Sample Answers to Tutorial Exercises, Week 2

6. The function is well-typed: `f :: Num a => [a] -> a` and all the equations make sense. The function takes a list of numbers and returns a number, where "number" means element of some specific numeric type (Int, Integer, Float, Double). The first equation deals with an input list that is empty, the second deals with any singleton list. The third deals with the remaining case, that is, a list with more than one element. In the second equation, `x` gets bound to a number, of the third, `y` gets bound to a list (instead of `y` it would be customary to use the name `xs`, to suggest that we have a list (a plurality) of elements.

7. (a) Not equivalent:

P	Q	$\neg P$	$\Rightarrow$	Q	P	$\Rightarrow$	$\neg Q$
0	0	1	0	0	0	1	1
0	1	1	1	1	0	1	0
1	0	0	1	0	1	1	1
1	1	0	1	1	1	0	0
			X			X	

We see that the columns for the two implications are different.

(b) Not equivalent:

P	Q	$\neg P$	$\Rightarrow$	Q	Q	$\Rightarrow$	$\neg P$
0	0	1	0	0	0	1	1
0	1	1	1	1	1	1	1
1	0	0	1	0	0	1	0
1	1	0	1	1	1	0	0
			X			X	

(c) Equivalent:

P	Q	$\neg P$	$\Rightarrow$	Q	$\neg Q$	$\Rightarrow$	P
0	0	1	0	0	1	0	0
0	1	1	1	1	0	1	0
1	0	0	1	0	1	1	1
1	1	0	1	1	0	1	1
			👍			👍	

(d) Equivalent:

P	Q	$(P$	$\Rightarrow$	$Q)$	$\Rightarrow$	P
0	0	0	1	0	0	0
0	1	0	1	1	0	0
1	0	1	0	0	1	1
1	1	1	1	1	1	1
		👍				👍

(e) and (f): the pair in (e) equivalent; but the pair in (f) are not equivalent:

P	Q	R	P	$\Rightarrow$	$(Q$	$\Rightarrow$	$R)$	Q	$\Rightarrow$	$(P$	$\Rightarrow$	$R)$	$(P$	$\Rightarrow$	$Q)$	$\Rightarrow$	R
0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	0
0	0	1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1
0	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0
0	1	1	0	1	1	1	1	1	1	0	1	1	0	1	1	1	1
1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	0	1	0
1	0	1	1	1	0	1	1	0	1	1	1	1	1	0	0	1	1
1	1	0	1	0	1	0	0	1	0	1	0	0	1	1	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

☞ X ☜ X

Note that the formulas in (e) are both equivalent to $(P \wedge Q) \Rightarrow R$, which explains why the order of P and Q does not matter here.

(g) Equivalent:

P	Q	R	$(P$	$\wedge$	$Q)$	$\Rightarrow$	R	P	$\Rightarrow$	$(Q$	$\Rightarrow$	$R)$
0	0	0	0	0	0	1	0	0	1	0	1	0
0	0	1	0	0	0	1	1	0	1	0	1	1
0	1	0	0	0	1	1	0	0	1	1	0	0
0	1	1	0	0	1	1	1	0	1	1	1	1
1	0	0	1	0	0	1	0	1	1	0	1	0
1	0	1	1	0	0	1	1	1	1	0	1	1
1	1	0	1	1	1	0	0	1	0	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1

☞   ☜

(h) Equivalent:

P	Q	R	$(P$	$\vee$	$Q)$	$\Rightarrow$	R	$(P$	$\Rightarrow$	$R)$	$\wedge$	$(Q$	$\Rightarrow$	$R)$
0	0	0	0	0	0	1	0	0	1	0	1	0	1	0
0	0	1	0	0	0	1	1	0	1	1	1	0	1	1
0	1	0	0	1	1	0	0	0	1	0	0	1	0	0
0	1	1	0	1	1	1	1	0	1	1	1	1	1	1
1	0	0	1	1	0	0	0	1	0	0	0	0	1	0
1	0	1	1	1	0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	0	0	1	0	0	0	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

☞   ☜

8. The formula is equivalent to P.

9. We see that the columns for P and for $(P \oplus Q) \oplus Q$ are identical:

P	Q	$(P$	$\oplus$	$Q)$	$\oplus$	Q
0	0	0	0	0	0	0
0	1	0	1	1	0	1
1	0	1	1	0	1	0
1	1	1	0	1	1	1

☞   ☜

10. Even with three variables the truth table is manageable, so let us construct it.

			(1A)					(1B)					(2)		
P	Q	R	P	$\Leftrightarrow$	$(Q$	$\Leftrightarrow$	$R)$	$(P$	$\Leftrightarrow$	$Q)$	$\Leftrightarrow$	R	$P \wedge Q \wedge R$	$\vee$	$\neg P \wedge \neg Q \wedge \neg R$
0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	1
0	0	1	0	1	0	0	1	0	1	0	1	1	0	0	0
0	1	0	0	1	1	0	0	0	0	1	1	0	0	0	0
0	1	1	0	0	1	1	1	0	0	1	0	1	0	0	0
1	0	0	1	1	0	1	0	1	0	0	1	0	0	0	0
1	0	1	1	0	0	0	1	1	0	0	0	1	0	0	0
1	1	0	1	0	1	0	0	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

B is a logical consequence of A (we write $A \models B$) iff B is true for any assignment of variables which makes A true. So we see that (1) $\not\models$ (2), because cases like 1 0 0 that make (1) true but (2) false. Similarly, (2) $\not\models$ (1), because the case 0 0 0 makes (2) true but (1) false.

