Practice Final Solutions

Short Answer:

(1)

Γ |- ∃xΦx
Φ(a) |- Φ(a)
<u>Φ(a), Δ |- ψ</u>
Γ, Δ |- ψ

Restrictions: The name used in the typical instance (i.e. 'a') can't occur in Γ, the existential claim (i.e. '∃xΦ(x)'), ψ, or Δ.

(2)

True. If $A_1,\ldots,A_n$ are inconsistent then there is no model that makes $A_1,\ldots,A_n$ true and P & ~P false (since there is no model that makes $A_1,\ldots,A_n$ true). So the semantic sequent $A_1,\ldots,A_n$ |= P & ~P is valid. By the completeness of predicate logic, there is a proof from $A_1,\ldots,A_n$ to P & ~P.

(3)

Predicate logic sentences A and B are inconsistent just in case there is no model that makes A and B both true.

(4)

1	(1) P v Q	A
2	(2) P	A
3	(3) Q	A
2,3	(4) P & Q	2,3 &I
2,3	(5) P	4 &E
<u>1</u>	(6) P	1,2,2,3,5 vE

The underlined dependency is incorrect. The dependencies are 1 and 2 for line 6. Note that the desired conclusion cannot follow from the assumption of P v Q alone since P v Q is true in interpretations where P is false (and Q is true). By soundness, no proof exists.

(5)

1	(1) ~P	A
2	(2) ∃x(Fx & ~Fx)	A
2	(3) ~~P	1,2 RAA
2	(4) P	3 DN
	(5) ∃x(Fx & ~Fx) -> P	2,4 CP

RAA doesn't apply to (2) because that's not a conjunction of the form A & ~ A (it's an existentially quantified statement). The proof could be corrected as follows:

1	(1) ~P	A
2	(2) ∃x(Fx & ~Fx)	A
3	(3) Fa & ~Fa	A [for EE, choose ‘a’ so that P does not mention ‘a’]
3	(4) ~~P	1,3 RAA
3	(5) P	4 DN
2	(6) P	2,3,5 EE
	(7) ∃x(Fx & ~Fx) -> P	2,6 CP

(6)

In question #4 the bad line is (6). In question #5 there are no bad lines, rather the mistake was in the application of the RAA rule.

Translation:

(1) Every man who has a son adores him.

∀x∀y((Mx & Pxy & My & ~(x=y)) -> Axy)

(2) Every man who has a daughter adores his daughter's mother.

∀x∀y((Mx & Pxy & ~My) -> ∀z((Pzy & ~Mz & ~(z=y)) -> Axz))

(Maybe some explanation is called for here?)

(3) Everybody adores their own grandchildren.

∀x∀y∀z((Pxy & Pyz & ~(x=y) & ~(y=z) & ~(x=z) -> Axz)

(4) Every woman adores her brothers' children.

∀x∀y((~Mx & ∃z(Pzx & Pzy & My)) -> ∀z(Pyz & ~(x=z)-> Axz))

(5) No man adores children unless he has his own.

Situations that are compatible with (5):

1. A man has children but doesn't adore children.

2. A man has children and adores children.

3. A man doesn't have children and doesn't adore children.

Situations that are ruled out by (5):

4. A man doesn't have children and adores children.

Hypothesis: A rough paraphrase of (5) is:

(Suppose you are a man) Either you don't adore children or you have your own.

A test for this hypothesis: Fill in the truth values supplied by each of these scenarios. (1)-(3) should be true and (4) should be false. (And, in fact, this is the case. See below.)

1. T v F = T

2. F v T = T

3. T v F = T

4. F v F = F

Full translation:

∀x(Mx -> (∀y(∃zPzy & ~(x=y) & ~(y=z) & ~(x=z) -> (~Axy v ∃z(~(x=z) & Pxz))))

English paraphrase of the full translation: If x is a man, then for any y such that y is a child (i.e. for any y for which there is something that is a parent of y), either it's not the case that x adores y or x is a parent (i.e. there is something such that x is the parent of it).

(6)

Someone has no more than two children.

