Hamilton College
Russell Marcus
rmarcus1@hamilton.edu

Class 42 - December 10
Second-Order Logic

I. Introducing Predicate Variables

Our last logical language is a controversial extension of predicate logic.
Consider a red apple and a red fire truck.

1. $(\exists x)(Rx \bullet Ax)$
2. $(\exists x)(Rx \bullet Fx)$

We might want to infer that they have something in common, that they share a property.
One natural way to make that inference is to quantify over the predicates themselves.
That is, 3 is easily derived from 1 and 2.

3. $Ra \bullet Rb$

From 3, we might want to produce 4.

4. $(\exists X)(Xa \bullet Xb)$

We have now treated the predicates as subject to quantification, like variables or constants.
A language which allows quantification over predicate places is called a second-order language.
A system of logic which uses a second-order language is called second-order logic.

We have previously allowed all capital letters to be predicate constants.
In our new second-order logic, we are going to reserve 'V', 'W', 'X', 'Y', and 'Z' as predicate variables.
Introducing predicate variables allows us to regiment some new sentences.

5. No two distinct things have all properties in common.
   $(x)(y)[x \neq y \supset (\exists X)(Xx \bullet {\sim}Xy)]$

6. Identical objects share all properties.
   $(x)(y)[x = y \supset (Y)(Yx \equiv Yy)]$

6 is called Leibniz's law.
The converse of Leibniz's law, the controversial claim of the identity of indiscernibles, is written more clearly in second-order logic.

7. $(x)(y)[(Z)(Zx \equiv Zy) \supset x = y]$

The law of the excluded middle is also best regimented in second-order logic, with sentential variables, which you may recall we can take as zero-place predicates.

8. $(X)(X \vee {\sim}X)$

Second-order logic allows us to regiment analogies.

9. Cat is to meow as dog is to bark.
10. (∃X)(Xcm • Xdb)

Actually, 10 is unlikely to be the best regimentation of 9, because of its odd use of constants.
But, it gives us an idea of the power of the second-order quantifiers.

Using the induction schema, in Peano Arithmetic, meant that the theory was not finitely axiomatizable: there are infinitely many instances of the induction schema.
Second-order logic allows us to replace the induction schema with single axiom.

11. (P){{Na • Pa • (x)[(Nx • Px) ⊃ Pf(x)]} ⊃ (x)(Nx ⊃ Px)}

II. Formation rules for S

Vocabulary of S

Capital letters
- A...U, used as predicates
- V, W, X, Y, and Z, used as predicate variables

Lower case letters
- a, b, c, d, e, i, j, k...u are used as constants.
- f, g, and h are used as functors.
- v, w, x, y, z are used as variables.

Five connectives: ~, •, ∨, ⊃ ≡
Quantifier: ∃
Punctuation: (), [], {}

Formation rules for wffs of S.

1. An n-place predicate or predicate variable followed by n terms (constants, variables, or functor terms) is a wff.
2. If α is a wff, so are
   (∃x)α, (∃y)α, (∃z)α, (∃w)α, (∃v)α
   (x)α, (y)α, (z)α, (w)α, (v)α
3. If α is a wff, then the following are wffs
   (∃V)α, (∃W)α, (∃X)α, (∃Y)α, (∃Z)α
   (V)α, (W)α, (X)α, (Y)α, (Z)α
4. If α is a wff, so is ~α.
5. If α and β are wffs, then so are:
   (α · β)
   (α ∨ β)
   (α ⊃ β)
   (α ≡ β)
6. These are the only ways to make wffs.

III. More sample translations

12. Everything has some relation to itself.
    $(x)(\exists V)Vxx$

13. All people have some property in common.
    $(x)(y)[(Px \bullet Py) \supset (\exists Y)(Yx \bullet Yy)]$

14. No two people have every property in common.
    $(x)(y)[(Px \bullet Py) \supset (\exists Z)(Zx \bullet \sim Zy)$

Second-order logic allows us to regiment three important characteristics of relations: reflexivity, symmetry, and transitivity.

