CS 4100: Introduction to AI

Wayne Snyder
Northeastern University

Lecture 4: Introduction to First-Order Logic

English	**FOL**
Bob is hungry	hungry(Bob)
Socrates is a man	is-a(Socrates, Man)
Man is mortal	mortal(Man)

First-Order Logic: Motivations

Propositional logic is very limited in its ability to describe the world:

A formula with N symbols (A_1, A_2, ..., A_N) can describe at most 2^N states of the world:

P	Q	R	$P \to Q$	$Q \to R$	$(P \to Q) \wedge (Q \to R)$
T	T	T	T	T	T
T	T	F	T	F	F
T	F	T	F	T	F
T	F	F	F	T	F
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	T	T	T
F	F	F	T	T	T

$2^3 = 8$

2^{2^3}

2^8

256

T/F

Such a formula can expression logical relationships, but only between propositions which are True or False:

Socrates is a man
All men are mortal
Socrates is mortal

Socrates => Man
Man => Mortal
Socrates => Mortal

$A \wedge B$

A B

Note: This is not all bad, after all, computers operate by Boolean logic circuits, and the cornerstone of modern computer theory, the theory of NP-Completeness, is based on the Satisfiability Problem.

3SAT

2^N

First-Order Logic: Motivations

Socrates is a man
All men are mortal
Socrates is mortal

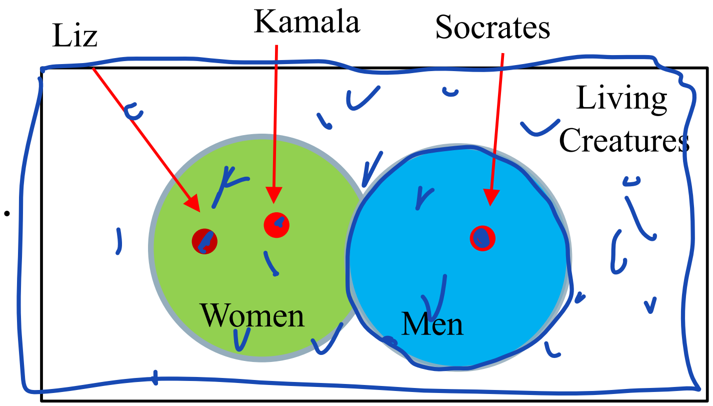
Socrates => Man
Man => Mortal
Socrates => Mortal

If we want to mention several different men, we would have to introduce a separate proposition for each one:

Pericles => Man   Plato => Man   Herodotus => Man ....

We can not speak of a collection of individuals and make assertions about them:

All living creatures are mortal
All women are living creatures.
Kamala is a woman
Liz is a woman
Liz and Kamala are mortal.



First-Order Logic: Motivation

Most kinds of basic mathematical reasoning can not be done without referring to individuals in a collection, and making assertions about their properties and relationships to other individuals.

Example:

Theorem 1. *If a and b are consecutive integers, then the sum $a+b$ is odd.*

Proof. Assume that a and b are consecutive integers. Because a and b are consecutive we know that $b = a + 1$. Thus, the sum $a + b$ may be re-written as $2a + 1$. Thus, there exists a number k such that $a + b = 2k + 1$ so the sum $a + b$ is odd. $\square$

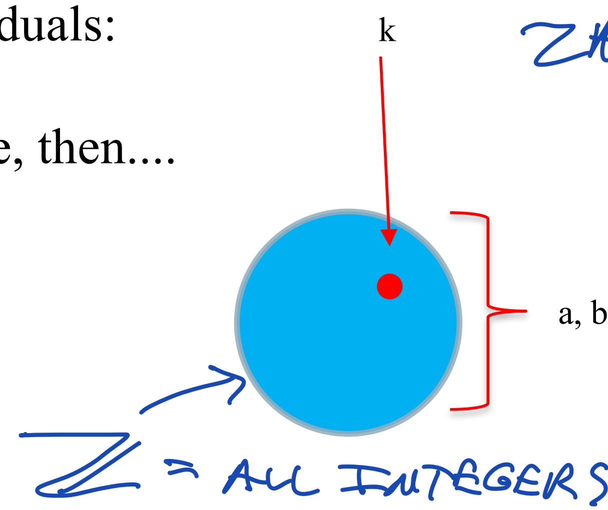
This implicitly makes two kinds of reference to individuals:

ZAHLEN

$\forall$

For all integers a and b, if a and b are consecutive, then....

