From Truth Evaluations to Validity

In the the exercises for 6.3, we learned how to use "truth evaluations" to determine the truth value of a compound statement where we are given the truth values of its component parts. So, if we know that "P" and "Q" are both true, we can use our knowledge of how the connectives function to calculate that "(~P • Q)" must be false. I am assuming at this point you understand this, and understand how to do these "calculations" or "evaluations."

We also know at this point that, given the way a truth table is constructed, it will contain all the possible assignments of truth values to a group of simple statements (i.e., statements we represent with capital letters). This is why the truth table for negation has only 2 lines--one for when the simple statement is false, and another for when it is true. When we have 2 simple statements (as we do in conjunctions, disjunctions, conditionals, and biconditionals), our truth tables must have 4 lines, because there are four possible combinations of truth values for two simple statements (i.e., false, false; false, true; true, false; and true, true).

But now we can put these two notions together: that is, we can use truth tables not simply to define how our five connectives work, but to evaluate the truth value of a statement for *all possible* combinations of truth value assignments to its simple components. Each line of a truth table, that is, is simply an "evaluation" of the truth value of complex statement given an assignment of truth values to its component parts. And a truth table is simply a list of all the possible truth value assignments to these component parts.

In the exercises for 6.5, you are asked to construct "Complex Truth Tables." These exercises simply require you to do the very same steps you did in the exercises for 6.3 (Truth Evaluations), but this time doing it for each line of a truth table. So, if you have mastered the exercises for 6.3, what you are being asked in 6.5 is simply to do the same thing four times, one time for each line of a truth table. (In some of the "harder" exercises, you may need to this eight times.)

Using such "Complex" truth tables we can show, for example, that "(P • ~P)" comes out false on all possible assignments of truth values to its parts. (The truth value of "(P • ~P)" is "0" on all rows of the table.) Likewise, we can use a complex truth table to show that the truth value of “(P v P)” comes out true for all possible truth value assignments. A statement that is false on all possible truth value assignments is *logically false*, or *contradictory*. One that is true for all possible truth value assignments is *logically true*, and is called a “tautology.”

But we can also use truth tables to show that an argument is valid or invalid. To do this, we must construct a truth table that contains, on the left side, all of the simple components (i.e., all of the capital letters) contained in all of the statements in that argument--i.e, all the letters contained in the both the premises and the conclusion. This truth table will therefore list all the possible combinations of truth value assignments to all of the simple statements out of which all the premises and the conclusion are composed. On the right part of the table, we will list all of the premises and the conclusion. Each line of the table, therefore, will contain a “truth evaluation” for each of these statements on a given assignment of truth values. And since the table contains all the possible assignments of truth values of the simple components, the truth table will then show us all the possible evaluations of all the statements that make up the argument.

To say that an argument is *valid*, remember, means to say that it is not possible for the premises all to be true and the conclusion false. Truth tables for arguments simply list all the possibilities for the statements in an argument. So, if it is possible for all the premises to be true and the conclusion false, then this will appear in the table: i.e., there will be at least one line of the truth table where the truth value of all the premises is “1” and the truth value of the conclusion is “0.” Alternatively, if there is no such line (with the premises all true and the conclusion false), then it is not possible. (Again, since a truth table lists all the possibilities, if its not on the table, then its not a possibility.)

So, we can use truth tables for arguments to establish validity and invalidity. To do so, we construct a truth table for the argument in the manner described above. **In a valid argument, there will be no line where the truth value for all of the premise =1 and the truth value of the conclusion =0.** Alternately put, **If there is even one line of the truth table where all the premises =1 and the conclusion =0, then the argument is *in*valid.**

Examples:

Consider the following simple argument:

P

~Q

∴ (~P ⊃ Q)

(Note: "∴" is the symbol for "therefore." We use it to indicate the conclusion of an argument.)

