The Language of First-Order Predicate Logic (FOPL)

(Note: First-Order Predicate Logic differs from ordinary Predicate Logic in that it contains individual variables and quantifiers. The designation "first-order" reflects the fact that our variables only range over individuals (i.e., the possible denotations for individual constants). A "second-order" logic is one that also contains variables ranging over sets of individuals, sets of ordered pairs of individuals, sets of ordered triplets of individuals, etc. (i.e., the possible denotations for predicate constants).)

Vocabulary (list of basic expressions)

(i) predicate constants: GREEK, MAN, ... (one-place)
BITE, FATHER, ... (two-place)
GIVE, BETWEEN, ... (three-place)

(ii) individual constants: a, b, c, d, e, f, ...

(iii) individual variables: x_1, x_2, x_3, x_4, ...

Together, the individual constants of FOPL and the individual variables of FOPL constitute the **terms** of FOPL.

(iv) connectives: ~ (negation)
& (conjunction), ∨ (disjunction), → (matieral implication)

(v) quantifiers: ∀ (universal, read as 'for all/every individual...')
∃ (existential, read as 'there is/exists an individual ...')

(vi) parentheses: (,)

Syntax (rules for forming grammatical sentences, or "formulas")

(i) If P is an *n*-place predicate constant and t_1, t_2, ... , t_n are *n* terms, then $P(t_1, t_2, \dots, t_n)$ is a formula of PredL.

(ii) If *A* is a formula of FOPL, then so is ~*A*.

(iii) If *A* and *B* are formulas of FOPL, then so are (*A* & *B*), (*A* ∨ *B*), and (*A* → *B*).

(iv) If *A* is a formula of FOPL, then so are $\forall x_n A$ and $\exists x_n A$, for any individual variable x_n.

(v) Nothing else is a FOPL formula.

(Note: typically, we omit the outermost pair of parentheses in a FOPL formula. But all other parentheses are necessary to avoid any potential ambiguity.)

Semantics (rules that assign truth conditions to FOPL formulas)

Two-step procedure for assigning truth conditions to FOPL formulas:

(A) Specify denotations for individual/predicate constants and individual variables by providing a **model** and an **assignment function**.

A **model** *M* consists of: (i) a set *D* of individuals (the "inhabitants" of *M*), and (ii) a "valuation function" *Val*, which specifies a denotation, or semantic value, for each individual/predicate constant in FOPL.

An **assignment function** *g* associates each individual variable in FOPL with member of *D* (an inhabitant of our model *M*).

We also give ourselves a handy means of referring to the **denotation of a term t** relative to a model *M* and an assignment function *g*:

$$[\![\text{t}]\!]^{M,g} = Val(\text{t}) \text{ if t is an individual constant}$$
$$= g(\text{t}) \text{ if t is an individual variable}$$

(B) Show how the truth conditions of a FOPL formula depend upon the denotations of the vocabulary items that appear within it.

(i) If P is a one-place predicate constant and t is term, then P(t) is true relative to a model *M* and an assignment function *g* if $[\![\text{t}]\!]^{M,g} \in Val(\text{P})$. Otherwise, P(t) is false relative to *M* and *g*.

(ii) If P is a two-place predicate constant and t_1, t_2 are terms, then $P(t_1, t_2)$ is true relative to *M* and *g* if $<[\![t_1]\!]^{M,g}, [\![t_2]\!]^{M,g}> \in Val(\text{P})$. Otherwise, $P(t_1, t_2)$ is false relative to *M* and *g*.

(iii) If P is a three-place predicate constant and t_1, t_2, t_3 are terms, then $P(t_1, t_2, t_3)$ is true relative to *M* and *g* if $<[\![t_1]\!]^{M,g}, [\![t_2]\!]^{M,g}, [\![t_3]\!]^{M,g}> \in Val(\text{P})$. Otherwise, $P(t_1, t_2, t_3)$ is false relative to *M* and *g*.

(read on for rules (iv) and (v), which deal with formulas involving ∀ and ∃ …)

(vi) The truth conditions for complex formulas constructed with ~, &, ∨, and → are given by our familiar truth tables:

A	**~A**
T	**F**
F	**T**

A	B	**(A & B)**	**(A ∨ B)**	**(A → B)**
T	T	**T**	**T**	**T**
T	F	**F**	**T**	**F**
F	T	**F**	**T**	**T**
F	F	**F**	**F**	**T**

How do we determine whether the universally quantified formula $\forall x_3$GREEK(x_3) is true or false relative to the following model *M* and assignment function *g*?

