# COMS10014 Solutions 2: Logic 1

### 1. Truth Tables

1. The truth table:

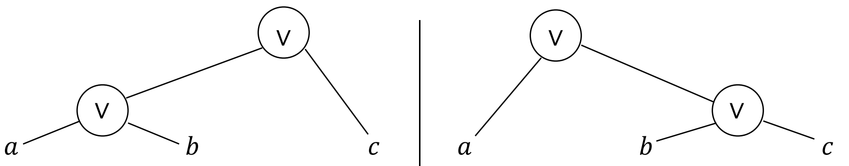
| $a$ | $b$ | $a \to b$ | $b \to a$ |
|---|---|---|---|
| F | F | T | T |
| F | T | T | F |
| T | F | F | T |
| T | T | T | T |

shows that the $\to$ operation is not commutative, as $a \to b \not\equiv b \to a$.

2. With this table we can see that $a \vee b \equiv b \vee a$ so the $\vee$ operation is commutative:

| $a$ | $b$ | $a \vee b$ | $b \vee a$ |
|---|---|---|---|
| F | F | F | F |
| F | T | T | T |
| T | F | T | T |
| T | T | T | T |

For associativity, we make a truth table with three variables $a, b, c$. Then we fill in the columns on the right as follows: columns 1, 3, 5, 6, 8 and 10 below are for variable terms, so we just copy over the relevant column from the variable. Columns 2 and 9 are for operation terms where we have just completed the columns for their child terms, so we can fill them in next. Finally, columns 4 and 7 are operation terms whose child terms are now filled in, so we can complete them now. It turns out that columns 4 and 7 are identical, so the two terms are equivalent.




| $a$ | $b$ | $c$ | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| F | F | F | F | F | F | F | F | F | F | F | F | F |
| F | F | T | F | F | F | T | T | F | T | F | T | T |
| F | T | F | F | T | T | T | F | F | T | T | T | F |
| F | T | T | F | T | T | T | T | F | T | T | T | T |
| T | F | F | T | T | F | T | F | T | T | F | F | F |
| T | F | T | T | T | F | T | T | T | T | F | T | T |
| T | T | F | T | T | T | T | F | T | T | T | T | F |
| T | T | T | T | T | T | T | T | T | T | T | T | T |

3. DeMorgan's laws say you can pull a $\neg$ into or out of brackets around a $\wedge$ or $\vee$ by flipping the binary operator. With a truth table:

| | | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. |
|---|---|---|---|---|---|---|---|---|---|
| $a$ | $b$ | $a \vee b$ | $a \wedge b$ | $\neg(a \vee b)$ | $\neg(a \wedge b)$ | $\neg a$ | $\neg b$ | $\neg a \wedge \neg b$ | $\neg a \vee \neg b$ |
| F | F | F | F | T | T | T | T | T | T |
| F | T | T | F | F | T | T | F | F | T |
| T | F | T | F | F | T | F | T | F | T |
| T | T | T | T | F | F | F | F | F | F |

we can see that $\neg(a \vee b) \equiv \neg a \wedge \neg b$ and $\neg(a \wedge b) \equiv \neg a \vee \neg b$. Here, we computed columns 1 and 2 as the $\vee$ and $\wedge$ respectively of the $a$ and $b$ columns, column 3 as the negation of column 1, column 4 as the negation of column 2, columns 5 and 6 as the negations of the $a$ and $b$ columns, column 7 as the $\wedge$ of columns 5 and 6, and column 8 as the $\vee$ of columns 5 and 6.

4. With a table:

| $a$ | $b$ | $\neg a$ | $\neg a \vee b$ | $a \to b$ |
|---|---|---|---|---|
| F | F | T | T | T |
| F | T | T | T | T |
| T | F | F | F | F |
| T | T | F | T | T |

Here we computed the $\neg a$ column as the negation of the $a$ column, the $\neg a \vee b$ column as the $\vee$ of the $\neg a$ and the $b$ columns, and the $a \to b$ column using the definition of $\to$. We can see that $\neg a \vee b \equiv a \to b$.

5. Here we compute $a \oplus b$, $a \vee b$ and $a \wedge b$ from the definition, then $\neg(a \wedge b)$ by negating the column for $a \wedge b$, then the last column by taking the $\wedge$ of the two columns for the relevant sub-expressions. This shows that $a \oplus b \equiv (a \vee b) \wedge \neg(a \wedge b)$ as the two columns for these expressions are identical.

| $a$ | $b$ | $a \oplus b$ | $a \vee b$ | $a \wedge b$ | $\neg(a \wedge b)$ | $(a \vee b) \wedge \neg(a \wedge b)$ |
|---|---|---|---|---|---|---|
| F | F | F | F | F | T | F |
| F | T | T | T | F | T | T |
| T | F | T | T | F | T | T |
| T | T | F | T | T | F | F |

6. Here we compute the last column, as usual, as the $\vee$ of the two columns for the relevant sub-expressions. This shows, because the columns are identical, that $a \oplus b \equiv (a \wedge \neg b) \vee (\neg a \wedge b)$.