... there exists a number k .....



First-Order Logic: Syntax

First-Order Logic adds the following features to the basic syntax of propositional logic:

≈ PROP. SYMBOLS

- A Universe of Discourse of individuals we wish to describe;
- Functions, constants, and predicates (T/F assertions) on these individuals;
- Quantifiers expressing "for all individuals" and "there exists an individual";

$\forall x$ $\exists x$

Two important classes of predicates are

- Equality, which asserts that two individuals are the same;
- Relations, which assert some connection or relationship between individuals.
- SETS

First-Order Logic: Syntax

NESTED FUNCTION CALLS

DEF F(X,Y): RETURN X+Y

Definition 3.1 Let V be a set of variables, K a set of constants, and F a set of function symbols. The sets V, K and F are pairwise disjoint. We define the set of *terms* recursively:

- All variables and constants are (atomic) terms.
- If $t_1, \dots, t_n$ are terms and f an n-place function symbol, then $f(t_1, \dots, t_n)$ is also a term.

$f(x)$

$x+y$

2^N

$f(g(a), x)$

f $+$ $a = 0$

g succ

$(0+1)+x$

A TERM REFERS TO AN INDIVIDUAL.

COULD BE

CONSTANT (NAMES OF INDV.S) $\emptyset$ 1 ~MAXINT

USUALLY, FUNCTIONS ARE WRITTEN PREFIX

VARIABLES (REFER TO INDIVIDUALS, CAN CHANGE

FUNCTIONS APPLIED TO TERMS

$+(x, y)$
$=$
$x+y$

2^N

$f(a, x)$

PLUS$(\emptyset, x)$

First-Order Logic: Syntax

$3 + 2 * X$

$3 + (2 * X)$

$+(3, *(2, X))$

(PLUS . . .

(+ ( ) ( ))

ALL FUNCTIONS ARE TOTAL, I.E., DEFINED ON ALL INPUTS

$\mathbb{Z}$

ZAHLEN

$f : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$

— . .

*

NULL

TREE(X, Y)

TREE(TREE(2,3), NULL)

CONSTRUCTOR

CONS(X, Y)   NIL

$\boxed{\text{CONS(2, CONS(1, NIL))}}$

2 3

2:[1] [2, 1]

First-Order Logic: Syntax

P(x) Q(x, y)

PREDICATE SYMBOL ≈ PROPOSITIONAL SYMBOL

Definition 3.2 Let P be a set of predicate symbols. *Predicate logic formulas* are built as follows:

- If $t_1, \ldots, t_n$ are terms and p an n-place predicate symbol, then $p(t_1, \ldots, t_n)$ is an (atomic) formula.
- If A and B are formulas, then $\neg A$, (A), $A \wedge B$, $A \vee B$, $A \Rightarrow B$, $A \Leftrightarrow B$ are also formulas.
- If x is a variable and A a formula, then $\forall x\, A$ and $\exists x\, A$ are also formulas. $\forall$ is the universal quantifier and $\exists$ the existential quantifier.
- $p(t_1, \ldots, t_n)$ and $\neg p(t_1, \ldots, t_n)$ are called literals.

