Chapter 9

Exercise 9.1

(a) $Srj \wedge Skj$; r: Keren, k: Keziah, j: Jemima, S: – is a sister of –
(b) $\forall x(Mx \rightarrow Ox)$; M: – is a myriapod, O: – is oviparous
(c) $\exists x(Cx \wedge Nx \wedge Hfx)$; f: Fred, C: – is a car, N: – is new, H: – has –
(d) $\neg\forall x(Sx \rightarrow Lxl)$; l: logic; S: – is a student, L: – loves –
(e) $\forall x((Sx \wedge Lxl) \rightarrow \exists yLxy)$; l: logic; S: – is a student, L: – loves –

Exercise 9.2

Let the model M be given by $D = \{\text{Rome}, \text{Paris}\}$ and $V(F) = \{\text{Rome}\}$. By clause (a) of definition 9.2, $M, g' \models Fx$ holds for every assignment function g' that maps x to Rome, because then $g'(x) \in V(F)$. By clause (h) it follows that $M, g \models \exists xFx$ for every assignment function g. By clause (a) again, $M, g' \not\models Fx$ for every assignment function g' that maps x to Paris. By clause (g), it follows that $M, g \not\models \forall xFx$ for every assignment function g. So $\exists xFx$ is true (in M) relative to every assignment function while $\forall xFx$ is false relative to every assignment function. By clause (e) it follows that $\exists xFx \rightarrow \forall xFx$ is false in M relative to every assignment function.

Exercise 9.3

For both cases, use Fx as the sentence A, and $\neg Fx$ as B, and consider a model in which F applies to some but not to all individuals. Both Fx and $\neg Fx$ are then true relative to some assignment functions and false relative to others. So neither sentence is true in the model. But $Fx \vee \neg Fx$ is true relative to every assignment function.

Exercise 9.4

There are many non-reflexive models in which $\Box p \rightarrow p$ is true at some world – for example, any non-reflexive model in which p is false at all worlds.

For the more general question, let M_1 be a model with a single world that can see itself. Let M_2 be a model with two worlds, each of which can see the other but not itself. In both models, all sentence letters are false at all worlds. The very same $\mathfrak{L}_M$-sentences are true at all worlds in these models (as a simple proof by induction shows). But the first model is reflexive and the second isn't. So there is no

$\mathfrak{L}_M$-question that is true at a world in a model iff the model's accessibility relation is reflexive.

Exercise 9.5

Use wolfgangschwarz.net/trees/.

Exercise 9.6

If a sentence is valid (in first-order predicate logic) then a fully expanded tree for the sentence will close and show that the sentence is valid. But if a sentence is not valid, the tree might grow forever. There is no algorithm for detecting whether a tree will grow forever.

Exercise 9.7

(a) $\Box Fa$
a: John, F: – is hungry.
(Might be classified as either *de re* or *de dicto*.)

(b) $\Box\forall x(Fx \rightarrow Gx)$
F: – is a cyclist, G: – has legs.
This is *de dicto*. Also correct (but different in meaning) is the *de re* translation $\forall x(Fx \rightarrow \Box Gx)$. Close but incorrect (and *de re*): $\forall x\Box(Fx \rightarrow Gx)$.

(c) $\forall x(Fx \rightarrow \Diamond Gx)$
F: – is a day, G: – is our last day.
This is *de re*. The English sentence could also be understood *de dicto*, as $\Diamond\forall x(Fx \rightarrow Gx)$, but that would be a very strange thing to say.

(d) $\forall x\, \mathsf{O}(Fx \rightarrow Gx)$
F: – wants to leave early, G: – leaves quietly.
Even better, if we can use the conditional obligation operator: $\forall x\, \mathsf{O}(Gx/Fx)$.
These aren't too far off either: $\forall x(Fx \rightarrow \mathsf{O}\, Gx)$, $\mathsf{O}\, \forall x(Fx \rightarrow Gx)$.
All of these are *de re*.

(e) $\forall x(\exists y(Fy \land Hxy) \rightarrow \mathsf{P}\, Gx)$
F: – is a ticket, G: – enters, H: – bought –.

Perhaps even better: $\forall x\, \mathsf{P}(Gx / \exists y(Fy \wedge Hxy))$. Both of these are *de re*.

You could translate 'bought a ticket' as a simple predicate here; you could also use a temporal operator to account for the past tense of 'bought' (but it's confusing to use two different kinds of 'P' in one sentence).

(f) $\mathsf{F}\forall x(Fx \rightarrow Gx)$
F: – is rich, G: – is poor.

Assuming that 'rich' and 'poor' are incompatible, this is equivalent to $\mathsf{F}\forall x \neg Fx$. The more natural reading of the sentence can only be captured with the 'now' operator: $\mathsf{F}\forall x(\mathsf{N}\, Fx \rightarrow Gx)$.

Both translations are *de re*.

Exercise 9.8

See the previous answer.

Exercise 9.9

Use wolfgangschwarz.net/trees/.

Exercise 9.10

We assume that some branch on a tree contains nodes $b = c$ and A. We have to show that we can add $A[b//c]$ without using the second version of Leibniz' Law.

k.	$b = c$	
n.	A	
m.	$b = b$	(SI)
m+1.	$c = b$	(k, m, LL (first version))
m+2.	$A[b//c]$	(m+1, n, LL (first version))

Exercise 9.11

(a)

1.	$a = a$	(SI)
2.	$\forall x\, x \neq a \to a \neq a$	(UI)
3.	$\neg\forall x\, x \neq a$	(1, 2, CPL)
4.	$\neg\exists x\, x = a \leftrightarrow \forall x\, x \neq a$	($\forall\exists$)
5.	$\exists x\, x = a$	(3, 4, CPL)
6.	$\Box\exists x\, x = a$	(5, Nec)

(b) There are many correct answers. For example: historians debate whether Homer ever existed. If a translates 'Homer' then $\exists x\, x = a$ is arguably false if Homer isn't a real person. Since the available evidence is compatible with $\neg\exists x\, x = a$, the sentence $\Box\exists x\, x = a$ is false on an epistemic interpretation of the box.

Where does the proof go wrong? Each of steps 1, 2, and 6 might be blamed.

Exercise 9.12

(a) $\exists x\exists y(Fx \land Fy \land x \neq y \land \forall z(Fz \to (z = x \lor z = y)))$
(b) $\forall x\forall y\forall z\forall v(Fx \land Fy \land Fz \land Fv \to (x = y \lor x = z \lor x = v \lor y = z \lor y = v \lor z = v))$

Exercise 9.13

The *de dicto* reading of (a) can be translated as

$$\Diamond\exists x(Px \land \forall y(Py \to x = y) \land x = c),$$

where 'P' translates '– is 45th US President' and 'c' denotes Hillary Clinton. The *de re* reading can be translated as

$$\exists x(Px \land \forall y(Py \to x = y) \land \Diamond x = c).$$

The answers to (b) and (c) are analogous.