A relation is reflexive if every object bears that relation to itself.
Being the same size as something is a reflexive relation.
So is being equidistant from a given point.

15. Reflexivity: $(x)Rxx$

A relation is symmetric if whenever one thing bears that relation to another, the reverse is also true.
Being a sibling is a symmetric relation.
Being older than is asymmetric.

16. Symmetry: $(x)(y)(Rxy \equiv Ryx)$

Lastly, transitivity is exemplified by hypothetical syllogism.
It will be most clearly explained in logical form.

17. Transitivity: $(x)(y)(z)[(Rxy \bullet Ryz) \supset Rxz]$

Being older than, or larger than, or earlier than are all transitive relations.
These three properties of relations are important because they characterize identity.
We call any relation which is reflexive, symmetric, and transitive an equivalence relation.
We can regiment these characteristics of relations without second-order logic, as we did in 15-17.
But, second-order logic allows us to do more.

18. Some relations are transitive.
    $(\exists X)(x)(y)(z)[(Xxy \bullet Xyz) \supset Xxz]$

19. Some relations are symmetric, while some are asymmetric.
    $(\exists X)(x)(y)(Xxy \supset Xyx) \bullet (\exists X)(x)(y)(Xxy \supset \sim Xyx)$

The additional power of second-order logic also entails that we need not reserve a special identity predicate.
Instead, we can just introduce it as shorthand for the following second-order claim:

20. $x=y$ iff $(X)(Xx \equiv Xy)$

IV. Higher-Order Logics

Second-order logic is only one of the higher-order logics.
All logics after first-order logic are called higher-order logic
To create third-order logic, we introduce attributes of attributes, for which I will use boldfaced italics.

21. All useful properties are desirable.
$$(X)(\boldsymbol{U}X \supset \boldsymbol{D}X)$$

22. A man who possesses all virtues is a virtuous man, but there are virtuous men who do not possess all virtues:
$$(x)\{[Mx \bullet (X)(\boldsymbol{V}X \supset Xx)] \supset \boldsymbol{V}x\} \bullet (\exists x)[Mx \bullet Vx \bullet (\exists X)(\boldsymbol{V}X \bullet {\sim}Xx)]$$

21 and 22 are technically ill-formed.
Both predicate variables in 21 are missing objects.
In 22, the third-order variables are applied both to predicates and terms, which is a category error.
Still, they are clear enough to give us the idea of third-order variables, and proper regimentations would obscure the key idea.
We won't spend much time on higher-order logics, so I won't trouble to get these right.

The only higher-order concept that will be useful to us is a version of identity for properties.
That is, we might want to say something like 23.

23. There are at least two distinct properties.

One potential regimentation of 23 is 24.

24. $(\exists X)(\exists Y)X \neq Y$

But 24 is ill-formed, since we have not defined identity for predicates, and since there are no objects attached to the predicates.
We do have the '$\equiv$', which is an equivalence relation among predicates.
Thus, we can translate 23 as 25.

25. $(\exists X)(\exists Y)(\exists x){\sim}(Xx \equiv Yx)$

Of course, 25 only indicates that there are distinct monadic properties.
In order to generalize these claims, higher-order logics are required.

V. Exercises. Translate each of the following sentences into **S**.

1. Jared has some properties, but he lacks some properties.
2. Mike and Nick share no attributes.
3. Some attributes are properties of nothing.
4. Everyone shares some property with Tudor.
5. Gillian shares some attributes with a famous scientist.
6. All philosophers and scientists have properties in common.
7. Reva has at least two different properties.
8. Ron has all of his father's properties.
9. Some relations are both reflexive and symmetric.
10. There is something which lacks all transitive relations.

VI. Second-order logic and set theory

Many philosophers have argued that higher-order logics are not really logic.
Quine takes first-order logic with identity as his canonical language, the privileged language used for expressing his most sincere beliefs and commitments, for a variety of reasons.
Perhaps most influentially, Quine calls second-order logic, "set theory in sheep's clothing" (*Philosophy of Logic*, p 66).
Let's see why.