A, B, C, ... ≈

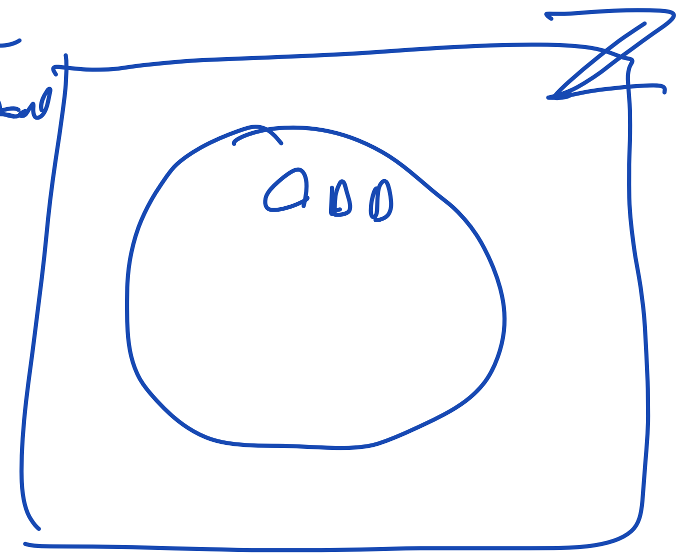
PREDICATES TRUE OR FALSE

UNARY PREDICATE ≈ SET

BINARY PREDICATE ≈ BINARY RELATION

< >=

P(X, Y, Z) BETWEEN(2, 3, 7)



First-Order Logic: Syntax

PROP LOGIC

- SYMBOLS
- ¬ SYMBOLS

LITERALS

FOL

- ATOMIC FORMULA (= PREDICATE)
- ¬ ATOMIC FORMULA

LITERALS

$\forall x\, (\mathrm{Odd}(x) \lor \mathrm{Even}(x))$

$\forall x\, (\ldots x \ldots x \ldots x \ldots)$

SCOPE OF x

$\exists x (\ldots x \ldots x \ldots)$

x

$\mathrm{Odd}(x) \lor \mathrm{Even}(x)$

$\mathrm{Odd}(x) \Leftrightarrow \mathrm{Even}(x+1)$

$\mathrm{Even}(x) \Leftrightarrow \mathrm{Odd}(x-1)$

$\neg \mathrm{Odd}(x) \lor \mathrm{Even}(x+1)$

$\mathrm{Succ}(x) = x+1$

BOUND VARIABLE

$\mathrm{Succ} = \lambda x\, (x+1)$

$\mathrm{Succ} = (\text{LAMBDA } x : x+1)$

$\mathrm{Succ}(3)$

DEF isTautology(C):

C = ∅

C ...

SCOPE OF A

SCOPE OF LOCAL VARIABLE C

DEF

$(x+1)\ [x \mapsto 3]$

First-Order Logic: Syntax

- Formulas in which every variable is in the scope of a quantifier are called *first-order sentences* or *closed formulas*. Variables which are not in the scope of a quantifier are called *free variables*.
- Definitions 2.8 (CNF) and 2.10 (Horn clauses) hold for formulas of predicate logic literals analogously.

FREE VARIABLE IS NOT IN SCOPE OF $\forall, \exists$.

First-Order Logic: Syntax

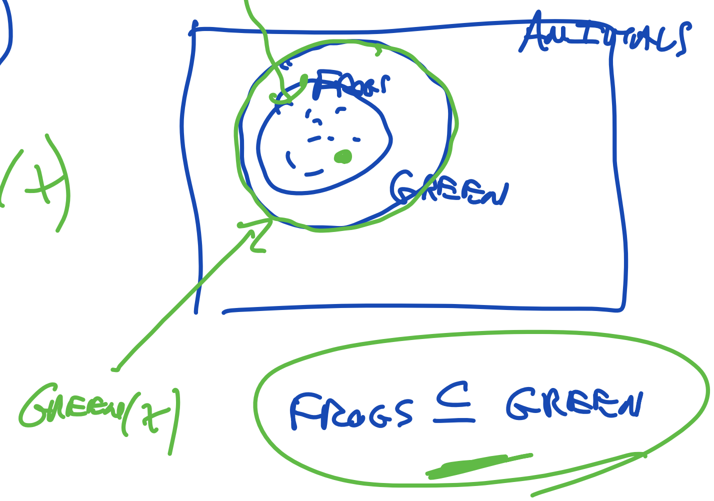
Examples of FOL formulae and what they express. (see p.41 of textbook)

ALL FROGS ARE GREEN.

