Solutions to CS511 Homework 12

Nicholas Ikechukwu - U71641768

December 05, 2024

Exercise 1. Do the exercise on page 42.

Exercise: Define $X \sim Y$ differently in second-order logic by asserting the existence of a unary function F from X to Y which is both injective and surjective.

Solution to: Defining $X \sim Y$ in Second-Order Logic

If we want to define $X \sim Y$ using a bijective function $F : X \to Y$, we can express it as follows:

$$X \sim Y \equiv \exists F.(\forall x_1, x_2 \in X.(F(x_1) = F(x_2) \to x_1 = x_2)) \wedge (\forall y \in Y.\exists x \in X.F(x) = y)$$

Let's' break the formula down into two separate parts:

1. **Injectivity**: Here, the condition $\forall x_1, x_2 \in X.(F(x_1) = F(x_2) \to x_1 = x_2)$ ensures that F is injective, meaning each element in X maps to a unique element in Y.

2. **Surjectivity**: For this, the condition $\forall y \in Y.\exists x \in X.F(x) = y$ ensures that F is surjective, meaning every element in Y has a pre-image in X.

The conditions, together, ensure that F is a bijection from X to Y, thereby establishing the equivalence relation $X \sim Y$.

Exercise 2. both parts of the exercise on page 47.

Exercise :

1. Define a second-order sentence $\Psi_{\text{countable-infty}}$ such that $A \models \Psi_{\text{countable-infty}}$ if and only if A is countably infinite.

2. Define a second-order sentence $\Psi_{\text{uncountable}}$ such that $A \models \Psi_{\text{uncountable}}$ if and only if A is uncountably infinite.

Note that $\Psi_{\text{countable-infty}}$ and $\Psi_{\text{uncountable}}$ in this exercise are sentences, i.e., closed well-formed formulas (wff's) which do not contain any free variables.

Solution:

Part 1. Countably Infinite Set :

If we want to define $\Psi_{\text{countable-infty}}$, basically, we need to express that a set A is countably infinite. We do this by simply ensuring that A is infinite and every infinite subset of A has a bijection to A. The second-order sentence is:

$$\Psi_{\text{countable-infty}} = \Phi_{\text{infty}}(A) \wedge (\forall X \subseteq A.(\Phi_{\text{infty}}(X) \rightarrow (X \sim A)))$$

Where: - $\Phi_{\text{infty}}(A)$ asserts that A is infinite. - $X \sim A$ shows there exists a bijection between X and A.

Part 2. Uncountably Infinite Set :

If we want to define $\Psi_{\text{uncountable}}$, we also want to express that a set A is uncountably infinite by negating the condition for countability:

$$\Psi_{\text{uncountable}} = \Phi_{\text{infty}}(A) \land (\neg(\forall X \subseteq A.(\Phi_{\text{infty}}(X) \to (X \sim A))))$$

Where: - The negation of the second part ensures that there exists at least one infinite subset of A that does not have a bijection with A, indicating uncountability.

Exercise 3. Do part 1 only of the exercise on page 50.

Exercise : Write a second-order wff $\theta(x, y)$ such that:

$$\theta(x, y) \iff \text{"no binary predicate } Y \text{ can discern } x \text{ and } y\text{"}$$

Your task here is to write a wff of second-order logic modeling the English phrase to the right of "iff".

Solution:

For us to express that no binary predicate Y can discern x and y, we basically need to ensure that for any binary relation Y, the truth value of $Y(x, z)$ matches $Y(y, z)$ for all possible z, and similarly, $Y(z, x)$ matches $Y(z, y)$. We can express this as:

$$\theta(x, y) = \forall Y.(\forall z.(Y(x, z) \leftrightarrow Y(y, z)) \wedge (Y(z, x) \leftrightarrow Y(z, y)))$$

In the formula above: - This first part, $\forall z.(Y(x, z) \leftrightarrow Y(y, z))$, ensures that for any element z, if x is related to z, then y must also be related to z, and vice versa.

- The second part, $(Y(z, x) \leftrightarrow Y(z, y))$, inturn, ensures symmetry in the sense that if any element z is related to x, it must also be related to y, and vice versa.

Clearly, the above ensures that no binary predicate can distinguish between the elements x and y.

PROBLEM 1. Open [LCS, page 165]: Do parts (a), (b), and (c) only in Exercise 2.6.5.

Hint: Part (c) is already done in Lecture Slides 32, pp 20-24. You may want to do it differently!

Exercise :

Let P and R be predicate symbols of arity 2. Write formulas of existential second-order logic of the form $\exists P\, \psi$ that hold in all models of the form $M = (A, R^M)$ if and only if:

(a) R contains a reflexive and symmetric relation.

(b) R contains an equivalence relation.

(c) There is an R-path that visits each node of the graph exactly once – such a path is called Hamiltonian. (Using a different approach from Lecture Slides 32, pp 20-24).

Solution:

Part a: Reflexive and Symmetric Relation

For us to express that R contains a reflexive and symmetric relation, we define:

$$\exists P.(\forall x.P(x,x)) \wedge (\forall x \forall y.(P(x,y) \rightarrow P(y,x)))$$

Our formula asserts:

- Reflexivity: $\forall x.P(x,x)$

- Symmetry: $\forall x \forall y.(P(x,y) \rightarrow P(y,x))$

Part b: Equivalence Relation

An equivalence relation is reflexive, symmetric, and transitive. Therefore:

$$\exists P.(\forall x.P(x,x)) \land (\forall x\forall y.(P(x,y) \to P(y,x))) \land (\forall x\forall y\forall z.((P(x,y) \land P(y,z)) \to P(x,z)))$$

This formula includes:

- Reflexivity: $\forall x.P(x,x)$

- Symmetry: $\forall x\forall y.(P(x,y) \to P(y,x))$

- Transitivity: $\forall x\forall y\forall z.((P(x,y) \land P(y,z)) \to P(x,z))$

Part c.

We can express the existence of a Hamiltonian path using a different approach thus:

$$\exists P. (\forall x \exists ! y. P(x, y)) \land (\forall x \exists ! z. P(z, x)) \land (\forall x \forall y. (P(x, y) \to R(x, y)))$$

Where:

- $P(x, y)$ is a binary predicate indicating that y is the successor of x in the path.

- **Unique Successor**: For every node x, there exists exactly one node y such that $P(x, y)$.

- **Unique Predecessor**: For every node x, there exists exactly one node y such that $R(x, y)$.

- **Path Condition**: For every pair of nodes x, y, if $P(x, y)$, then there must be an edge between them in $R(x, y)$.

ON LEAN-4

Solutions in one file at: https://github.com/nich-ikech/CS511-hw-macbeth/blob/main/cs511HwSolutions/hw12/hw12_nicholas_ikechukwu.lean

Exercise 4. From Macbeth's book:

Solutions

https://github.com/nich-ikech/CS511-hw-macbeth/blob/main/cs511HwSolutions/hw12/hw12_nicholas_ikechukwu.lean

Exercise 5. From Macbeth's book

Solutions

https://github.com/nich-ikech/CS511-hw-macbeth/blob/main/cs511HwSolutions/hw12/hw12_nicholas_ikechukwu.lean

Exercise 6. From Macbeth's book:

Solutions

https://github.com/nich-ikech/CS511-hw-macbeth/blob/main/cs511HwSolutions/hw12/hw12_nicholas_ikechukwu.lean

PROBLEM 2. From Macbeth's book

Solutions

https://github.com/nich-ikech/CS511-hw-macbeth/blob/main/cs511HwSolutions/hw12/hw12_nicholas_ikechukwu.lean