Logic: lecture 2

Validity: a review

- For our purposes, the following are to be regarded as saying the same thing (here 'P1', 'P2', and 'C' stand for arbitrary sentences):
  - The argument with premises P1, P2 and conclusion C is valid / logically valid / deductively valid.
  - P1 and P2 [logically] entail / imply C
  - C follows [logically] from P1 and P2
  - If P1 and P2 are true, C must be true
  - It can't be the case that P1 and P2 are true without C being true
  - C is true in any circumstance which makes P1 and P2 true

- Of course, arguments can have any number of premises, not just two.
- There's some unclarity about the exact meaning of these expressions. What does 'must' mean, for example? Do we say that if the Atlantic contains water, it "must" contain hydrogen?
  - Fortunately, for our purposes it'll never matter just how this unclarity is resolved.
  - Our primary aim is to get a grip on the way in which words like 'and', 'or', 'not', 'some', 'all'... contribute to validity.

- A sound argument is a valid argument with true premises.
- Review: can a valid argument have...
  - True premises and a true conclusion?
  - True premises and a false conclusion?
  - At least one false premise and a false conclusion?
  - At least one false premise and a true conclusion?

FOL: a basic tool in logic

- English is a messy language in lots of ways.
  - Ambiguity.
  - An immense profusion of ways to say the same thing: e.g. ‘All dogs are animals’, ‘Whenever something is a dog, it is an animal’, ‘The dogs are among the animals’, etc.
  - Complicated syntax: just working out the rules for what counts as a grammatical sentence is the subject matter of a whole branch of linguistics.

- That's why it will be much more convenient to use an artificial language, FOL (eff-oh-ell), or 'the language of first-order logic'.
- FOL is used all the time in mathematics, physics, philosophy, linguistics and computer science.
- It can help you to formulate your thoughts clearly and rigorously in any area.
- And it will help you get a better understanding of your own language.

- Traditionally, FOL has been taught primarily by teaching people how to translate from their own language into it and back.
- But in this course we'll be using something much more like the method of "total immersion", thanks to the computer programs in LPL.

FOL is a family of languages

- What they have in common are certain vocabulary items—which do the same sort of work in FOL that the words 'and', 'or', 'not', 'if...then...', 'some' and 'all' do in English—and their syntax or grammar.
  - although there are some unimportant differences in notation
- The rest of the vocabulary can be different from one dialect of FOL to the next.
- These words are of three kinds: *individual constants*, *predicates*, and (in some dialects) *function symbols*. We'll ignore function symbols for the present.

Individual constants

- They are the equivalent of *names* in English.
  - Each name *refers* to one, and only one, object.
  - Not to more than one or to none, like ‘John Smith’ and ‘Pegasus’ in English!
- They are to be written in *lower-case* (NB!)

Predicates

- Their role is to express a property of objects or a relation between objects.
- Each predicate has a fixed *arity*. A predicate whose arity is 1 is called *unary*; a predicate whose arity is 2 is called *binary*; a predicate whose arity is 3 is called *ternary*.
  - The arity of a predicate is the number of individual constants it needs to be combined with to form a meaningful sentence.
- A predicate begins with an *upper-case* letter (NB!)

Atomic sentences

- The simplest and most basic kind of sentence in FOL.
- An atomic sentence consists of: an *n*-ary predicate, a left parenthesis, a list of *n* individual constants separated by commas, and a right parenthesis, in that order.
  - Happy(john)
  - Loves(anthony, cleopatra)
  - Between(c, f, a)

- An atomic sentence is either *true* or *false*.
  - An atomic sentence with a unary predicate is true if the referent of the individual constant has the property expressed by the predicate, and false if it doesn't.
  - An atomic sentence with an *n*-ary predicate is true if the referents of the individual constant (taken in the specified order) stand in the relation expressed by the predicate, and false if they don't.
  - We say that a true sentence has the "truth-value" TRUE, while a false sentence has the "truth-value" FALSE.
- In fact, this holds for every sentence in FOL.

- What about vagueness? Is the sentence ‘A 5’11” man is tall’ true or false?
  - Some philosophers say it isn’t.
  - If that’s right, we must stipulate that predicates in FOL cannot be vague.

The blocks language

- This is the dialect of FOL used in the program *Tarski's World*; it's designed for talking about the chessboard-like "worlds" you construct in that program.

- Names: a, b, c, d, e, f, n_1, n_2, n_3, ...
- Unary predicates: Cube, Tet, Dodec, Small, Medium, Large
- Binary predicates: Smaller, Larger, LeftOf, RightOf, BackOf, FrontOf, SameSize, SameShape, SameRow, SameCol, Adjoins, =
- Ternary predicate: Between

Using Tarski's World

Problem Set 1

- Read the Introduction, sections 1.1-1.4 and 2.1. Do the **You try it** sections on pages 8 and 24.
- Do exercises 1.2 (15%), 1.3 (25%),1.5 (25%) and 2.2 (35%).
- If you can't get the book or can't get the software to work, send an email to your instructor explaining your situation.