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 \( \llbracket A \rrbracket \) 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 \( c_A \) 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 \( B \in \text{Prop}_S \), \( c(A[B]) \geq c_A \).

   (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:

   \[
   \begin{align*}
   #(p, q \rightarrow (p \land p)) &= 2 \\
   #(p, \bot \rightarrow q) &= 0 \\
   #(p, p) &= 1
   \end{align*}
   \]

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

   (d) Show that if \( A \in \text{Prop}_1 \), then

   \[
   c(A[B]) = c(A) + c(B) \cdot #(p, A) \quad (2)
   \]

   (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:

   \[
   a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{for any } a, b, c \in \mathbb{N} \quad (3)
   \]
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 $sl : \text{Prop} S \rightarrow PS$ that takes each sentence $A$ to the set of the sentence letters that occur in $A$. Here are some examples:

$$sl(p \rightarrow q) = \{p, q\}$$
$$sl((p \land \bot) \rightarrow p) = \{p\}$$
$$sl(\bot \land (\bot \rightarrow (\bot \rightarrow \bot))) = \{\} \quad (4)$$

(b) Let $A$ be any sentence and $s$ any sentence letter such that $s \notin sl A$. 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 $A, B \in \text{Prop} S$, and suppose $A$ and $B$ have no sentence letters in common. More specifically, suppose

$$\rho \notin sl A$$
$$sl B = \{\rho\} \quad (5)$$

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

*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?