle Jill, case \rangle, \}$

$Above^{\mathcal{B}} = \{\langle Joe's\ hat, Joe \rangle, \langle Joe's\ hat, bike \rangle, \dots\}$

$Clear^{\mathcal{B}} = \{\langle Jill \rangle\ (no\ hat), Joe's\ hat\}$

# Semantics of First-Order Logic

- Let  $(\mathcal{D}, \sigma)$  be an interpretation and  $E$  an expression of FOL. We write  $\llbracket E \rrbracket_{\sigma}^{\mathcal{D}}$  to denote the *meaning of  $E$  in the domain  $\mathcal{D}$  under the variable assignment  $\sigma$* .
- The meaning  $\llbracket t \rrbracket_{\sigma}^{\mathcal{D}}$  of a term  $t$  is an object of  $\mathcal{D}$ . It is inductively defined as follows.

$$\llbracket x \rrbracket_{\sigma}^{\mathcal{D}} \quad := \quad \sigma(x) \quad \text{for all variables } x$$

$$\llbracket C \rrbracket_{\sigma}^{\mathcal{D}} \quad := \quad C^{\mathcal{D}} \quad \text{for all constant symbols } C$$

$$\llbracket F(t_1, \dots, t_n) \rrbracket_{\sigma}^{\mathcal{D}} \quad := \quad F^{\mathcal{D}}(\llbracket t_1 \rrbracket_{\sigma}^{\mathcal{D}}, \dots, \llbracket t_n \rrbracket_{\sigma}^{\mathcal{D}}) \quad \text{for all function symbols } F \text{ of arity } n$$

# Example

- Consider the symbols  $MotherOf$ ,  $SchoolOf$ ,  $Bill$  and the interpretation  $(\mathcal{D}, \sigma)$  where

$MotherOf^{\mathcal{D}}$  is a unary fn mapping people to their mother

$SchoolOf^{\mathcal{D}}$  is a unary fn mapping people to their school

$FchildOf^{\mathcal{D}}$  is a binary fn mapping a couple to their first child

$Bill^{\mathcal{D}}$  is Bill Clinton

$$\sigma := \{x \mapsto \text{Chelsea Clinton}, y \mapsto \text{Hillary Clinton}\}$$

- What is the meaning of  $MotherOf(x)$  according to  $(\mathcal{D}, \sigma)$ ?

$$\llbracket MotherOf(x) \rrbracket_{\sigma}^{\mathcal{D}} = \llbracket MotherOf \rrbracket_{\sigma}^{\mathcal{D}} (\llbracket x \rrbracket_{\sigma}^{\mathcal{D}}) = MotherOf^{\mathcal{D}} (\sigma(x)) = \text{Hillary Clinton}$$

- What is the meaning of  $SchoolOf(FchildOf(y, Bill))$ ?

$$\llbracket SchoolOf(FchildOf(y, Bill)) \rrbracket_{\sigma}^{\mathcal{D}} = SchoolOf^{\mathcal{D}} (FchildOf^{\mathcal{D}} (\sigma(y), Bill^{\mathcal{D}})) = S$$

# Semantics of First-Order Logic

- The meaning  $\llbracket \varphi \rrbracket_{\sigma}^{\mathcal{D}}$  of a formula  $\varphi$  is either *True* or *False*.
- It is inductively defined as follows.

$\llbracket t_1 = t_2 \rrbracket_{\sigma}^{\mathcal{D}} := \text{True}$  iff  $\llbracket t_1 \rrbracket_{\sigma}^{\mathcal{D}}$  is the same as  $\llbracket t_2 \rrbracket_{\sigma}^{\mathcal{D}}$

$\llbracket R(t_1, \dots, t_n) \rrbracket_{\sigma}^{\mathcal{D}} := \text{True}$  iff  $\langle \llbracket t_1 \rrbracket_{\sigma}^{\mathcal{D}}, \dots, \llbracket t_n \rrbracket_{\sigma}^{\mathcal{D}} \rangle \in R^{\mathcal{D}}$

$\llbracket \neg \varphi \rrbracket_{\sigma}^{\mathcal{D}} := \text{True/False}$  iff  $\llbracket \varphi \rrbracket_{\sigma}^{\mathcal{D}} = \text{False/True}$

$\llbracket \varphi_1 \vee \varphi_2 \rrbracket_{\sigma}^{\mathcal{D}} := \text{True}$  iff  $\llbracket \varphi_1 \rrbracket_{\sigma}^{\mathcal{D}} = \text{True}$  or  $\llbracket \varphi_2 \rrbracket_{\sigma}^{\mathcal{D}} = \text{True}$