M: D = { Dexter, Rita, Frank, Maria, Fido }

Val(m) = Maria	*Val*(f) = Fido	*Val*(d) = Dexter
Val(MAN) = { Dexter, Frank }	*Val*(WOMAN) = { Rita, Maria }	
Val(DOG) = { Fido }	*Val*(GREEK) = { Rita, Maria, Dexter }	
Val(BITE) = { <Fido, Dexter> , <Fido, Charlie> }		

$$g: \begin{pmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{pmatrix}$$

	$\forall x_3$	GREEK(x_3)
	↑	↑
TRUE if…	every way of assigning a denotation to x_3…	makes this formula true

$$g[x_3 \mapsto \text{Dexter}]: \begin{pmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ \mathbf{x_3 \mapsto Dexter} \\ \ldots \end{pmatrix}$$

Is GREEK(x_3) true rel. to *M* and $g[x_3 \mapsto \text{Dexter}]$?

$$g[x_3 \mapsto \text{Rita}]: \begin{pmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ \mathbf{x_3 \mapsto Rita} \\ \ldots \end{pmatrix}$$

Is GREEK(x_3) true rel. to *M* and $g[x_3 \mapsto \text{Rita}]$?

$$g[x_3 \mapsto \text{Frank}]: \begin{pmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ \mathbf{x_3 \mapsto Frank} \\ \ldots \end{pmatrix}$$

Is GREEK(x_3) true rel. to *M* and $g[x_3 \mapsto \text{Frank}]$?

$$g[x_3 \mapsto \text{Maria}]: \begin{pmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ \mathbf{x_3 \mapsto Maria} \\ \ldots \end{pmatrix}$$

Is GREEK(x_3) true rel. to *M* and $g[x_3 \mapsto \text{Maria}]$?

$$g[x_3 \mapsto \text{Fido}]: \begin{pmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ \mathbf{x_3 \mapsto Fido} \\ \ldots \end{pmatrix}$$

Is GREEK(x_3) true rel. to *M* and $g[x_3 \mapsto \text{Fido}]$?

Conclusion: is $\forall x_3$GREEK(x_3) true relative to *M* and *g*?

How do we determine whether the existentially quantified formula
$\exists x_1 \text{GREEK}(x_1)$ is true or false relative to M and g?

M: D = { Dexter, Rita, Frank, Maria, Fido }

Val(m) = Maria *Val*(f) = Fido *Val*(d) = Dexter

Val(MAN) = { Dexter, Frank } *Val*(WOMAN) = { Rita, Maria }

Val(DOG) = { Fido } *Val*(GREEK) = { Rita, Maria, Dexter }

Val(BITE) = { <Fido, Dexter> , <Fido, Charlie> }

g: $\begin{bmatrix} x_1 \mapsto \text{Frank} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{bmatrix}$

	$\exists x_1$	GREEK(x_1)
	↑	↑
TRUE if…	there is at least one way to assign a denotation to x_1…	that makes this formula true

$g[x_1 \mapsto \text{Dexter}]$: $\begin{bmatrix} \mathbf{x_1 \mapsto Dexter} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{bmatrix}$ Is GREEK(x_1) true rel. to M and $g[x_1 \mapsto \text{Dexter}]$?

$g[x_1 \mapsto \text{Rita}]$: $\begin{bmatrix} \mathbf{x_1 \mapsto Rita} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{bmatrix}$ Is GREEK(x_1) true rel. to M and $g[x_1 \mapsto \text{Rita}]$?

$g[x_1 \mapsto \text{Frank}]$: $\begin{bmatrix} \mathbf{x_1 \mapsto Frank} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{bmatrix}$ Is GREEK(x_1) true rel. to M and $g[x_1 \mapsto \text{Frank}]$?

$g[x_1 \mapsto \text{Maria}]$: $\begin{bmatrix} \mathbf{x_1 \mapsto Maria} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{bmatrix}$ Is GREEK(x_1) true rel. to M and $g[x_1 \mapsto \text{Maria}]$?

$g[x_1 \mapsto \text{Fido}]$: $\begin{bmatrix} \mathbf{x_1 \mapsto Fido} \\ x_2 \mapsto \text{Rita} \\ x_3 \mapsto \text{Maria} \\ \ldots \end{bmatrix}$ Is GREEK(x_1) true rel. to M and $g[x_1 \mapsto \text{Fido}]$?

Conclusion: is $\exists x_1 \text{GREEK}(x_1)$ true relative to M and g?

Semantics (rules that assign truth conditions to FOPL formulas, cont'd)

Rules for assigning truth conditions to quantified formulas involving $\forall$ and $\exists$:

(iv) If *A* is a FOPL formula and x_n is an individual variable, then $\forall x_n A$ is true relative to *M* and *g* if for each individual *d* in *D*, the formula *A* is true relative to *M* and $g[x_n \mapsto d]$. Otherwise, $\forall x_n A$ is false relative to *M* and *g*.

(v) If *A* is a FOPL formula and x_n is an individual variable, then $\exists x_n A$ is true relative to *M* and *g* if for at least one individual *d* in *D*, the formula *A* is true relative to *M* and $g[x_n \mapsto d]$. Otherwise, $\exists x_n A$ is false relative to *M* and *g*.

The **modified assignment function** $g[x_n \mapsto d]$ is just like *g*, except that it associates the variable x_n with the individual *d*.