Boolos and Jeffrey - HW4

David Maldonado, *david.m.maldonado@gmail.com*

October 21, 2014

In the following theorems Q stands for *either* quantifier and Q' for its counterpart. v stands for a quantified variable, and F and G for first-order logical formulas with no free variables.

1 some equivalence proofs...

Theorem 1.1. $\neg QvF \cong Q'v\neg F$

Proof. We'll begin with the first case:

$$\neg\forall vF \cong \exists v\neg F \tag{1}$$

The implication $\neg\forall vF \implies \exists v\neg F$ is proven simply by noting that if we assume $\neg\forall vF$ to be **true** that means there exists at least one term in a model that makes $\neg F$ **true**, which is precisely the statement on the right-hand side.

The converse implication $\neg\forall vF \impliedby \exists v\neg F$ is proven in the same way by assuming $\exists v\neg F$ to be **true**. It follows directly that because there is at least one term in a model that makes $\neg F$ **true** not all terms make F **true** which is the statement on the left-hand side.

For the second case:

$$\neg\exists vF \cong \forall v\neg F \tag{2}$$

The implication $\neg\exists vF \implies \forall v\neg F$ is proven by first assuming $\neg\exists vF$ is **true**. With this assumption we can say that there *does not* exist a term v such that F is **true**, this leads to the right-hand statement that for all terms $\neg F$ is **true**.

The converse implication $\neg\exists vF \impliedby \forall v\neg F$ is proven by assuming $\forall v\neg F$ is **true**. Now we can see that for all terms $\neg F$ is **true**, therefore there *does not* exist a term that makes F **true**, which is the left-hand statement. $\square$

2 proof of prenex normal form

Theorem 2.1. *Where* ***prenex normal form*** *is a logical formula where all the quantifiers are written as a string at the front and range over the quantifier-free matrix, every formula in first-order logic has an equivalent prenex normal form.*

Proof. We will proceed by induction on the complexity of the formula. Let us first agree on the following equivalences:

$$\neg QvF \cong Q'v\neg F \tag{1}$$

$$QvF \wedge G \cong Qv(F \wedge G) \tag{2}$$

$$G \wedge QvF \cong Qv(G \wedge F) \tag{3}$$

$$QvF \vee G \cong Qv(F \vee G) \tag{4}$$

$$G \vee QvF \cong Qv(G \vee F) \tag{5}$$

$$QvF \to G \cong Q'v(F \to G) \tag{6}$$

$$G \to QvF \cong Qv(G \to F) \tag{7}$$

and the **principle of substitution of equivalents**:

$$QvF \cong QwF_vw. \tag{8}$$

Our *base case* is the logical formula with no quantified variables. This formula is automatically prenex normal form. The *inductive step* assumes that every formula F with n or fewer logical symbols is equivalent to a formula F' in prenex normal form. Now we consider a formula with $n+1$ logical symbols. This well-formed formula is one of the following:

$$\neg\phi$$

$$\phi \wedge \psi$$

$$\phi \vee \psi$$

$$\phi \to \psi$$

By repeated applications of (1) - (7) and the usage of (8) to change bound variables it is possible to convert any of these formulas into prenex normal form as we'll show below.

Case 1. For $\neg\phi$ if ϕ has no quantifiers it is in prenex normal form. If ϕ has one or more quantifiers by the induction hypothesis it is in prenex normal form $(Q_1v \dots Q_nvF)$. The negation can be moved to the matrix by n applications of (1) using (8) to change bound variables when needed.

Case 2. For $\phi \wedge \psi$ if ϕ and ψ both have no quantifiers $\phi \wedge \psi$ is in prenex normal form. If either (or both) have quantifiers mixed applications of (2) and (3), depending on which side the quantifier is on, and use of (8) to change bound variables when needed will yield a formula in prenex normal form.

Case 3. For $\phi \vee \psi$ the proof is the same as *Case 2* except using (4) and (5) as the main operations.

Case 4. For $\phi \rightarrow \psi$ the proof is the same as *Case 2* and *Case 3* except using (6) and (7) as the main operations.

$\square$