Phil 201
Practice Final

Extra Credit

Note: We provided the wrong definition of a pure monadic sentence. What we defined was actually a simple monadic sentence. Here are the correct definitions:

A simple monadic sentence is one of the form $Qv\,\Phi(v)$ where Q is a quantifier (either $\forall$ or $\exists$) and Φ contains no quantifiers and only 1-place predicates.

A pure monadic sentence is a truth functional combination of simple monadic sentences.

1.) Claim: Let A be a pure monadic sentence. If A has a model of size n, then it has a model of size m for any finite $m > n$.

Proof: First put A in "disjunctive normal form." Here what we mean is that $A \equiv A_{DNF} = \bigvee_{i=0}^{k} \left(\bigwedge_{j=0}^{\ell} (\neg) S_j \right)_i$

where each S_j is a simple monadic sentence. That is, A_{DNF} is a finite disjunction of finite conjunctions of simple or negated simple monadics.

Assume that A has a model, $\mathfrak{M}$, with n elements. It follows that $\mathfrak{M}$ is a model of at least one disjunct in A_{DNF}. Choose one such disjunct. Use the quantifier duality rules to convert each negated simple monadic into a simple monadic. Therefore the disjunct looks like

$$Q_1 v_1 \Phi_1(v_1) \wedge \ldots \wedge Q_p v_p \Phi_p(v_p)$$

where each conjunct is a simple monadic sentence. Now, consider the model $\mathfrak{M}$. It looks like this:

$$DoQ = \{1, \ldots, n\}$$

$$\left.\begin{array}{l} Ext(F_1) \subseteq DoQ \\ \vdots \\ Ext(F_r) \subseteq DoQ \end{array}\right\} \text{ for each } F_i \text{ occuring in } A$$

$$\left.\begin{array}{l} Ref(a_1) \in DoQ \\ \vdots \\ Ref(a_s) \in DoQ \end{array}\right\} \begin{array}{l}\text{for each name} \\ a_i \text{ occuring in} \\ A\end{array}$$

We extend $\mathfrak{M}$ to an m element model as follows: first add $m-n$ new elements to DoQ. Next choose new names, $a_{s+1}, \ldots, a_m$, for every object in DoQ without one. Leave the referent of $a_1, \ldots, a_s$ unchanged. For the predicates, if F_i occurs in an existential simple monadic conjunct, leave $Ext(F_i)$ unchanged. Otherwise consider instances of each universal simple monadic conjunct for each name $a_1, \ldots, a_m$. To the original extension of each predicate occuring in a universal conjunct, add the referent of $a_j \in \{a_{s+1}, \ldots, a_m\}$ for each a_j that the predicate applies to. The result of this entire process is an m-element model of the chosen disjunct and thus of A.

$\square$

2.) <u>claim</u>: If A is a consistent pure monadic sentence, then it has a finite model.

<u>Proof</u>: We will use the fact that algorithms A, B, and C provide a decision procedure for any pure monadic sentence. If A is consistent, then the output of the joint algorithm is a model of A. By construction, such a model is finite.

To see this recall the steps involved in applying the algorithms. First, using algorithm C as in problem 1.), we find A_{DNF}, Use the quantifier duality rules to convert each disjunct into a conjunction of simple monadic sentences, then feed each disjunct into algorithm B.

Let the disjunct have k existential pieces and l universal pieces. Algorithm B introduces new names $a_1, \ldots, a_k$, and replaces each existential conjunct, $\exists v_i \Phi_i v_i$ $i=1,\ldots,k$ with an instance $\Phi_i a_i$ $i=1,\ldots,k$. For each universal conjunct, $\forall u_j \Psi_j u_j$ $j=1,\ldots,l$ the algorithm takes k instances for each name $a_1, \ldots, a_k$ (i.e. $\Psi_j a_1, \ldots, \Psi_j a_k$ for each $j=1,\ldots,l$).

These pieces are fed into algorithm A which finds an interpretation (using the method of truth tables) that makes each conjunct $\Phi_i a_i$ $i=1,\ldots,k$ and $\Psi_j a_i$ $i=1,\ldots k$, $j=1,\ldots,l$ true. The model that is constructed by algorithm A has one element for each name in A and for each name introduced by algorithm B. Since the number of names originally in A is finite, and the number of quantifiers in A_{DNF} is finite, and thus the number of new names introduced by algorithm B is finite, the model ultimately produced by algorithm A is finite.

$\square$

3.) <u>Claim:</u> $\{\neg, \leftrightarrow\}$ is not complete relative to truth-functions of two variables.

<u>Proof:</u> Consider the two-variable truth function $P \rightarrow Q$. Its truth table looks like this:

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

Note that it has an unequal number of T's and F's in the main column (3 T's and 1 F). We will show that any truth-function of two variables definable from $\{\neg, \leftrightarrow\}$ will always have an equal number of T's and F's in the main column of its truth table.

Let Σ be the set of all propositional sentences constructed from P, Q and $\neg, \leftrightarrow$. We define Σ inductively as follows:

Base: $P, Q \in \Sigma$

Gen: (i) If $X \in \Sigma$, then $\neg X \in \Sigma$

(ii) If $X, Y \in \Sigma$, then $X \leftrightarrow Y \in \Sigma$

First, it's easy to check that $\neg P$ and $P \leftrightarrow Q$ have an equal number of T's and F's in their respective truth tables. Now, assume that X, Y are functions of both P and Q with similarly symmetric truth tables. We have four possible valuations of P and Q TT, TF, FT, FF. Let $v_1 - v_4$ be arbitrary labels for these valuations. By hypothesis we have

P, Q	X
v_1	T
v_2	T
v_3	F
v_4	F

, thus

P, Q	$\neg X$
v_1	F
v_2	F
v_3	T
v_4	T

and so $\neg X$ has an equal number of

T's and F's. Similarly by hypothesis

P, Q	X	Y
V_1	T	T
V_2	T	F
V_3	F	F
V_4	F	T

, thus

P, Q	$X \Longleftrightarrow Y$
V_1	T
V_2	F
V_3	T
V_4	F

and so $X \Longleftrightarrow Y$ has an equal number of T's or F's.

Hence $P \rightarrow Q$ is not definable in terms of P, Q and $\neg, \Longleftrightarrow$, so $\{\neg, \Longleftrightarrow\}$ is not truth-functionally complete.

$\square$