Additional Practice Exercises: Recursion and Induction

1. Call a truth function $f \in \text{Bool}\, n$ **positive** iff $f(\top, \top, \ldots, \top) = \top$. (That is, if the input is a sequence of all True, then the output is True.)

   (a) Show by induction that if A is a sentence with no occurrences of $\bot$, then $[\![A]\!]$ is positive.

   (b) Conclude that the sentences in $\text{Prop}\, n$ with no occurrences of $\bot$ are *not* expressively complete: there are truth-functions in $\text{Bool}\, n$ which are not expressed by any sentence in $\text{Prop}\, n$ without $\bot$.

2. Let A be any sentence in $\text{Prop}\, 1$ that contains no conditionals. Show that for any sentence B, $A[B] \vDash B$.

3. For any sentence $A \in \text{Prop}\, S$, let cA be the number of conjunctions that occur in A.

   (a) Give a recursive definition of c.

   (b) Show by induction that for any sentences $A \in \text{Prop}\, 1$ and $B \in \text{Prop}\, S$, $c(A[B]) \geq cA$.

   (c) Let $\#(s, A)$ be the *number of occurrences of* s in A, for a sentence letter $s \in S$ and a sentence $A \in \text{Prop}\, S$. Here are some examples:

$$\begin{aligned} \#(p, q \to (p \wedge p)) &= 2 \\ \#(p, \bot \to q) &= 0 \\ \#(p, p) &= 1 \end{aligned} \tag{1}$$

   Give a recursive definition of $\#(s, A)$.

   (d) Show that if $A \in \text{Prop}\, 1$, then

$$c(A[B]) = c(A) + c(B) \cdot \#(p, A) \tag{2}$$

   (That is: the number of conjunctions in $A[B]$ is the number of conjunctions in A plus (the number of conjunctions in B times the number of occurrences of the sentence letter p in A). For this problem, go ahead and assume basic facts about addition and multiplication, even though we may not have proved all of them. In particular, the *distributive law* for multiplication might be useful:

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{for any } a, b, c \in \mathbb{N} \tag{3}$$

Also: convince yourself that an analogous fact holds for the *complexity* of $A[B]$.)

4. (These exercises show a simple form of the "interpolation theorem", which is about how the semantics of sentences are constrained by the sentence letters that occur in them.)

   (a) Give a recursive definition of a function $\mathrm{sl} : \mathsf{Prop}\, S \to PS$ that takes each sentence A to the set of the sentence letters that occur in A. Here are some examples:

$$\begin{aligned} \mathrm{sl}(p \to q) &= \{p, q\} \\ \mathrm{sl}((p \wedge \bot) \to p) &= \{p\} \\ \mathrm{sl}(\bot \wedge (\bot \to (\bot \to \bot))) &= \{\} \end{aligned} \tag{4}$$

   (b) Let A be any sentence and s any sentence letter such that $s \notin \mathrm{sl}\, A$. Suppose that M and M' are truth-assignments such that for every sentence letter $t \neq s$, $Mt = M't$. Show that $[\![A]\!]_M = [\![A]\!]_{M'}$.

   (c) Let $A, B \in \mathsf{Prop}\, S$, and suppose A and B have no sentence letters in common. More specifically, suppose

$$\begin{aligned} p &\notin \mathrm{sl}\, A \\ \mathrm{sl}\, B &= \{p\} \end{aligned} \tag{5}$$

   Suppose also that $A \vDash B$. Show that either $A \vDash \bot$ or $\vDash B$.

   Hint. To begin, suppose that there is a model in which A is true, and also that there is a model in which B is false.

   Can you come up with any interesting generalizations of this fact?