Logical Operators

Math 251

Formal Propositional Logic

Syntax

- set of **propositional variables**: p, q, r, etc. (like legal identifiers)
- set of **logical operators**: $\wedge$, $\vee$, $\neg$, etc.
  - **operator**: input(s) and output are same type of object
    - logical operators have propositions as both input and output
    - different formal logics might allow different lists of operators
  - **arity**: number of inputs to the operator
  - Examples
    - 1-ary (unary): $\neg$
    - 2-ary (binary): $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, etc.
- **syntactic rules** for combining propositions with operators
  - **atomic propositions:** every propositional variable is a proposition
  - T and F are propositions (can think of as 0-ary operators or constants)
  - if p is a proposition and $*$ is unary, then $*p$ is a proposition
  - if p and q are propositions and $*$ is binary op, then $(p * q)$ is a proposition
  - if $p_1, \ldots p_k$ are propositions and $*$ is k-ary, then $*(p_1, p_2, ...p_k)$ is a proposition

Two optional pieces of syntax

- Extra parens allowed
  - if p is a proposition, then (p) is a proposition
- Some parens can be omitted.
  - **Precedence rules** allow us to omit some parens without ambiguity.
  - **Precedence Order**:
    - negation (not): $\neg$
    - conjunction (and): $\wedge$
    - disjunction (or): $\vee$
    - other 2-ary: $\rightarrow$, $\leftrightarrow$, etc.
  - *Better safe than sorry*: use parens to make sure things are clear

Semantics

Truth Assignments

- A declaration of which atomic propositions are true/false is called a **truth assignment**
- To know whether a proposition is true, we only need to know which of the atomic propositions are true – the rest is determined by the semantics of the operators

Truth Tables

- A formal mechanism for determining whether a complex (non-atomic) proposition is TRUE or FALSE given a particular truth assignment.
- In particular, truth tables are used to say precisely what each logical operator "means".
- Note: Informal logic can be ambiguous, but formal logic is unambiguous (given its specification).

p	q	$p \wedge q$	$p \vee q$	$p \rightarrow q$
T	T	T	T	T
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

See the text for additional details about the semantics of common logical operators.

Exercises

1. Use the precedence of logical opertors to rewrite each proposition fully parenthesized:

   a. $\neg p \wedge \neg q \leftrightarrow r$
   b. $\neg p \rightarrow q \wedge r$
   c. $p \vee \neg q \rightarrow r$
   d. $a \wedge b \vee \neg c \rightarrow d \rightarrow e \wedge \neg f$

   (Don't ever write propositions without parens like this.)

2. How many unary operators are there?
3. Build a truth tables for (a) $(p \vee q) \rightarrow p$, (b) $p \vee (q \rightarrow p)$, (c) $(p \vee q) \rightarrow r$.
4. How many rows does a truth table have? (What does the answer depend on?)
5. How many binary logical operators are there?
6. For a proposition of the form $p \rightarrow q$, we have the following three related propositions:

   - converse: $q \rightarrow p$
   - inverse: $\neg p \rightarrow \neg q$
   - contrapositive: $\neg q \rightarrow \neg p$

   Which of the converse, contrapositive, and inverse are equivalent to the original? (What does it mean to be equivalent?)

7. Build a truth table for "not p but q"
8. Build a truth table for "p unless q"
9. Translate the following English sentences into formal propositional logic. (Begin by assigning a propositional variable to each atomic proposition.)

   a. You may have desert if you eat your vegetables.

   b. You may have desert only if you eat your vegetables.

   c. You may have desert if and only if you eat your vegetables.

   d. You may not ride the roller coaster if you are under 4 feet tall unless you are older than 16 years old.