Fundamentals of Computing

Tutorial 2: Boolean formulas and functions; arguments

Model answers

1. Which of these sentences are propositions? (What are the truth values of those that are propositions?)

   (a) Liverpool is the capital of the UK.

   (b) $x + 2 = 11$.

   (c) $2 + 3 = 5$.

   (d) Answer this question!

   Answer. (a) is either true or false (the latter is the case in our world), so it is a proposition; however, we cannot say that (b) is either true or false because x is a variable; (c) is a (true) proposition; while (d) is an imperative sentence, so it is not a proposition.

2. Consider the statements S_1–S_3 below:

   S_1: Charlie is not a cook.

   S_2: Alice is an architect or Bob is a builder.

   S_3: If Bob is a builder, then Charlie is a cook.

   Which of the following arguments are logically correct?

   (*i*) Suppose S_1–S_3 are true. Then Alice is an architect.

   (*ii*) Suppose S_1–S_3 are true. Then Bob is a builder.

   (*iii*) Suppose S_1–S_3 are true. Then Charlie is a builder.

   (*iv*) Suppose S_1–S_3 are true. Then Charlie is not a builder.

   Answer. First, we represent the given statements as Boolean formulas. We introduce the following *propositional variables* (with unknown truth-values):

   – CC, standing for the proposition 'Charlie is a cook'
   – AA, standing for the proposition 'Alice is an architect'
   – BB, standing for the proposition 'Bob is a builder'
   – CB, standing for the proposition 'Charlie is a builder'

   Then the given statements can be represented as the following Boolean formulas:

   $$S_1 = \neg CC, \qquad S_2 = AA \vee BB, \qquad S_3 = BB \to CC$$

   Next, we construct the truth table for the variables that occur in S_1–S_3 and compute the corresponding truth-values of S_1–S_3 using the table for $\neg$, $\vee$ and $\to$ on page 25 in FoC-I:

CC	AA	BB	S_1	S_2	S_3
0	0	0	1	0	1
0	0	1	1	1	0
0	1	0	1	1	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	0	1	1
1	1	0	0	1	1
1	1	1	0	1	1

Observe that there is only one 'situation' (= row in the truth-table) where all of S_1–S_3 are true: $CC = 0$, $AA = 1$, $BB = 0$. We are now in a position to answer questions (i)–(iv):

(i) By the definition on page 30 in FoC-I, AA is a logical consequence of S_1–S_3 if, and only if, in every situation where the premisses S_1–S_3 are true, AA is also true. This is indeed the case. So yes, AA is a logical consequence of S_1–S_3, that is, Alice is an architect.

(ii) Similarly, BB is a logical consequence of S_1–S_3 if, and only if, in every situation where the premisses S_1–S_3 are true, BB is also true. This is not the case. So BB is *not* a logical consequence of S_1–S_3. Moreover, if S_1–S_3 are true, then Bob is *not* a builder.

(iii) S_1–S_3 do not say anything about Charlie being a builder (and CB does not occur in the truth-table). We can set $CC = 0$, $AA = 1$, $BB = 0$, $CB = 0$, which makes S_1–S_3 true and CB false. Therefore, CB is not a logical consequence of S_1–S_3, and so we cannot conclude that Charlie is a builder.

(iv) We can also set $CC = 0$, $AA = 1$, $BB = 0$, $CB = 1$, in which case S_1–S_3 true are and $\neg CB$ is false. Therefore, we cannot conclude that Charlie is not a builder either.

3. Identify all of the formulas below that are satisfiable but not tautologies.

F_1: $(A \to B) \wedge \big((A \wedge \neg B) \vee \neg(C \to \neg A)\big)$

F_2: $(A \to B) \vee \big((A \vee \neg B) \wedge \neg(C \to \neg A)\big)$

F_3: $\big((C \to \neg B) \wedge (B \vee A)\big) \wedge \neg(C \to A)$

F_4: $(A \wedge B) \to \big((C \vee A) \wedge \neg B\big)$

Answer. A formula is *satisfiable* if there is an assignment of truth-values to its variables that makes the formula true. A formula is a *tautology* if every assignment of truth-values to its variables makes the formula true.

F_1, F_2 and F_4 are satisfiable but not tautologies. Indeed:

F_1: setting $A = 1,\ B = 1,\ C = 1$ makes $F_1 = 1$, while $A = 0,\ B = 0,\ C = 0$ makes $F_1 = 0$.

F_2: setting $A = 1,\ B = 1,\ C = 1$ makes the formula 1, $\ A = 1,\ B = 0,\ C = 0$ makes the formula 0.

