Please supply proofs for four of the following questions. Students enrolled in 205B should include problems 1 and 6 among their four. You may work on the homework questions together, but you must hand in your own assignment.

1. Show that for any \( n < \omega \), \( \{ p_0 \} \cup \{ p_{k-1} \supset p_k : 1 \leq k \leq n \} \models p_n \). Now suppose \( \Gamma \) contains \( p_0 \) plus an “infinitely long chain” of conditionals, i.e., \( \Gamma = \{ p_0, p_0 \supset p_1, p_1 \supset p_2, \ldots, p_n \supset p_{n+1}, \ldots, p_\alpha \supset p_{\alpha+1} \} \). Use the compactness theorem to prove or disprove: \( \Gamma \models p_{\alpha+1} \). \([Hint: \Gamma \) is a poorly described set!\]
   a. \emph{Extra and utterly optional:} Use the completeness theorem to give another proof of your answer.
   b. \emph{Extra and inconceivably unnecessary:} Use set theory to give another explanation of your answer.

2. Prove or disprove: For all \( \phi \) and all \( \Gamma \), either \( \Gamma \models \phi \) or \( \Gamma \models \neg \phi \).

3. Prove or disprove: \( \Gamma \models \phi \) iff there is some finite \( \Gamma' \subseteq \Gamma \) such that \( \Gamma' \models \phi \).

4. Consider the “Nand” operator: \( \phi \mid \psi \), where \( v[\phi \mid \psi] = \top \) iff \( v[\phi] = v[\psi] = \bot \). Show that a language like L, but where \( \{ \mid \} \) replaces \( \{ \neg, \supset \} \) is truth-functionally complete.

5. Consider the “Nor” operator: \( \phi \downarrow \psi \), where \( v[\phi \downarrow \psi] = \top \) iff \( v[\phi] = v[\psi] = \bot \). Show that a language like L, but where \( \{ \downarrow \} \) replaces \( \{ \neg, \supset \} \) is truth-functionally complete.

6. Let \( M[ , , ] \) be a three-place Boolean function that takes the truth-value assigned to the \emph{minority} of the three component sentences. Let \( \bot \) be the zero-place Boolean function that always takes the value \( \bot \). Show that a language like L, but where \( \{ M[ , , ], \bot \} \) replaces \( \{ \neg, \supset \} \) is truth-functionally complete.

7. **For any \( n \in \omega \), let \( f \) be an \( n \)-ary Boolean function. Say that \( f \) is truth-functionally complete iff a language like ours except that \( \{ f \} \) replaces \( \{ \neg, \supset \} \) is truth-functionally complete. For a fixed arbitrary \( n \), can you place some interesting upper or lower boundaries on the number of truth-functionally complete \( n \)-ary Boolean functions?