So, we want to construct a complex truth table that contains all the premises and the conclusion. The left side of the table will contain all the simple statements--i.e., capital letters--we find in any of the statements of this argument, and the right side will contain a list of all the premises, and the conclusion. (In this case one of our premises, "P," is one of those simple statements that occur on the left side. But we will repeat it on the right side for the purposes of clarity.)

Here is what this truth table would look like, when filled in:

P	Q	P	~	Q	(~P	⊃	Q)
0	0	0	1	0	1	0	0
0	1	0	0	1	1	1	1
1	0	1	1	0	0	1	0
1	1	1	0	1	0	1	1
		⇑	⇑			⇑	
		P1	P2			C	

The arrows below the chart indicate the columns that contain the truth evaluations for the premises and conclusion. Above the first arrow, we a column for "P," which is our first premise. Above the second arrow, we have a column with the truth values for "~Q," our second premise. And above the final arrow, we have a column with the truth values for "(~P ⊃ Q)," which is the conclusion.

So, is there any line where all the premises are true and the conclusion false? The only line where both of the premises are true is the third line. But on this line the conclusion is true. So, there is no line where the premises are all true and the conclusion is false. Therefore, this argument is *valid*, and we have proven this using this truth table.

The only thing "tricky" here is that we prove validity by "looking for" invalidity, and not finding it. That is, if we "find" what we are "looking for"--a line with all true premises and a false conclusion, then we know that the argument is *invalid*. It is only when there is no such line that we prove that the argument is valid.

This is a simple point to understand, but *very easy to get tripped up by in practice*! So remember, **an argument is valid when you *don't* find a line with a "1" for every premise and a "0" for the conclusion.** If you find such a line, you have proven that the argument is not valid.

<u>Constructing Truth Tables with more than 2 components</u>

Thus far we have looked at truth tables for compound statements with a single simple component (i.e., a single capital letter), and for those with two simple components. In the first case, we constructed a table with two rows, and in the second case, we constructed one with four rows. But we can, in principle, use truth tables to evaluate statements and arguments with any number of simple statements. (But they can get *very* large!)

You may have noticed a pattern: with 1 simple statement, we need 2 rows. With 2 simple statements we need twice as many rows, i.e., 4. In general, every time we add another simple statement to the left hand side of the table, we need twice as many total rows as before. So, if we have 3 simple statements, we will need 8 lines to list all the possible truth value assignments to these simple statements. If we had 4 simple statements, we would need 16. For 5 simple statements, we would need 32. (If you like this kind of equation: for n simple statements, the number of rows we need to list all the possible combinations of truth value assignment is 2^n.)

The next question is the *order* to list all these rows. Strictly speaking, the order is arbitrary, as long as you list all the possibilities. But we will adopt a single method, so that we list things in the same order. Here is how I do it: You will have a number of columns, one for each simple statement. Start with the rightmost column, alternate “0,” and “1,” in that order, all the way down--for as many rows as you need. (2 rows for 1 simple statement, 4 rows for 2 simple statements, 8 rows for 3 simple statements, and so on.) Now move to the next column to the left. List “0”s and “1”s in the same order, but double the number in each group, i.e., fill in “0,” “0,” “1,” “1,” all the way down. Each time you move to the next column to the left, double the number of “0”s and “1”s in each group.

So, a truth table with three simple components would look like this:

P	Q`	R
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

⇑

Alternating 0, 1,
all the way down

⇑

Alternating 0, 0, 1, 1,
all the way down

⇑

Alternating 0, 0, 0, 0, 1, 1, 1, 1 all the way down.

Aa truth table with 4 components would like:

P	Q	R	S
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

(Again, for those of you who like such things: There is an order here. The first line starts with all "0"s and the last line ends with all "1"s. The order? The lines are in *numerical order* if you are counting in *base two*!)

On the test, you won't be asked to deal with truth tables with more than 3 simple components--i.e., with 8 lines. But it is important to know how they could be constructed for any number of simple statements.