# Introduction to Logic 13

**Last time:**

- Propositional Logic
- Limitations of Propositional Logic
- The Structure of Propositions
- Individuals, Properties and Quantifiers

**This time:**

- The Language of the Predicate Calculus
  - Basic Expressions
  - Terms and Formulas
- Expressing Propositions
- Semantic Preliminaries

# The Language of FOPC

• The basic expressions of Predicate Logic fall into four separate categories:

1. **Individual names:** $a$, $b$, $c$, $\ldots$
   These represent specific objects, persons or events

2. **Individual variables:** $x$, $y$, $z$, $\ldots$
   These are variables that range over individuals.

3. **Predicate Symbols:** $P$, $Q$, $R$, $\ldots$
   Predicate symbols are used to represent properties or relations over individuals

4. **Function Symbols:** $f$, $g$, $h$, $\ldots$
   Function symbols denote functions that map individuals to individuals.

# A Note about Predicates and Functions

- **Note:** The predicate and function symbols represent relations over individuals.

  e.g.

  | | |
  |---|---|
  | *Bill* **talks** | 1 individual |
  | *Bill* **likes** *Logic* | 2 individuals |
  | *Bill* **teaches** *Moira Logic* | 3 individuals |

- More generally, we can have relations or functions over an arbitrary number $n$ of individuals

  **Terminology:** An $n$-place predicate or function symbol is said to have **arity** $n$.

- We assume that each predicate or function symbol is associated with a known, fixed arity

- In addition to the four classes of basic expression, the predicate calculus includes:
  - A truth-functionally complete set of connectives: e.g. $\neg, \wedge, \vee, \rightarrow$, and $\leftrightarrow$.
  - Two quantifier symbols:
    **The Universal Quantifier:** $\forall$
    **The Existential Quantifier:** $\exists$
  - Brackets '(' and ')', and punctuation symbols ',' and '.'.
- These symbols, taken together with the basic expressions, form the **alphabet** of the language of First Order Predicate Logic.
- The language is defined in two stages: **terms**, and **formulas**

# Terms

- Terms are used to pick out individuals:

**Definition** *A term $t$ is either:*

1. *an individual name; or*
2. *an individual variable; or*
3. *a* functional term $f(t_1, \ldots, t_n)$ *where $f$ is a function symbol of arity $n$, and $t_1, \ldots, t_n$ are terms*

**Question:** *Which of the following are terms?*

$$a$$
$$y$$
$$P$$
$$f(a, x)$$
$$Q(x, y)$$
$$(f(x, x))$$
$$f(g(x, a), h(b))$$

# Formulas

**Definition** *A* **well-formed formula** *of Predicate Logic is either:*

1. *an* atomic formula $P(t_1, \ldots, t_n)$ *where* $P$ *is a predicate symbol of arity* $n$ *and* $t_1, \ldots, t_n$ *are terms; or*

2. *a* compound formula *of one of the following forms:*

   (a) $(\neg A)$

   (b) $(A \wedge B)$

   (c) $(A \vee B)$

   (d) $(A \rightarrow B)$

   (e) $(A \leftrightarrow B)$

   (f) $\forall v.A$

   (g) $\exists v.A$

   *where* $A$ *and* $B$ *are wffs, and* $v$ *is an individual variable.*

**Question:** *Which of the below are well-formed formulas of Predicate Logic?*

$$P(a)$$

$$(P(a) \rightarrow Q(a))$$

$$P(Q(a))$$

$$(\neg P(f(a, x)))$$

$$\forall x.(P(x) \rightarrow (\neg Q(x)))$$

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

$$(P(a) \vee \exists x.Q(x))$$

$$\exists x.(P(x) \wedge \forall y.(Q(y) \rightarrow R(x, y)))$$

**Notes:**

- We can drop brackets (as for Propositional Logic) by adopting conventions for operator precedence, etc.
- We may relax conventions for naming predicate symbols, individual names etc.

# Representing Propositions

- The FOPC permits finer-grained representation of propositions.

e.g.

*Logic is fun*

$$F(l)$$

*If Logic is fun, then Bill is happy*

$$F(l) \to H(b)$$

*Either Logic is fun or Bill is not happy*

$$F(l) \vee \neg H(b)$$

*All lecturers are happy*

$$\forall x.(L(x) \to H(x))$$

*Some lectures are happy*

$$\exists x.(L(x) \wedge H(x))$$

**Question:** *How might you represent the following?*

*Some lecturer is not happy*

*No lecturer is happy*

*All lecturers teach Logic*

*All lecturers teach some course*

*Every course tutor is happy*

# Semantic Preliminaries

- The semantics of Propositional Logic was particularly simple:
  - **Valuations:** $V : Prop \rightarrow \{t, f\}$
  - Truth tables for connectives:
- For Predicate Logic, the picture is more complicated:
  - four kinds of basic expression;
  - quantification;
  - terms and formulas.
- Meaning is no longer just a matter of **true** and **false** .

- We need a richer semantic domain:
  - individuals – for names
  - functions over individuals – for function symbols
  - relations over individuals - for predicates
- Our semantics should:
  1. map basic expressions onto elements of the semantic domain;
  2. associate individuals with terms; and
  3. give us a way of determining the truth-values of
     - atomic formulas
     - compound formulas (including quantified formulas)

# Summary

- The language of First Order Predicate Logic has an alphabet consisting of four basic kind of expression.
- The language is defined in two stages: first order terms, and first order formulas.
- Predicate Logic provides a richer language for representing propositions.
- We can represent propositions concerned with particular individuals, or capture generalisations about groups of classes of individual.
- Our definition of meaning must be correspondingly rich.