∃w∀x∀y∀z((Pwx & Pwy & Pwz & ~(w=x) & ~(w=y) & ~(w=z)) -> (x=y v y=z v x=z))

Proofs and Counterexamples:

(1)

Show: (P & ~Q) -> ~(P->Q)

1	1 P & ~Q	A
2	2. P -> Q	A
1	3. P	1 &E
1,2	4. Q	2,3 MPP

1	5. ~Q	1 &E
1,2	6. Q & ~Q	4,5 &I
1	7. ~(P -> Q)	2,6 RAA
	8. (P & ~Q) -> ~(P -> Q)	1,7 CP

Show: ~(P -> Q) -> (P & ~Q)

1	1. ~(P -> Q)	A
2	2. ~(P & ~Q)	A
3	3. P	A
4	4. ~Q	A
3,4	5. P & ~Q	3,4 &I
2,3,4	6. ~(P & ~Q) & (P & ~Q)	2,5 &I
2,3	7. ~~Q	4,6 RAA
2,3	8. Q	7 DN
2	9. P -> Q	3,8 CP
1,2	10. (P -> Q) & ~(P -> Q)	1,9 &I
1	11. ~~(P & ~Q)	2,10 RAA
1	12. P & ~Q	11 DN
	13. ~(P -> Q) -> (P & ~Q)	1,12 CP

Conjoining the last line of each proof and applying the definition of <-> gives us a proof of ~(P -> Q) <-> (P & ~Q) using only basic rules of inference.

(2)

Show: ∃x(Fx & ∀y(Gy -> Rxy)), ∀x(Fx -> ∀y(Hy -> ~Rxy)) |- ∀x(Gx -> ~Hx)

1	1. ∃x(Fx & ∀y(Gy -> Rxy))	A
2	2. ∀x(Fx -> ∀y(Hy -> ~Rxy))	A
3	3. Fa & ∀y(Gy -> Ray)	A [for EE]
2	4. Fa -> ∀y(Hy -> ~Ray)	2 UE
3	5. Fa	3 &E
2,3	6. ∀y(Hy -> ~Ray)	4,5 MPP
2,3	7. Hb-> ~Rab	6 UE
3	8. ∀y(Gy -> Ray)	3 &E
3	9. Gb-> Rab	8 UE
10	10. Gb	A
3,10	11. Rab	9,10 MPP
3,10	12. ~~Rab	11 DN
2,3,10	13. ~Hb	7,12 MTT
2,3	14. Gb -> ~Hb	10,13 CP
1,2	15. Gb -> ~Hb	1,3,14 EE
1,2	16. ∀x(Gx -> ~Hx)	15 UI

(3a)

1	1. ∀x∀y((Qxy -> ∀z(Rzx -> Rzy)) & (∀z(Rzx -> Rzy) -> Qxy))	A
1	2. ∀y((Qay -> ∀z(Rza -> Rzy)) & (∀z(Rza -> Rzy) -> Qay))	1 UE
1	3. ((Qaa -> ∀z(Rza -> Rza)) & (∀z(Rza -> Rza) -> Qaa))	2 UE
1	4. Qaa -> ∀z(Rza -> Rza)	3 &E
1	5. ∀z(Rza -> Rza) -> Qaa	3 &E
6	6. ~Qaa	A
1,6	7. ~∀z(Rza -> Rza)	5,6 MTT
8	8. Rba	A
	9. Rba -> Rba	8,8 CP
	10. ∀z(Rza -> Rza)	9 UI
1,6	11. ~∀z(Rza -> Rza) & ∀z(Rza -> Rza)	7,10 &I
1	12. ~~Qaa	6,11 RAA
1	13. Qaa	12 DN
1	14. ∀xQxx	13 UI

(3b)

DoQ: {a,b}
Ext(R): {<b,b>, <a,b>}
Ext(Q): {<a,b>, <a,a>, <b,b>}

(3c)

(i)

Show: ∀x∀y(Qxy <-> ∀z(Rzx -> Rzy)) |- ∃y∀xRxy -> ∃y∀xQxy

1	1. ∃y∀xRxy	A
2	2. ∀xRxb	A
3	3. ∀x∀y(Qxy <-> ∀z(Rzx -> Rzy))	A
3	4. ∀y(Qay <-> ∀z(Rza -> Rzy))	3 UE
3	5. Qab <-> ∀z(Rza -> Rzb)	4 UE
3	6. (Qab -> ∀z(Rza -> Rzb)) & (∀z(Rza -> Rzb) -> Qab)	5 <-> df
3	7. ∀z(Rza -> Rzb) -> Qab	6 &E
8	8. Rca	A
2	9. Rcb	2 UE
2	10. Rca -> Rcb	8,9 CP
2	11. ∀z(Rza -> Rzb)	10 UI
2,3	12. Qab	7,11 MPP
2,3	13. ∀xQxb	12 UI
2,3	14. ∃y∀xQxy	13 EI
1,3	15. ∃y∀xQxy	1,2,14 EE
3	16. ∃y∀xRxy -> ∃y∀xQxy	1,15 CP

(ii)

DoQ: {a}
Ext(Q): {<a,a>}
Ext(R): ∅

(4)

DoQ: {a,b}
Ext(F): {a}
Ext(G): ∅

Metatheory:

(1)

I'm going to show by induction that there is a truth function with two inputs that can't be expressed using the connective v.