| $a$ | $b$ | $\neg a$ | $\neg b$ | $a \oplus b$ | $(a \wedge \neg b)$ | $(\neg a \wedge b)$ | $(a \wedge \neg b) \vee (\neg a \wedge b)$ |
|---|---|---|---|---|---|---|---|
| F | F | T | T | F | F | F | F |
| F | T | T | F | T | F | T | T |
| T | F | F | T | T | T | F | T |
| T | T | F | F | F | F | F | F |

**2. Propositions**

1. $\neg q$
2. $p \wedge \neg q$
3. $r \rightarrow q$
4. $(r \vee q) \rightarrow \neg p$
5. $\neg p \vee (\neg p \rightarrow q)$
6. $\neg p \vee \big((\neg p \wedge r) \rightarrow q\big)$
7. $p \leftrightarrow \neg q$
8. $(p \wedge r) \rightarrow q$
9. If it is raining, then it is freezing.
10. It is snowing if and only if it is freezing.
11. If it is snowing, then it is both raining and freezing.
12. If and only if it is raining or freezing, then it is snowing.

For 5. and 6., it is not clear in English (different books do this different ways) whether "either … or" means an inclusive or exclusive or. An "or" without "either" is always inclusive in logic, however. If you used $\oplus$ instead of $\vee$, you can count that as correct too.

**3. Brackets**

1. You cannot eliminate anything here, as $\rightarrow$ is not associative and logical operators bind to the *right*. Removing the brackets would change the semantics of the term.
2. You cannot eliminate anything here either, as the top-level operator $\wedge$ has lower priority than the $\vee$ children.
3. $a \wedge b \vee c \wedge d$. Here, the higher priority of the $\vee$ means the unbracketed term must be parsed the same way.
4. You cannot eliminate anything here, as $\neg a \vee b$ would parse as $(\neg a) \vee b$ which is not semantically equivalent, as you can check with a truth table. (You could apply DeMorgan, but that is not the same thing as eliminating brackets, and not what was asked for here.)
5. $a \wedge b \wedge c$. Although we said that logical operators bind to the right, $\wedge$ is associative, so this gives a different term that is however semantically equivalent to the original. The exercise explicitly allowed doing this.
6. $p \wedge q \rightarrow p \rightarrow q$. The higher priority of $\rightarrow$ makes sure that the $\wedge$ ends up in the right place, and the $\rightarrow$ binds to the right so the rest of the term is parsed the same way.
7. $p \vee \neg q \vee s \leftrightarrow \neg(s \wedge \neg r)$.
8. $(\neg p \vee s) \wedge q \vee \neg\neg\neg s$.

**4. Theory Questions**

1. There are four: (T, T), (T, F), (F, T), (F, F).

2. A binary operation assigns a truth value to each of the four assignments above. This gives $2^4 = 16$ possible functions. Put another way: how many ways are there are to fill in a column in a truth table with 2 variables? Since there are 4 rows in this table (excluding headers), and there are two ways to fill in the box in each row, this gives $2^4$ ways.

3. We will display the table with rows/columns transposed to better fit on a page. Note that, if we replace T with 1 and F with 0, then the body of the table is counting from 0 to 31 in binary.

| $p$ | T | T | F | F | name |
|---|---|---|---|---|---|
| $q$ | T | F | T | F | |
| | F | F | F | F | $F$ |
| | F | F | F | T | $\downarrow$ |
| | F | F | T | F | |
| | F | F | T | T | $\neg L$ |
| | F | T | F | F | |
| | F | T | F | T | $\neg R$ |
| | F | T | T | F | $\oplus$ |
| | F | T | T | T | $\uparrow$ |
| | T | F | F | F | $\wedge$ |
| | T | F | F | T | $\leftrightarrow$ |
| | T | F | T | F | $R$ |
| | T | F | T | T | $\rightarrow$ |
| | T | T | F | F | $L$ |
| | T | T | F | T | $\leftarrow$ |
| | T | T | T | F | $\vee$ |
| | T | T | T | T | $T$ |

The functions $T$ and $F$ ignore their inputs and just return the same truth value all the time; the functions $L$ and $R$ ignore one of their inputs, and always return the left (resp. right) input. $\neg L$ and $\neg R$ are functions that ignore one of their inputs, and negate the other. $\leftarrow$ is the $\rightarrow$ with the order of its inputs swapped (since this is not commutative, it makes a difference). You have not seen $\uparrow$ and $\downarrow$ yet in this unit, but you will soon so we give their symbols here. The two rows that do not have a name are negated implications, that is $\neg(a \rightarrow b)$ and $\neg(b \rightarrow a)$, but these do not normally get a symbol of their own.

4. This is like 2., but with ‘2 inputs replaced by ‘$n$ inputs’. For $n$ inputs, there are $2^n$ different possible assignments; a function maps each one to a truth value so there are $2^{(2^n)}$ logical operations with $n$ inputs. Note the brackets, as the exponential function is not associative.