Truth Tables

Now, given a particular compound statement, we can use a truth table to determine which values of the boolean variables result in the statement being true.

The idea here is to simply make a table, listing all the possible combinations of values for each of the boolean variables in a statement. Then, plug these values into the statement and see if it is true or not with these values. This is probably easiest seen with an example.

Consider the statement: p □ (q □ □r). Here is a truth table:

p	q	r	q □ □r	p□(q□□r)
0	0	0	1	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

Thus there are three possible combinations of values for p, q, and r that make p □ (q □ □r) true.
To see an example, let p, q and r be the following statements:

p: I have taken out the trash.
q: I have finished cleaning the dishes.
r: I have watched 5 hours of TV.

Assume that a child is a good one if p □ (q □ □r) holds true, and bad otherwise. Under what conditions is a child good?

If it turns out that a statement is always true (such as p □ □p), then it is called a *tautology*.

If a statement is always false (such as make p □ □p), then it is called a contradiction.

EXERCISE: Make a truth table for the statement: (p □□q)□r.

Algebra of Propositions

It would be nice if we had some methodology for determining if two logical expressions are equal, or a method of simplifying a given logical expression.

Perhaps the most obvious way to check for the equality of two logical expressions it to write out truth tables for both. However, this could be quite tedious. (But it does always work...)

Two statements s and t are considered to be logically equivalent if s □ t.

We will need some laws to simplify logical expressions. Here is the list of laws from the text:

1) □□p □ p (Law of Double Negation)
2) □(p □ q) □ □p □ □q (De Morgan's Laws)
3) □(p □ q) □ □p □ □q
4) p □ q □ q □ p (Commutative Laws)
5) (p □ q) □ r □ p □ (q □ r) (Associative Laws)
6) p □ (q □ r) □ (p □ q) □ (p □ r) (Distributive Laws)
7) p □ (q □ r) □ (p □ q) □ (p □ r)
6) p □ p □ p (Idempontent Laws)
7) p □ F □ p, p □ T □ p (Identity Laws)
8) p□□p □ T, p□□p □ F (Inverse Laws)
9) p □ T □ T, p □ F □ F (Domination Laws)
10) p □ (p □ q) □ p (Absorption Laws)
11) p □ (p □ q) □ p