Base case: Show that P & Q can't be expressed by P v Q.

We can represent the values of the truth function picked out by '&' with P and Q as inputs as follows:

T & T = T
T & F = F
F & T = F
F & F = F

and we can represent the values of the truth function picked out by 'v' with P and Q as inputs as follows:

T v T = T
T v F = T
F v T = T
F v F = F

And, plainly, '&' represents a truth function that can't be expressed using a single instance of the connective v.

But maybe some more complex combination of v, P, and Q would do the trick. We want to rule out this possibility.

Let's consider how we might build sentences of greater complexity with these parts. Consider these sentences, which are just slightly more complex:

(P v Q) v Q

(P v Q) v P

Now, given that v is associative and commutative (a quick wiki search will tell you what this means – but it amounts to the claim that the order of operations and the brackets don't make a difference to the truth value of the sentence) we have these sentences, essentially:

P v (Q v Q)

Q v (P v P)

And since Q v Q is equivalent to Q and P v P is equivalent to P, if we substitute these equivalents, we get:

P v Q

Q v P (which is just equivalent to P v Q).

So, these ways of increasing in complexity just give us sentences that are reducible to a sentence that we know isn't expressed with a single instance of v.

Since anyway of increasing the complexity of these sentences little by little gives us something that is so reducible, we may conclude that & is a truth function taking two inputs that is not expressed by v.

(2)

Soundness of predicate logic:

For an arbitrary string of predicate logic sentences $A_1,\ldots,A_n$, if there is a proof of B from $A_1,\ldots,A_n$, then there is no model relative to which $A_1,\ldots,A_n$ are true and B false (i.e. $A_1,\ldots,A_n \models B$) .

Completeness of predicate logic:

If there is no model relative to which $A_1,\ldots,A_n$ are true and B false, then there is a proof from $A_1,\ldots,A_n$ to B.

RAA:

$\Phi \vdash \Phi$
$\underline{\Gamma \vdash \psi \ \& \sim\psi}$

$\Gamma - \{\Phi\} \vdash \sim\Phi$

Suppose that there is a model such that $\Phi \models \Phi$ and $\Gamma \models \psi \ \& \sim\psi$ relative to that model. Since there is no model that makes $\psi \ \& \sim\psi$ true and since $\Gamma \models \psi \ \& \sim\psi$, this model does not make (the members of) Γ true. What do we know about $\Gamma - \{\Phi\} \vdash \sim\Phi$? Well, if this model does not make $\Gamma - \{\Phi\}$ true, then trivially $\Gamma - \{\Phi\} \models \sim\Phi$.

But if this model makes $\Gamma - \{\Phi\}$ true, then this is only because Φ made Γ inconsistent—made it such that the members of that set were not true relative to that model. And so, if this model makes $\Gamma - \{\Phi\}$ true, then it also makes $\sim\Phi$ true. How do we know that?

Well, if Φ is in Γ and $\Gamma - \{\Phi\}$ is consistent when Φ is removed then our model doesn't make Φ true…if it did then $\Gamma \cup \{\Phi\}$ would be consistent and it's not. Since any model that doesn't make Φ true does make $\sim\Phi$ true, we know that our model makes $\sim\Phi$ true in this case.

Since this was an arbitrary model, we've shown that if $\Phi \models \Phi$ and $\Gamma \models \psi \ \& \sim\psi$ then $\Gamma - \{\Phi\} \models \sim\Phi$. So, RAA is sound.

(3)

We want to formalize the sentence: "There are at most 2 objects." This sentence will clearly be true in any schminterpretation by the definition of schminterpretation.

If we could use equality, it would be pretty straightforward. The sentence

∃x∃y∀z (z=x v z=y)

works nicely. Since we can't use equality by stipulation, we just need to take this sentence, replace '=' with an uninterpreted 2-place relation, 'R' say, and add axioms to ensure that R behaves like equality. Namely we want R to be refelxive, symmetric, and transitive. Thus:

∃x∃y∀z (Rzx v Rzy) & ∀x∀y∀z(Rxx & (Rxy->Ryx) & ((Rxy &Ryz)->Rxz))

fits the bill.