$\llbracket \exists x \varphi \rrbracket_{\sigma}^{\mathcal{D}} := \text{True}$  iff  $\llbracket \varphi \rrbracket_{\sigma'}^{\mathcal{D}} = \text{True}$  for some  $\sigma'$  coinciding with  $\sigma$  except maybe for  $x$

# Semantics of First-Order Logic

- The meaning of formulas built with the other logical symbols can be defined by reduction to the previous symbols.

$$\llbracket \varphi_1 \wedge \varphi_2 \rrbracket_{\sigma}^{\mathcal{D}} := \llbracket \neg(\neg\varphi_1 \vee \neg\varphi_2) \rrbracket_{\sigma}^{\mathcal{D}}$$

$$\llbracket \varphi_1 \Rightarrow \varphi_2 \rrbracket_{\sigma}^{\mathcal{D}} := \llbracket \neg\varphi_1 \vee \varphi_2 \rrbracket_{\sigma}^{\mathcal{D}}$$

$$\llbracket \varphi_1 \Leftrightarrow \varphi_2 \rrbracket_{\sigma}^{\mathcal{D}} := \llbracket (\varphi_1 \Rightarrow \varphi_2) \wedge (\varphi_2 \Rightarrow \varphi_1) \rrbracket_{\sigma}^{\mathcal{D}}$$

$$\llbracket \forall x \varphi \rrbracket_{\sigma}^{\mathcal{D}} := \llbracket \neg \exists x \neg \varphi \rrbracket_{\sigma}^{\mathcal{D}}$$

- If a sentence is closed (no free variables), its meaning *does not depend* on the the variable assignment (although it may depend on the domain):

$$\llbracket \forall x \exists y R(x, y) \rrbracket_{\sigma}^{\mathcal{D}} = \llbracket \forall x \exists y R(x, y) \rrbracket_{\sigma'}^{\mathcal{D}}, \quad \text{for any } \sigma, \sigma'$$

# Models, Validity, etc. for Sentences

- An interpretation  $(\mathcal{D}, \sigma)$  **satisfies** a sentence  $\varphi$ , or is a **model** for  $\varphi$ , if  $\llbracket \varphi \rrbracket_{\sigma}^{\mathcal{D}} = \text{True}$ .
- A sentence is **satisfiable** if it has at least one model.  
*Examples:*  $\forall x x \geq y, P(x)$
- A sentence is **unsatisfiable** if it has no models.  
*Examples:*  $P(x) \wedge \neg P(x), \neg(x = x)$
- A sentence  $\varphi$  is **valid** if every interpretation is a model for  $\varphi$ .  
*Examples:*  $P(x) \Rightarrow P(x), x = x$
- $\varphi$  is valid/unsatisfiable iff  $\neg\varphi$  is unsatisfiable/valid.
- Valid sentences do not tell us anything about the world.  
(They are always true!)

# Models, Validity, etc. for Sets of Sentences

- An interpretation  $(\mathcal{D}, \sigma)$  **satisfies** a set  $\Gamma$  of sentences, or is a **model** for  $\Gamma$ , if it is a model for *every* sentence in  $\Gamma$ .
- A set  $\Gamma$  of sentences is **satisfiable** if it has at least one model.

$$\text{Ex: } \{\forall x x \geq 0, \forall x x + 1 > x\}$$

- $\Gamma$  is **unsatisfiable**, or **inconsistent**, if it has no models.

$$\text{Ex: } \{P(x), \neg P(x)\}$$

- As in Propositional Logic,  $\Gamma$  **entails** a sentence  $\varphi$  ( $\Gamma \models \varphi$ ), if every model of  $\Gamma$  is also a model of  $\varphi$ .

$$\text{Ex: } \{\forall x P(x) \Rightarrow Q(x), P(A_{10})\} \models Q(A_{10})$$

- **Note:** Again,  $\Gamma \models \varphi$  iff  $\Gamma \wedge \neg\varphi$  is unsatisfiable.

# Possible Interpretations Semantics

- Sentences can be seen as *constraints* on the set  $S$  of all possible interpretations.
- A sentence *denotes* all the possible interpretations that satisfy it (the models of  $\varphi$ ).  
If  $\varphi_1$  denotes a set of interpretations  $S_1$  and  $\varphi_2$  denotes a set  $S_2$ , then
  - $\varphi_1 \vee \varphi_2$  denotes  $S_1 \cup S_2$ ,
  - $\varphi_1 \wedge \varphi_2$  denotes  $S_1 \cap S_2$ ,
  - $\neg\varphi_1$  denotes  $S \setminus S_1$ ,
  - $\varphi_1 \models \varphi_2$  iff  $S_1 \subseteq S_2$ .
- A sentence denotes either no interpretations or an infinite number of them!