Section 1.1: Propositional logic.

- Definition: A proposition is a declarative sentence that is either true or false, but not both.

  For example: "Jeff brought us food today." is a proposition.

- We can also form what are called compound propositions by using logical operators with existing propositions. Logical operators are things like "not," "and," "or," and "implies."

  For example "Jeff didn't bring us food today and I dropped math 55 is a compound proposition. It uses the "not" operator on the proposition "Jeff brought us food today" and the "and" operator to join that to the proposition "I dropped math 55."

We have symbols for operators. Here are some of them:

operator	not	and	or	implies
symbol	$\neg$	$\wedge$	$\vee$	$\longrightarrow$

If we let p represent "Jeff brought us food today" and then let q represent "I dropped math 55," then the above compound proposition can be expressed: $\neg p \wedge q$.

- One way to understand how operators behave is by making truth tables. A truth table lists all possible combinations of truth values of variables in a compound proposition, and then uses the rules of logical operators to find the truth value of the overall compound proposition.

  Example: Here's the truth table for some basic operators:

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

It's a good idea to come up with your own examples to make sure the values of this truth table make sense.

For example: let p = "It is raining" and q = "The roads are wet." Now look at the compound proposition, $p \rightarrow q$. The only row of our truth table that contradicts $p \rightarrow q$ is the second row, which says it's raining but the roads are dry. Notice that the third row is not false, because for instance there might be an open fire hydrant wetting the road.

- If two compound propositions have the same truth table, then we say they are logically equivalent.

  Example: $p \to q$ is logically equivalent to $\neg p \vee q$.

- Def: the converse of $p \to q$ is $q \to p$, the inverse is $\neg p \to \neg q$, and the contrapositive is $\neg q \to \neg p$.

$p \to q$ is logically equivalent to its contrapositive.
Also, the converse is logically equivalent to the inverse.

- A shorthand for "p if and only if q" is the biconditional operator "$\leftrightarrow$". It has truth table:

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

And it is equivalent to $(p \to q) \wedge (q \to p)$.

Sometimes compound propositions get complicated, so we have an established order of precedence of common operators:

operator	$\neg$	$\wedge$	$\vee$	$\to$	$\leftrightarrow$
precedence	1	2	3	4	5

Section 1.2: Propositional Equivalences

People care a lot about logical equivalences. We will learn a few tricks to help determine if two compound propositions are actually equivalent.

- Def: A compound proposition that is always true is called a tautology, and one that is always false is called a contradiction.

  Example: $p \vee \neg p$ is a tautology, and $p \wedge \neg p$ is a contradiction.

Here are some nice tricks to use in simplifying compound propositions:
(the "$\equiv$" symbol means "logically equivalent.")

De Morgan's Laws	Distributive Laws
$\neg(p \wedge q) \equiv \neg p \vee \neg q$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
$\neg(p \vee q) \equiv \neg p \wedge \neg q$	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

There are several other logical equivalences listed on page 24 and 25 in the book, but these two laws are the most important ones.

The point of learning laws like this is so we can compare compound propositions without writing out pesky truth tables.

An interesting question is this: Given a truth table, can you find a compound proposition to match it? It turns out the answer is yes, and that you can always build any truth table solely from $\neg$, $\wedge$, and $\vee$. If you want to see how this works, check out exercise 42 in 1.2.

Since the operators $\neg$, $\wedge$, and $\vee$ can be used to build any truth table, we say they are functionally complete. Notice that since $p \wedge q \equiv \neg(\neg p \vee \neg q)$, we can conclude that $\neg$ and $\vee$ alone form a functionally complete set. Similarly, $\neg$ and $\wedge$ do too. In your homework you will discover a single operator that is itself functionally complete. Wow!