F_3: there is no truth-value assignment making the formula 1, so it is not satisfiable. Here is the truth table:

A	B	C	$C \to \neg B$	$B \vee A$	$(C \to \neg B) \wedge (B \vee A)$	$\neg(C \to A)$	F_3
1	1	1	0	1	0	0	0
1	1	0	1	1	1	0	0
1	0	1	1	1	1	0	0
1	0	0	1	1	1	0	0
0	1	1	0	1	0	1	0
0	1	0	1	1	1	0	0
0	0	1	1	0	0	1	0
0	0	0	1	0	0	0	0

F_4: $A = 1,\ B = 0,\ C = 0$ makes the formula 1, $\ A = 1,\ B = 1,\ C = 1$ makes the formula 0.

4. Translate the given statements into propositional logic using the atomic propositions provided:

– You cannot edit a protected Wikipedia entry unless you are an administrator. Express your answer in terms of atomic propositions E: 'You can edit a protected Wikipedia entry' and A: 'You are an administrator.'

– You are eligible to be President of the USA only if you are at least 35 years old, were born in the USA, or at the time of your birth both of your parents were citizens, and you have lived at least 14 years in the country. Express your answer in terms of E: 'You are eligible to be President of the USA,' A: 'You are at least 35 years old,' B: 'You were born in the USA', P: 'At the time of your birth, both of your parents where citizens,' and R: 'You have lived at least 14 years in the USA.'

Answer. Try to rephrase the sentences to make their meaning clear to you. For example, the first one can be paraphrased as 'If you are not an administrator, then you cannot edit a protected Wikipedia entry'. In general, 'X unless Y' means 'if $\neg Y$, then X'. (See, for example, `https://www.cs.miami.edu/home/geoff/Courses/TPTPSYS/Practicum/EnglishToLogic.shtml`.)

– $\neg A \to \neg E$, which is equivalent to $E \to A$;

– $E \to A \wedge (B \vee (P \wedge R))$.

5. Determine whether the formulas $A \wedge (B \oplus C)$ and $(A \wedge B) \oplus (A \wedge C)$ are equivalent, where $\oplus$ denotes 'exclusive OR' (or 'XOR').

Answer. The formulas are equivalent because they both have the same truth-table:

A	B	C	the formulas
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

There is also an alternative solution using 'equivalent transformations' (see page 41 in FoC-I). We know that

$$X \oplus Y \equiv (X \wedge \neg Y) \vee (\neg X \wedge Y).$$

So

$$A \wedge (B \oplus C) \equiv \underline{A \wedge ((B \wedge \neg C) \vee (\neg B \wedge C))}.$$

On the other hand,

$$\begin{aligned}
(A \wedge B) \oplus (A \wedge C) &\equiv ((A \wedge B) \wedge \neg(A \wedge C)) \vee (\neg(A \wedge B) \wedge (A \wedge C)) \\
&\equiv ((A \wedge B) \wedge (\neg A \vee \neg C)) \vee ((\neg A \vee \neg B) \wedge (A \wedge C)) \\
&\equiv ((\neg A \wedge (A \wedge B)) \vee (\neg C \wedge (A \wedge B))) \vee ((\neg A \wedge (A \wedge C)) \vee (\neg B \wedge (A \wedge C))) \\
&\equiv (\neg C \wedge (A \wedge B)) \vee (\neg B \wedge (A \wedge C)) \\
&\equiv (A \wedge (B \wedge \neg C)) \vee (A \wedge (\neg B \wedge C)) \\
&\equiv \underline{A \wedge ((B \wedge \neg C) \vee (\neg B \wedge C))}
\end{aligned}$$

6. Construct a Boolean formula that realises the Boolean function given by the following truth-table:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

(a) use the connectives $\vee, \wedge, \neg$ only;

(b) find a simplest possible formula.