When we interpret first-order logic, we specify a domain for the variables to range over.
Sometimes we use restricted domains.
If we want to interpret number theory, for example, we restrict our domain to the integers.
If we want to interpret a biological theory, we might restrict our domain to species.
For our most general reasoning, we take an unrestricted domain: the universe, everything there is.
Consider the sentence, 'there are blue hats'.

26. $(\exists x)(Bx. \bullet Hx)$

As we have seen, on standard semantics, for 26 to be true there must exist a thing which will serve as the value of the variable 'x', and which has both the property of being a hat and being blue.
As Quine says, to be is to be the value of a variable.
Our most basic commitments arise from examining the domain of quantification for our best theory of everything.

Now, consider a sentence of second-order logic, 'some properties are shared by two people'.

27. $(\exists X)(\exists x)(\exists y)(Px \bullet Py \bullet x \neq y \bullet Xx \bullet Xy)$

For 27 to be true, there must exist two people, and there must exist a property.
The value of the variable 'X' is not an ordinary object, but a property of an object.
By quantifying over properties, we take properties as kinds of objects; we need some thing to serve as the value of the variable.
We could take the objects which serve as the values of predicate variables to be Platonic forms, or eternal ideas.

But commitments to properties, in addition to the objects which have those properties, is metaphysically contentious.
The first-order sentence about blue hats referred only to an object with properties.
The second-order sentence reifies properties.
Is there really blueness, in addition to blue things?

The least controversial way to understand properties is to take them to be sets of the object which have those properties.
We call this conception of properties extensional.
On an extensional interpretation, 'blueness' refers to the collection of all blue things.
Thus, second-order logic at least commits us to the existence of sets.
We might want to include sets in our ontology if we think there are mathematical objects.
But, we need not include them under the guise of second-order logic.
We can instead take them to be values of first-order variables.
We can count them as among the objects in the universe, in the domain of quantification, rather than sneaking them in through the interpretations of second-order variables.
Quine's complaints about second-order logic are based, in part, on this sneakiness.

In favor of second-order logic, it is difficult to see how one could regiment sentences like 5, 6, 9, 12-14, 18, 19, and 21-23 in first-order logic.
The possibility of deriving the properties of identity from the second-order axioms, rather than introducing a special predicate with special inferential properties, is tempting.

Quine favors using schematic predicate letters in lieu of predicate variables.
I find his approach disingenuous.
With schematic letters, we regiment the law of the excluded middle, for example, as 28, rather than 8, with the understanding that any wff of **F** can be substituted for 'P'.

28. $P \vee {\sim}P$

Schematic letters are really meta-linguistic variables.
What Quine is really admitting is that we can not formulate claims like 8 in our canonical language.
We must, instead, ascend to a meta-language, using meta-linguistic variables.

The debate over second-order logic is worth examining.
Quine's objections to second-order logic are found in his *Philosophy of Logic*, among other places; he has good discussions of schematic letters in his *Methods of Logic*.
Stewart Shapiro makes a compelling case against Quine and for second-order logic in his *Foundations without Foundationalism: A Case for Second-Order Logic*.
(See the "The Right Logic" section of the Course Bibliography for further details.)
The debate over second-order logic is one aspect of a larger question of determining a canonical language, and whether there even is such a best language.

VII. Derivations in higher-order logics

We will not consider derivations in higher-order logics.

VIII. Solutions

1. (∃X)Xj • (∃X)~Xj
2. (X)(Xm ≡ ~Xn)
3. (∃X)(x)~Xx
4. (x)[Px ⊃ (∃X)(Xt • Xx)]
5. (∃x)[(Fx • Sx) • (∃X)(Xg • Xx)]
6. (x)(y)[(Px • Sy) ⊃ (∃X)(Xx • Xy)]
7. (∃X)(∃Y)[Xr • Yr • (∃x)~(Xx≡Yx)]
8. (X)(Xf(r) ⊃ Xr)
9. (∃X)[(x)Xxx • (x)(y)(Xxy ⊃ Xyx)]
10. (∃w)(X){(x)(y)(z)[(Xxy • Xyz) ⊃ Xxz] ⊃ ~Xw}