$\forall x \; \text{FROG}(x) \wedge \text{GREEN}(x)$

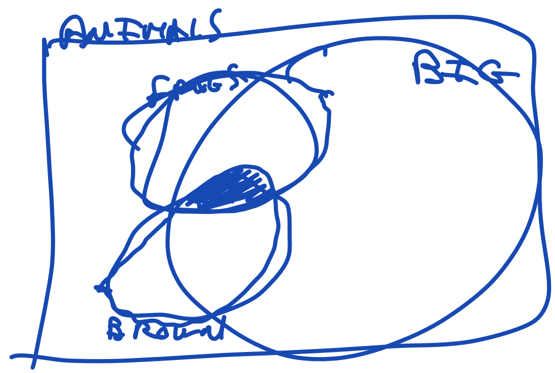
$\forall x \; \text{FROG}(x) \Rightarrow \text{GREEN}(x)$

FROGS(x) $\quad \forall x \quad \exists x$



ALL BROWN FROGS ARE BIG.

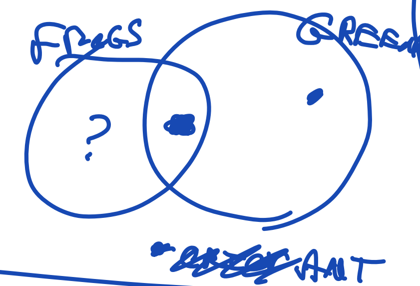
$\forall x. \; \text{BROWN}(x) \wedge \text{FROG}(x)$
$\Rightarrow \text{BIG}(x)$



First-Order Logic: Syntax

SOME FROGS ARE GREEN

$$\boxed{\exists x \; \text{FROG}(x) \Rightarrow \text{GREEN}(x)}$$



✓ $$\exists x \; \text{FROG}(x) \wedge \text{GREEN}(x)$$

GIVE ME 1 EXAMPLE

$$\forall x \; \text{FROG}(x) \rightarrow \text{GREEN}(x)$$

$$\exists x \left(\neg \text{FROG}(x) \vee \text{GREEN}(x) \right)$$

R IS BINARY RELATION $\quad R \subseteq \mathbb{Z} \times \mathbb{Z}$

$$\forall x \forall y . \; R(x,y) \wedge R(y,z) \Rightarrow R(x,z)$$ TRANSITIVITY

$$\forall x \forall y . \; Q(x,y) \Leftrightarrow Q(y,x)$$ SYMMETRY

First-Order Logic: Syntax

Theorem 1. *If a and b are consecutive integers, then the sum $a + b$ is odd.*

Proof. Assume that a and b are consecutive integers. Because a and b are consecutive we know that $b = a + 1$. Thus, the sum $a + b$ may be re-written as $2a + 1$. Thus, there exists a number k such that $a + b = 2k + 1$ so the sum $a + b$ is odd. □

$\text{Consecutive}(x, y) \Leftrightarrow y = x + 1$

$\text{Odd}(x) \Leftrightarrow \exists y.\ x = 2y + 1$

$\forall (a, b)\ \text{Consecutive}(a, b)$

$\Leftrightarrow b = a + 1$

$\Leftrightarrow a + b = a + (a + 1)$

$\Leftrightarrow a + b = 2a + 1$

$\Rightarrow \exists k.\ a + b = 2k + 1$

$\Leftrightarrow \text{Odd}(a + b)$

$\forall x \forall y\ F \equiv \forall y \forall x\ F$

$\exists x \exists y\ F \equiv \exists y \exists x\ F$

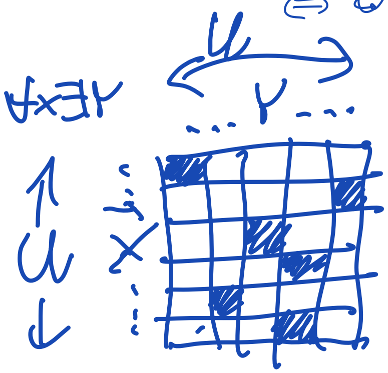
$\forall x \exists y\ F \not\equiv \exists y \forall x\ F$

