Chapter 1. The Foundations: Logic, Sets, and Functions

Section 1.1. Logic

Note. In this section we introduce elementary symbolic logic. We define propositions and put truth values on propositions.

Definition. A *proposition* is a statement that is either true or false, but not both. The *truth value* of a proposition is true, denoted T, if it is a true proposition and false, denoted F, if is is a false proposition.

Definition 1.1.1. Let p be a proposition. The statement "It is not the case that p" is another proposition called the *negation* of p, denoted $\neg p$ (read "not p").

Note. Negation is an example of a logical operator. New propositions, called *compound propositions* can be formed from existing propositions using logical operators. A *truth table* gives the truth values of various propositions. For example, consider

p	$\neg p$
T	F
F	T

Definition 1.1.2. Let p and q be propositions. The proposition "p and q," denoted $p \wedge q$, is the proposition that is T when p and q are true and is false otherwise. This proposition is called the *conjunction* of p and q.

Definition 1.1.3. Let p and q be propositions. The proposition "p or q," denoted $p \vee q$, is the proposition that is false when p and q are false and is true otherwise. This proposition is called the *disjunction* of p and q.

Note. We have the truth table for $p \vee q$ and $p \wedge q$ as:

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

Definition 1.1.4. Let p and q be propositions. The *exclusive or* of p and q, denoted $p \oplus q$, is the proposition that is T when exactly one of p and q is true and is false otherwise.

Note. We have the truth table for $p \oplus q$ as:

p	q	$p \oplus q$
T	T	F
T	F	F
F	T	T
F	F	F

Definition 1.1.5. Let p and q be propositions. The implication $p \to q$ (read "p implies q") is the proposition that is false when p is T and q is F and T otherwise. p is called the *hypothesis* (or *antecedent* ore *premise*) and q is called the *conclusion* (or *consequence*).

Note. We have the truth table for $p \to q$ as:

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

Note. On page 6 the text gives some alternative ways to read $p \to q$ (such as "q whenever Pp"). The text also gives some warnings with the proposition "If you make more than $25,000, then you must file a tax return."

Example. Page 11 number 6.

Definition 1.1.6. Let p and q be propositions. The *biconditional* $p \leftrightarrow q$ is the proposition that is true when p and q have the same truth values and is false otherwise.

Note. We have the truth table for $p \leftrightarrow q$ as:

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

Definition. The *converse* of proposition $p \to q$ is the proposition $q \to p$. The *contrapositive* of proposition $p \to q$ is the proposition $\neg q \to \neg p$.

Note. We have the truth table for the converse and contrapositive of $p \to q$ as:

p	q	$p \to q$	$q \to p$	$\neg p$	$\neg q$	$\neq q \to \neg p$
T	T	T	T	F	F	T
T	F	F	T	F	T	F
F	T	T	F	T	F	T
F	F	T	T	T	T	T

Notice that the truth value of $p \to q$ is the same as the truth value of $\neg q \to \neg p$. So the truth value of a proposition and its contrapositive are the same.

Example. Page 12 number 12. This is a biconditional proposition. The barber shaves those who do not shave themselves AND the barber shaves only those who shave themselves. Let p be "the barber shaves this person" and q be "this person does not shave himself." Then the story can be represented as $p \leftrightarrow q$. Now suppose "this person" is the barber. Then we consider two cases:

(1) If p is T and the barber shaves "this person" (i.e., himself). Then q is F and "this person (barber) does shave himself.

(2) If p is F and the barber does not shave "this person" (himself), then q is T and "this person (barber) does not shave himself."

So we have the truth table:

Case	p	q	$p \leftrightarrow q$
(1)	T	F	F
(2)	F	T	F

In either case $p \leftrightarrow$ if false and so there can be no such barber.

Definition. A *bit* (for **b**inary dig**it**) has two possible values: 0 (for OFF or FALSE) and 1 (for ON or TRUE). We can therefore treat bits as statements and use logical connections with them (which are called *bit operations*).

Definition 1.1.7. A *bit string* is a sequence of bits. The length of this string is the number of bits in the string.

Note. When dealing with bit strings, we may use OR, AND, XOR for $\vee$, $\wedge$, $\oplus$, respectively.

Example. Page 13 number 30.

Example. Page 13 number 24.

Revised: 4/1/2019