Answer. The method explained on page 38 of the FoC-I slides gives the following formula:

$$(\neg x_1 \wedge x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \neg x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3).$$

It can be simplified as follows. Using distributivity of $\wedge$ over $\vee$ (from right to left), we obtain:

$$(\underbrace{\neg x_1 \wedge x_2} \wedge \neg x_3) \vee (\underbrace{\neg x_1 \wedge x_2} \wedge x_3) \quad \equiv \quad (\neg x_1 \wedge x_2) \wedge (\neg x_3 \vee x_3) \quad \equiv \quad (\neg x_1 \wedge x_2).$$

Similarly,

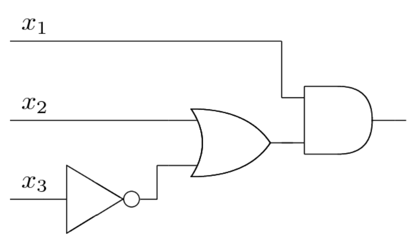
$$(x_1 \wedge \neg x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3) \quad \equiv \quad (x_1 \wedge x_3) \wedge (\neg x_2 \vee x_2) \quad \equiv \quad (x_1 \wedge x_3).$$

Thus, we obtain

$$(\neg x_1 \wedge x_2) \vee (x_1 \wedge x_3).$$

Alternatively, the same formula can be found by looking at the truth-table: (i) in the upper half, $f = 1$ iff $\neg x_1 \wedge x_2$ is true; (ii) in the lower half, $f = 1$ iff $x_1 \wedge x_3$ is true.

7. Consider the Boolean function $f(x_1, x_2, x_3)$ realised by the following Boolean circuit:



Construct the truth-table for this function and represent it by means of a Boolean formula.

Answer. The truth-table is given below:

x_1	x_2	x_3	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Thus, $f = 1$ iff x_1 is 1 and it is not the case that both $x_2 = 0$ and $x_3 = 1$. This condition can be written as the Boolean formula $(\neg x_3 \vee x_2) \wedge x_1$ giving the required answer.

8. Simplify the Boolean formula $\neg(A \wedge B) \wedge (\neg A \vee B) \wedge (\neg B \vee B)$.

Answer. The formula $\neg(A \wedge B) \wedge (\neg A \vee B) \wedge (\neg B \vee B)$ can be simplified using the following steps:

- the law of the excluded middle: $\neg(A \wedge B) \wedge (\neg A \vee B)$;
- the De Morgan law: $(\neg A \vee \neg B) \wedge (\neg A \vee B)$;
- the distributive law: $\neg A \vee (\neg B \wedge B)$;
- the law of contradiction: $\neg A$.

You can obtain the same result by constructing the truth-table for the given formula and observing that the truth-values of the formula coincide with the truth-values of $\neg A$.

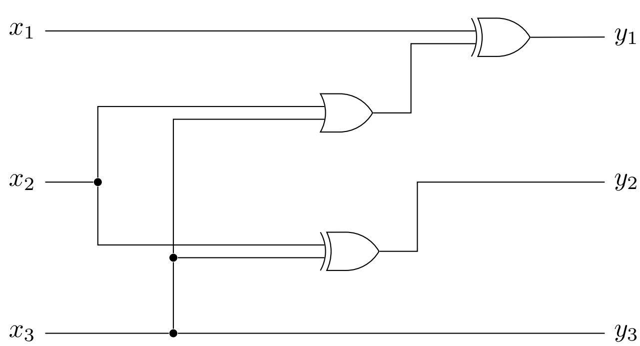
9. Design a Boolean circuit to input a 3-bit value and output its two's complement value.

Answer. The two's complement of an n-bit binary number N is $2^n - N$. To compute it, use the algorithm in FoC-I: invert the bits and add 1. The truth-table for $n = 3$ is as follows:

x_1	x_2	x_3	y_1	y_2	y_3
0	0	0	0	0	0
0	0	1	1	1	1
0	1	0	1	1	0
0	1	1	1	0	1
1	0	0	1	0	0
1	0	1	0	1	1
1	1	0	0	1	0
1	1	1	0	0	1

Given the input 3-bit number x_1, x_2, x_3, the outputs y_1, y_2, y_3 can be computed as follows:

$$y_3 = x_3, \qquad y_2 = x_2 \oplus x_3, \qquad y_1 = x_1 \oplus (x_2 \vee x_3).$$



10. Is the set $\{\wedge, \vee\}$ functionally complete? (Hint: is it possible to realise $\neg A$ as a formula with $\wedge$ and $\vee$ only?)

Answer. Suppose $\{\wedge, \vee\}$ is functionally complete. Then $\neg$ should be expressible via $\wedge$ and $\vee$, that is, there is a formula φ with connectives $\wedge$ and $\vee$ only such that $\varphi \equiv \neg A$. Consider an assignment of 1 to all of the variables in φ. Then the truth-value of φ must be 1 (because $(1 \wedge 1) = (1 \vee 1) = 1$), while the truth-value of $\neg A$ is 0, contrary to $\varphi \equiv \neg A$. Therefore, such a φ cannot exist, and so $\{\wedge, \vee\}$ is not functionally complete.