$\mathbb{Z}$

$(\forall x \exists y.\ y < x)$

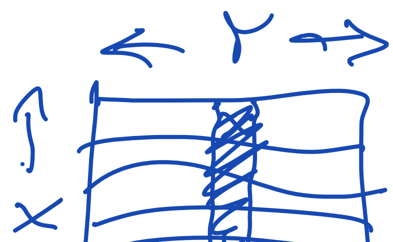
$(\exists y \forall x.\ y < x)$

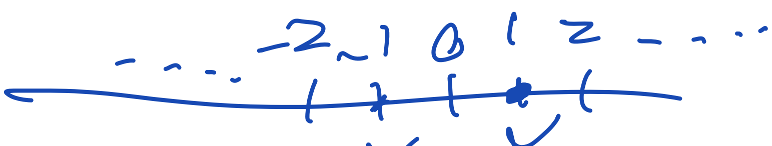
$\forall x \exists y$

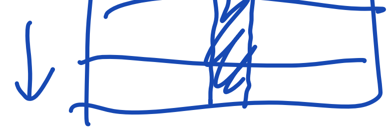


$\exists x, y, z \ldots$

$\exists y \forall x$







First-Order Logic: Semantics

In FOL, an interpretation maps the syntax to the semantics.

Definition 3.3 An *interpretation* $\mathbb{I}$ is defined as

- A mapping from the set of constants and variables $K \cup V$ to a set W of names of objects in the world.
- A mapping from the set of function symbols to the set of functions in the world. Every n-place function symbol is assigned an n-place function.
- A mapping from the set of predicate symbols to the set of relations in the world. Every n-place predicate symbol is assigned an n-place relation.

$I(c)$ = OBJECT IN $\mathcal{U}$

$f(x)$

$I(f)$ = A FUNCTION

x	$f(x)$
-2	-1
-1	0
0	1
1	2
2	

← SUCCESSOR

$P(x,y,z) \subseteq \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}$

First-Order Logic: Semantics

First-Order Logic: Semantics

$\mathbb{I}(f) = \lambda x. x+1$

$\mathbb{I}(\emptyset) = \emptyset$

$\mathbb{I}(p) = <$

$p(\emptyset, f(\emptyset))$

$\emptyset < \cancel{\emptyset}\,\emptyset + 1$

$\neg p(f(f(0)), f(0))$

Definition 3.4

- An atomic formula $p(t_1, \ldots, t_n)$ is *true* (or valid) under the interpretation $\mathbb{I}$ if, after interpretation and evaluation of all terms $t_1, \ldots, t_n$ and interpretation of the predicate p through the n-place relation r, it holds that

$$(\mathbb{I}(t_1), \ldots, \mathbb{I}(t_n)) \in r.$$

$\Rightarrow \quad \Leftrightarrow$

- The truth of quantifierless formulas follows from the truth of atomic formulas—as in propositional calculus—through the semantics of the logical operators defined in Table 2.1 on page 25.
- A formula $\forall x\ F$ is true under the interpretation $\mathbb{I}$ exactly when it is true given an arbitrary change of the interpretation for the variable x (and only for x)
- A formula $\exists x\ F$ is true under the interpretation $\mathbb{I}$ exactly when there is an interpretation for x which makes the formula true.

The definitions of semantic equivalence of formulas, for the concepts satisfiable, true, unsatisfiable, and model, along with semantic entailment (Definitions 2.4, 2.5, 2.6) carry over unchanged from propositional calculus to predicate logic.

