Reading Please read Sections 2.1 through 2.4 by the time this assignment is due. I suggest that you only read one section at a sitting and that you think about all of the exercises – assigned or not – to help make sure that you are understanding all the concepts as you read.

Problems

1. [20 pts.] Another take on the compactness theorem for propositional logic is that every infinite binary tree has an infinite branch.

   Nodes of a binary tree are naturally named by binary strings: The empty string, ϵ, is the root. Each node at height n is named by a binary string w of length n. The left child of w is $w0$ and the right child is $w1$.

   Let T be such a binary tree. T has the property that if $w \in \{0,1\}^n$ is a node in the tree, then so are all its ancestors, i.e., the prefixes of w.

   (a) Prove that if T is an infinite binary tree, i.e, it has infinitely many nodes, then it has an infinite branch, σ, i.e, an infinite sequence, $\sigma = (\epsilon, a_1, a_1a_2, a_1a_2a_3, \ldots)$ where each $a_i \in \{0,1\}$. [Hint: define σ inductively: since T is infinite, either the subtree whose root is 0 or the subtree whose root is 1 must be infinite. Keep going in this way.]

   Now suppose that S is a set of propositional formulas every finite subset of which is satisfiable. Assume that S has no two formulas that are equivalent – otherwise we can just delete all but the first formula of each equivalence class. Define $S_n = \{\alpha \in S \mid \text{atom}(\alpha) \subseteq \{A_1, \ldots, A_n\}\}$.

   Since S_n is a finite subset of S it must be satisfiable by assumption. Let's encode any truth assignment that is defined exactly on the atoms $\{A_1, \ldots, A_n\}$ as a binary string of length n. For example, the string 01011 is the truth assignment that assigns 1 to A_2, A_4, A_5 and 0 to A_1, A_3. Now define the tree of all satisfying truth assignments of the S_n's as follows: $T_S = \{w \mid \exists n \exists w \in \{0,1\}^n; w \models S_n\}$.

   (b) Show that T_S is an infinite binary tree. Conclude from (a) that T_S has an infinite branch, σ and that σ is a satisfying assignment of S. Thus you have proved the compactness theorem for propositional logic.

2. [15 pts.] Exercise 42, p. 43: Let Free(F) be the set of all variables that occur freely in F. Define Free(F) formally by induction on terms and then on formulas.

3. [15 pts.] Exercise 45, p. 49: Consider the following sentences R, S, T which express that the predicate P is reflexive, symmetric, and transitive:

$$
\begin{aligned}
R &= \forall x P(x,x) \\
S &= \forall xy (P(x,y) \rightarrow P(y,x)) \\
T &= \forall xyz (P(x,y) \wedge P(y,z) \rightarrow P(x,z))
\end{aligned}
$$

Show that these sentences are independent by constructing three structures that satisfy each possible pair of the sentences but not the third.

[Note that $\wedge$ and $\vee$ bind more tightly than $\rightarrow$, so I didn't use extra parentheses in the definition of T.]

4. [15 pts.], cf. Exercise 49, p. 50. Find two satisfiable sentences F_1, F_2, such that every model of F_1 or of F_2 has a universe of cardinality at least three. F_1 should be in predicate logic without equality, and F_2 should be in predicate logic with equality, and F_2 should have no function symbols and no predicate symbols except $=$.

5. [15 pts.] Exercise 55, p. 52: show that the following pairs of formulas are not equivalent by constructing structures that satisfy one of them but not the other:

$$(\forall x P(x) \vee \forall x Q(x)) \quad \not\equiv \quad \forall x (P(x) \vee Q(x))$$
$$(\exists x P(x) \wedge \exists x Q(x)) \quad \not\equiv \quad \exists x (P(x) \wedge Q(x))$$

6. [20 pts.] Exercise 57, p. 53: Prove that $\exists u \forall v P(v, u) \models \forall x \exists y P(x, y)$, but that these formulas are not equivalent.