11. Card 3 cannot be red, because red cannot lie, and Card 3 says it is the joker. So suppose Card 2 is red. Then Card 1 is black and Card 3 is the joker. But then Card 1 actually speaks a truth, which cannot happen. So we conclude that Card 1 is the red card. It follows that Card 3 is the joker and Card 2 is black.

12. There is no contradiction at all. The formula is true if (and only if) P is false. False implies anything. For the same reason, $\neg P \Rightarrow P$ is satisfiable; it is not a contradiction. But $P \Leftrightarrow \neg P$ is clearly a contradiction. (If you disagree with any of these statements, draw truth tables.)

13. The connective $\Leftrightarrow$ is part of the language that we study, namely the language of propositional logic. So $A \Leftrightarrow B$ is just a propositional formula.

    The symbol $\equiv$ belongs to a *meta-language*. The meta-language is a language which we use when we reason *about* some language. In this case we use $\equiv$ to express whether a certain relation holds between formulas in propositional logic.

    More specifically, $F \equiv G$ means that we have both $F \models G$ and $G \models F$. In other words, F and G have the same value for every possible assignment of truth values to their variables. The two formulas are logically equivalent.

    On the other hand $F \Leftrightarrow G$ is just a propositional formula (assuming F and G are propositional formulas). For some values of the variables involved, $F \Leftrightarrow G$ may be false, for other values it may be true. By the definition of validity, $F \Leftrightarrow G$ is *valid* iff it is true for *every* assignment of propositional variables in F and G.

    We want to show that $F \equiv G$ iff $F \Leftrightarrow G$ is valid.

    (a) Suppose $F \equiv G$. Then F and G have the same values for each truth assignment to their variables[1]. But that means that, when we construct the truth table for $F \Leftrightarrow G$, it will have a t in every row, that is, $F \Leftrightarrow G$ is valid.

    (b) Suppose $F \Leftrightarrow G$ is valid. That means we find a t in each row of the truth table for $F \Leftrightarrow G$. But we get a t for $F \Leftrightarrow G$ iff the values for F and G agree, that is, either both are f, or both are t. In other words, F and G agree for every truth assignment. Hence $F \equiv G$.

    You may think that this relation between validity and biimplication is obvious and should always be expected, and indeed we will see that it carries over to first-order predicate logic. But there are (still useful) logics in which it does not hold.

---

[1] We should perhaps be more careful here, because F and G can be logically equivalent without F having the exact same set of variables as G—can you see how? So we should say that we consider both of F and G to be functions of the *union* of their sets of variables.

14. If you negate a satisfiable proposition, you can never get a tautology, since at least one truth table row will yield false.

   You will get another satisfiable proposition iff the original proposition is not valid. For example, P is satisfiable (but not valid), and indeed $\neg P$ is satisfiable.

   Finally, if we have a satisfiable formula which is also valid, its negation will be a contradiction. Example: $P \vee \neg P$.

15. Clearly the columns for $P \Leftrightarrow Q$ and for $\neg P \Leftrightarrow \neg Q$ are identical.

P	Q	P	$\Leftrightarrow$	Q	$\neg$	P	$\Leftrightarrow$	$\neg$	Q
0	0	0	1	0	1	0	1	1	0
0	1	0	0	1	1	0	0	0	1
1	0	1	0	0	0	1	0	1	0
1	1	1	1	1	0	1	1	0	1

16. Yes, correct. You will want to check that with the truth tables.

17. The formula is equivalent to $P \Rightarrow Q$. This is easily checked with a truth table, but how can we simplify $(P \wedge Q) \Leftrightarrow P$ when we don't know what it is supposed to be equivalent to? Well, we can just try. Let us expand the biimplication and obtain $(P \Rightarrow (P \wedge Q)) \wedge ((P \wedge Q) \Rightarrow P)$. Intuitively, the conjunct on the right is just true, and we can check that with a truth table. So we have found that the original formula is equivalent to $P \Rightarrow (P \wedge Q)$, which isn't any shorter, but still. We can rewrite the result as $(P \Rightarrow P) \wedge (P \Rightarrow Q)$, and now it becomes clear that all we need is $P \Rightarrow Q$.

18. Let us draw the truth tables.

   (a)

P	Q	P	$\Leftrightarrow$	$((P$	$\Rightarrow$	$Q)$	$\Rightarrow$	$P)$
0	0	0	1	0	1	0	0	0
0	1	0	1	0	1	1	0	0
1	0	1	1	1	0	0	1	1
1	1	1	1	1	1	1	1	1

$\Uparrow$

   Hence satisfiable, and in fact valid (all 1).

   (b)

P	Q	$(P$	$\Rightarrow$	$\neg$	$Q)$	$\wedge$	$((P$	$\vee$	$Q)$	$\Rightarrow$	$P)$
0	0	0	1	1	0	1	0	0	0	1	0
0	1	0	1	0	1	0	0	1	1	0	0
1	0	1	1	1	0	1	1	1	0	1	1
1	1	1	0	0	1	0	1	1	1	1	1

$\Uparrow$

   Hence satisfiable (at least one 1), but not valid (not all 1). The truth table shows the formula is equivalent to $\neg Q$.

   (c)

P	Q	$((P$	$\Rightarrow$	$Q)$	$\Rightarrow$	$Q)$	$\wedge$	$(Q$	$\oplus$	$(P$	$\Rightarrow$	$Q))$
0	0	0	1	0	0	0	0	0	1	0	1	0
0	1	0	1	1	1	1	0	1	0	0	1	1
1	0	1	0	0	1	1	0	0	0	1	0	0
1	1	1	1	1	1	1	0	1	0	1	1	1

$\Uparrow$

   Hence not satisfiable (and so certainly not valid).

4