CAS LX 502 Spring 2011
Semantics 1

The Language of Propositional Logic (PropL)

Vocabulary (list of basic expressions)

(i) propositional variables: p, q, r, s, . . .
(ii) connectives: ~ (negation)
&, v, → (conjunction, disjunction, material implication)
(iii) parentheses: (,)

Intended correspondences between PropL vocabulary and English:

- propositional variables correspond to simple sentences

p = *Jim got a raise.*	q = *Pam is pregnant.*
r = *Michael is sick.*	s = *Dwight got a raise.*

- connectives correspond to certain "functional words" in natural language

~	*not*, *it is not the case that*, *it is not true that* (negation)
&	*and* (conjunction)
v	*or* (disjunction)
→	*if ... then* (conditional)

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

(i) Any propositional variable is a formula of PropL.
(These are the "atomic formulas" of PL.)
(ii) If *A* is a formula of PropL, then so is ~*A*.
(iii) If *A* and *B* are formulas of PropL, then so are (*A* & *B*), (*A* v *B*), and (*A* → *B*).
(In (ii) and (iii), *A* and *B* are not necessarily atomic formulas.)
(iv) Nothing else is a formula.

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

Semantics (rules that assign truth values (T or F) to formulas)

Two-step procedure for assigning truth values to PropL formulas:

(i) Propositional variables are assigned truth values by a "valuation function".

Val(p) = F
Val(q) = T
Val(r) = T
Val(s) = F
. . .

(ii) The truth value (T or F) of a complex formula is determined by (a) the truth values assigned to the component formulas that appear within it, and (b) the connectives that join these component formulas.

A	~A
T	F
F	T

A	B	A & B
T	T	T
T	F	F
F	T	F
F	F	F

A	B	A v B
T	T	T
T	F	T
F	T	T
F	F	F

A	B	A → B
T	T	T
T	F	F
F	T	T
F	F	T

From these basic truth tables for our logical connectives, we are able to construct a truth table for any arbitrarily complex PropL formula. In doing so, we are showing the exact circumstances in which that complex formula will be true. In other words, the truth table will specify the truth conditions for that complex formula. Our goal when translating an English sentence into PropL is to identify a formula with truth conditions that match the truth conditions of the original English sentence.

An example, using the following basic translation key:

p = *Jim got a raise.* q = *Pam is pregnant.*
r = *Michael is sick.* s = *Dwight got a raise.*

(1) It's not true that Jim or Dwight got a raise.
~(p v s)

p	s	p v s	~(p v s)
T	T	T	F
T	F	T	F
F	T	T	F
F	F	F	T

Do the truth conditions of the PropL formula in (1) match the truth conditions of the English sentence?

- PropL formula: T if *Val*(p) = F and *Val*(s) = F,
  F otherwise
- English sentence: T if Jim didn't get a raise and Dwight didn't get a raise
  F otherwise

Constructing the truth table for an arbitrarily complex PropL formula:

(i) Identify all of the propositional variables, and all possible assignments of truth values to them. In general, for a PropL formula with *n* different propositional variables, there will be 2^n rows in its truth table.

(2) If Pam is pregnant and Michael isn't sick, then Dwight didn't get a raise.
(q & ~r) → ~s

q	r	s
T	T	T
T	T	F
T	F	T
T	F	F
F	T	T
F	T	F
F	F	T
F	F	F

- since the PropL formula in (2) contains three propositional variables, its truth table must contain eight rows

(ii) Create a new column for any successively larger formulas that you encounter, until you've reached the column for the entire formula.

q	r	s	~r	~s	(q & ~r)	(q & ~r) → ~s
T	T	T	F	F	F	T
T	T	F	F	T	F	T
T	F	T	T	F	T	F
T	F	F	T	T	T	T
F	T	T	F	F	F	T
F	T	F	F	T	F	T
F	F	T	T	F	F	T
F	F	F	T	T	F	T

Why do parentheses matter? Compare (2) to (3):

(3) Pam is pregnant, and if Michael isn't sick, then Dwight didn't get a raise.
q & (~r → ~s)

q	r	s	~r	~s	(~r → ~s)	q & (~r → ~s)
T	T	T	F	F	T	T
T	T	F	F	T	T	T
T	F	T	T	F	F	F
T	F	F	T	T	T	T
F	T	T	F	F	T	F
F	T	F	F	T	T	F
F	F	T	T	F	F	F
F	F	F	T	T	T	F