$\forall x.\ \neg p(f(f(x)), f(x))$

$\forall x\ F$ IS TRUE UNDER $\mathbb{I}$

ANY SUBSTITUTION OF $u \in \mathcal{U}$ FOR X IN F MAKES F TRUE

$\boxed{\forall x.\ x+2 \not< x+1}$

$\exists x F$ SAME SUBSTITION OF u FOR x

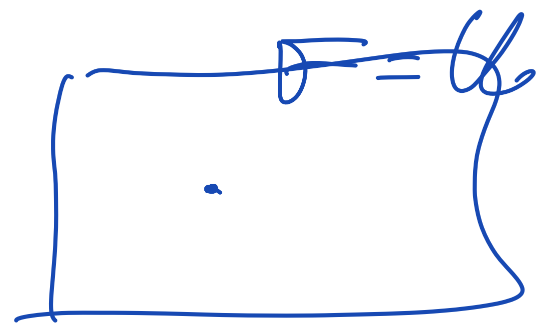
First-Order Logic: Semantics

RULES FOR EQUIVALENCE

$$\forall x \, F(x) \equiv \neg \exists x \, \neg F(x)$$

$$\underline{\exists x \, F(x)} \equiv \neg \forall x \, \neg F$$

$\neg \exists$



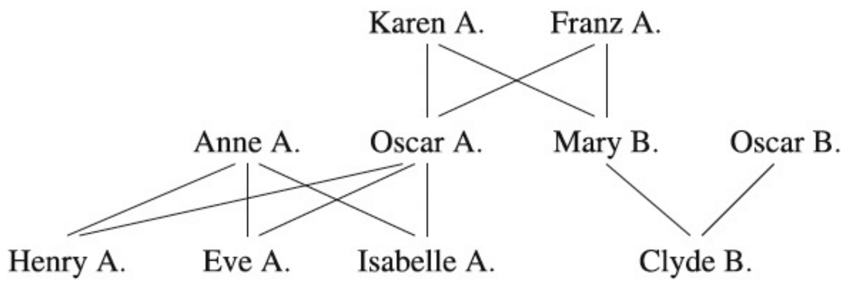
$\neg (A \lor B)$

$(\neg A \land \neg B)$

$$\neg \exists x \forall y . \big(P(x,y) \lor Q(x)\big)$$

$$\forall x \exists y \; \neg P(x,y) \land \neg Q(x)$$

First-Order Logic: Semantics



$$
\begin{aligned}
KB \equiv\ & female(karen)\ \wedge female(anne)\ \wedge female(mary) \\
& \wedge female(eve)\ \wedge female(isabelle) \\
& \wedge child(oscar, karen, franz)\ \wedge child(mary, karen, franz) \\
& \wedge child(eve, anne, oscar)\ \wedge child(henry, anne, oscar) \\
& \wedge child(isabelle, anne, oscar)\ \wedge child(clyde, mary, oscarb) \\
& \wedge (\forall x\, \forall y\, \forall z\, child(x, y, z)\ \Rightarrow\ child(x, z, y)) \\
& \wedge (\forall x\, \forall y\, descendant(x, y)\ \Leftrightarrow\ \exists z\, child(x, y, z) \\
& \vee (\exists u\, \exists v\, child(x, u, v)\ \wedge descendant(u, y))).
\end{aligned}
$$

First-Order Logic: Semantics

First-Order Logic: Semantics

First-Order Logic: Semantics

Equivalence of formulae in FOL

First-Order Logic: Semantics

Equality is a special case of a relation which is always given the natural interpretation, using the axioms:

$$
\begin{array}{rll}
\forall x & x = x & \text{(reflexivity)} \\
\forall x \, \forall y & x = y \Rightarrow y = x & \text{(symmetry)} \\
\forall x \, \forall y \, \forall z & x = y \wedge y = z \Rightarrow x = z & \text{(transitivity).}
\end{array}
\qquad (3.1)
$$

$$\forall x \, \forall y \; x = y \Rightarrow p(x) \Leftrightarrow p(y) \quad \text{(substitution axiom).}$$

$$\forall x \, \forall y \; x = y \Rightarrow f(x) = f(y) \quad \text{(substitution axiom)}$$

First-Order Logic: Semantics

First-Order Logic: Semantics